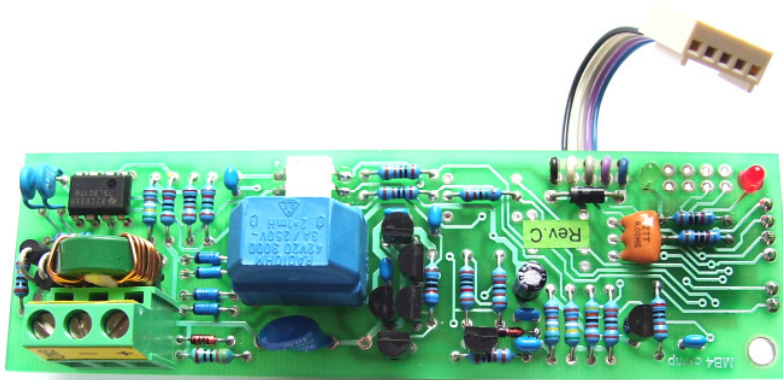


# CM485

**Communication module for  
RS-485 (Modbus) interface  
ThermoControl Plus M**



**Rev.C<sub>2</sub>**



Subject to technical changes. Version 12/10

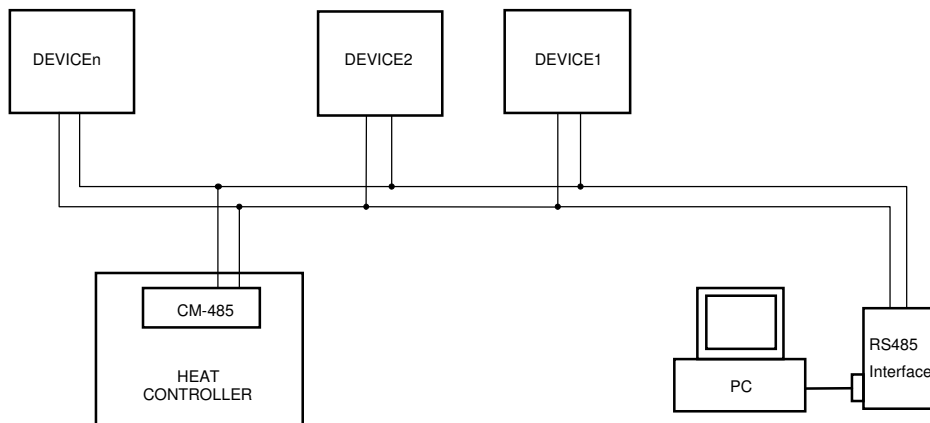
## CONTENTS

How to use CM485 .....	3
Installation.....	3
Technical Parameters .....	5
Connection.....	5
Communication .....	6
Register reading .....	6
Register writing .....	7
Test order .....	8
Error Codes: .....	8
Register addresses:.....	9
ThermoControl Plus M1 .....	9
ThermoControl Plus M2 .....	11
ThermoControl Plus M4 .....	14
Common Register Addresses:.....	18
Description of Clock (RTC) registers .....	19
Time program registers overview .....	20
Holiday program registers overview .....	21
Appendix 1 .....	23
Codes of the buttons .....	23

## How to use CM485

The module can be built into any type of the heat controllers in a very easy way. The controller will automatically recognise the presence of the module. The module makes it possible to see through the RS485 network the measured values and to modify any control parameter.

Any number of devices can be connected to the single twisted wire RS485 network. Every device has its own address to be able to communicate with the network controller (PC).

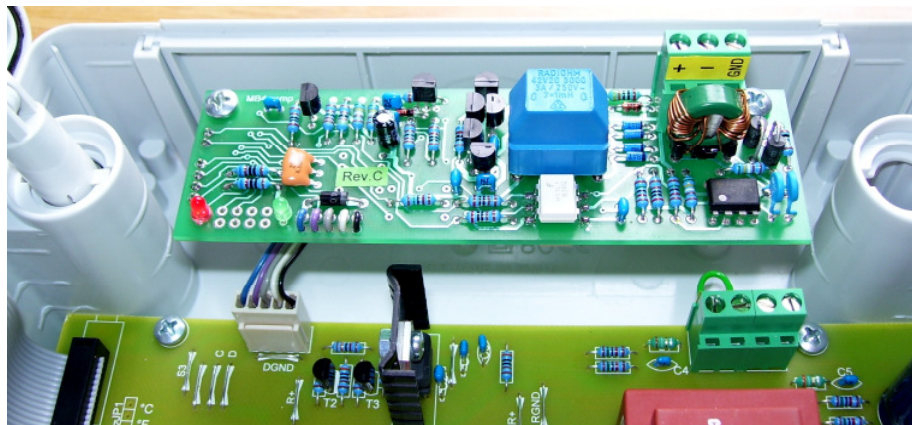


## Installation

Before installation, the mains supply must be switched off. It is not enough to switch off the I/O switch on the front panel of the controller, the outside double pole mains switch must be switched off first of all to avoid electric shock.

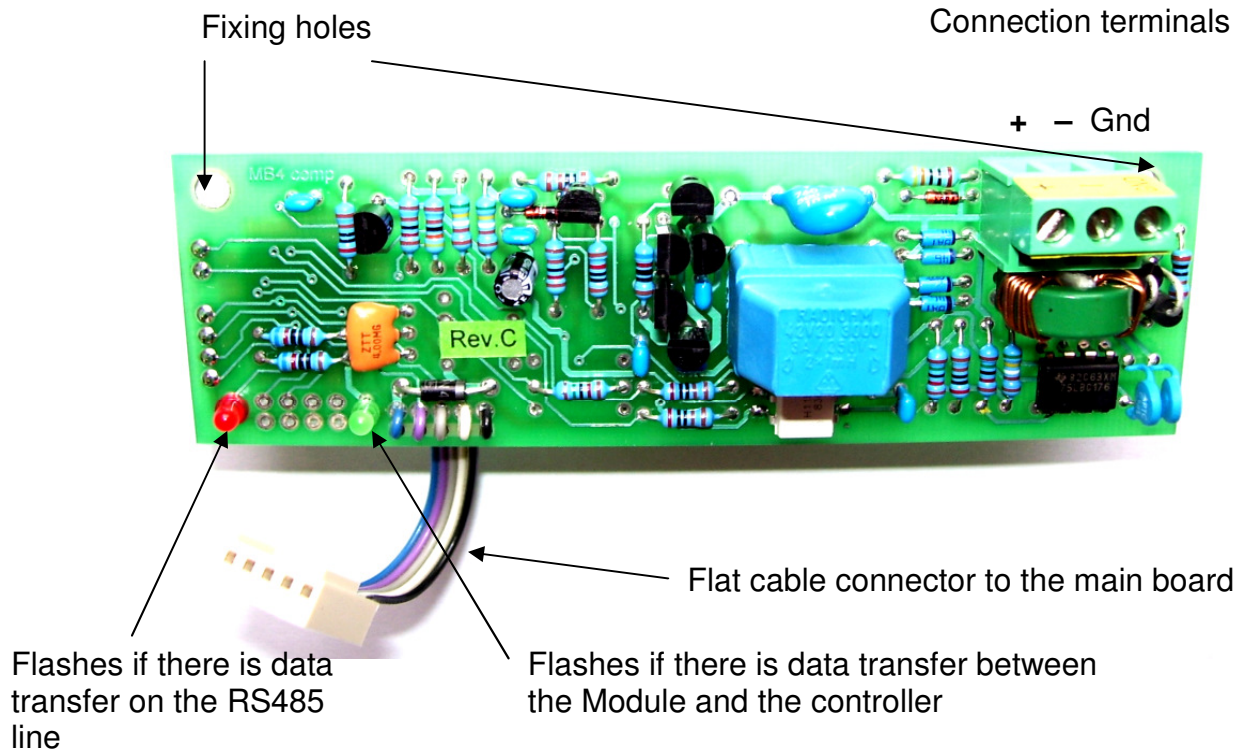
- Open the case after releasing the four big plastic screws at the corners.
- You will see two plastic spacer rods above the connection board. Fix the Module with the enclosed screws to the spacers (see picture below). If you have a controller in a wide case, use the enclosed plastic rail: fix the rail to the spacers then fix the Module to the rail.
- Snap the flat cable connector of the Module to the socket on the main connection board (see on the picture).
- Connect the communication cable (through the rubber inlets of the case) to the terminals of the module. Be careful to the correct connection order.
- Close the case and fix it with the plastic screws. Switch on the controller.
- Push the “SEL” button for 3 seconds and type in the PIN code to get access to the setup menu (refer to manual “service adjustments”)

- Push and keep pushed the **DISP** button until the parameter menu appears (Par). Push repeatedly the **SEL** button until the „br” sign appears: this is the first parameter of the communication module: „br”=bit rate. Set the desired bit rate by the +/- buttons. In a communication system all of the devices must use the same bit rate !
- Push the **SEL** button, the „bPar.n” sign will appear. Here (with the +/- buttons) the type of the parity bit can be set: n:none, o:odd, E:even.
- Push the **SEL** button, here you can select (with +/-) the format: „bASCII” means ASCII (7bits) format; or „b rtu” means RTU (8bits) format.
- Push the **SEL** button, here the network address of the controller will appear: „bA 02”. Set the desired unique address with the +/- buttons.
- When finished, push **DISP** button to go back to the base display mode.
- Leave the setup menu by pushing the “SEL” button for 3 seconds (refer to manual “service adjustments”)
- The installation of the Module is finished, it is ready to work.



## Technical Parameters

Bit rate: 600, 1200, 2400, 4800, 9600, 19200 bit/s can be selected  
Format: RTU 8 bit, or ASCII 7 bit, parity: none, odd, even  
Device address: 1-247 can be set



## Connection

It is recommended to use a shielded twisted pairs of wires to connect the RS-485 network. The shield of the cable must be connected to the "GND" terminal. If the bit rate is higher than 9600 bit/s it is recommended to connect termination resistors (120 ohm) on both ends of the network. The CM485 module doesn't contain any protection against surge voltages or lightning, so the necessary protection must be made by outside devices.

# Communication

The communication of the CM485 Modbus module is based on 16-bit registers. You can see or modify the parameters by reading or writing these registers. Each register has a 16-bit address.

Two formats can be selected, the 7-bit ASCII or the 8-bit RTU. The ASCII format uses two characters for the transfer of a number; the transfer error is checked by LRC. The RTU format can communicate at higher speed because the data is transferred in one character and the error detection is made by 16-bit CRC.

More detailed information about communication can be found at: [www.modbus.org](http://www.modbus.org)

In the following example ASCII format messages are used.

Two functions used for data transfer and one for test:

- Reading: code 03h
- Writing: code 06h
- Test: code 08h

Remark: the letter „h” after the figures always means the hexadecimal format.

The length of one ASCII message is always 17 characters; the length of the answers may be different.

## Register reading

In this example we ask for 1 register from address „0”.

The „:” is part of the message !  
The „↵” means the 0Dh+0Ah line-end signal.

**:020300000001FA↵**

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
:	0	2	0	3	0	0	0	0	0	0	0	1	F	A	0Dh	0Ah

As an ASCII code:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
3Ah	30h	32h	30h	33h	30h	30h	30h	30h	30h	30h	30h	31h	46h	41h	0Dh	0Ah

Structure:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
:	Dev addr		fcode		Reg. address			Numb. of registers				LRC		CR	LF	
					Hi	Lo	Hi	Lo								

	02	03	00	00	00	01	FAh		
--	----	----	----	----	----	----	-----	--	--

$\Sigma \text{ mod } 100h = 0$

LRC: Check sum: if we add up the marked figures (between 2-15) then the sum must be dividable by 100h without rest.  
CR: Carriage return  
LF: Line feed

In the example the "Dev addr" is the unique number of the device, the fcode=03 indicate the reading request. Reg.address=0000 gives the starting address. Number=0001 means the number of required data register (max:7Fh). This may be more than 1, in that case the answer will be longer.

The answer may be:

**:020302810375.↓**

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
:	Dev addr		Status		Length		Data				LRC		CR	LF
:	02		03		02		81		03		75		0Dh	0Ah
							Hi		Lo					

The „Length” indicates that there will follow 2 byte. The 16-bit figures will arrive in High-Low order.

Byte order of the 32-bit figures: (sample: 4byte work-hour counter)

Hi	Lo	Hi	Lo	2x16bit 32bit
Hi		Lo		

### Register writing

In this example we are writing „1” to address „2”.

**:020600020001F5.↓**

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
:	0	2	0	6	0	0	0	2	0	0	0	1	F	5	0Dh	0Ah

As an ASCII code:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
3Ah	30h	32h	30h	36h	30h	30h	30h	32h	30h	30h	30h	31h	46h	35h	0Dh	0Ah

Structure:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
:	Dev addr		fcode		Reg. address				Data				LRC		CR	LF
					Hi		Lo		Hi		Lo					

In the example the "Dev addr" is the unique number of the device.

The fcode=06 indicates the writing request.

The Reg.address gives the 16-bit address.

Data: contains the 16-bit data.

LRC, CR, LF: same as in case of reading.

Only one 16-bit register can be written at a time.

If there was no error, the answer will be the same as the order:

**:020600020001F5.↓**

## Test order

„fcode=08” serves for test purposes. The Module will repeat the order but the content of the registers will not change. This procedure can be used to check the transfer way with different bit-patterns. The address can only be 0000h.

**:02080000ABCD7E↵**

Structure:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
:	02	08	0000			X			LRC		CR	LF				

In the example the code fcode=08 indicates the test function.

X means any 16-bit figure.

LRC, CR, LF: same as above at reading.

The answer given to the test will be the same as the order:

**:02080000ABCD7E↵**

## Error Codes:

There may be three types of the error codes:

– Writing to invalid address. Error code: 02h

**:02860276↵**

– Writing to invalid data. Error code: 03h

**:02860375↵**

– Invalid function (fcode). Error code: 01h

**:02840179↵**

– The Module is busy,  
it is not able to answer. Error code: 06h

**:02830675↵**

Structure:

1	2	3	4	5	6	7	8	9	10	11
:	02	Status	Err	LRC	CR	LF				

If there is an error the upper bit of „status”=1. The following bits contain the function number where the error occurred.

Example: Status=86h error at writing, Status=83h error at reading

”Err”: contains the code (the type) of the error: 01, 02, 03, 06.



## Register addresses:

The function of the registers in the module depends on the type of the controller where the module is used. Detailed descriptions of the functions can be found in the manuals of the different controllers. Here you can find the list of addresses at the different types of controllers:

### ThermoControl Plus M1

The \*-marked addresses are „Read-only” types

- \*0000h: Hi: Controller type code (35h),  
Lo: Version number.
- 0002h: No function
- 0004h: Disable or enable buttons individually. Detailed in Appendix.
- 0006h: Simulation of pushing a button. Detailed in Appendix.
- 0008h: Special button combinations. Detailed in Appendix.
- 000Ah: Simulation of releasing a button.
- 000Ch: in Zone 1 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 000Eh: in Zone 2 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 0010h: No function
- 0012h: No function
- 0014h: No function
- 0016h: No function
- 0018h: No function
- 001Ah: No function
- 001Ch: Set value type selection in Zone 1:
  - 0000h: Based on time program,
  - 0001h: Frost protection value (continuously)
  - 0002h: Night value (continuously)
  - 0003h: Day value (continuously)
- 001Eh: Set value type selection in Zone 2: same as described at Zone 1
- 0020h: No function
- 0022h: No function
- 0024h: No function
- 0026h: No function
- 0028h: No function.
- 002Ah: No function
- 002Ch: Zone1 Day set value ×10 with plus/minus sign.  
Example: 00C7h 19.9°C
- 002Eh: Zone2 Day set value ×10 with plus/minus sign.
- 0030h: No function
- 0032h: No function
- 0034h: No function
- 0036h: No function
- 0038h: No function
- 003Ah: No function
- 003Ch: Zone1 Night set value ×10 with sign.
- 003Eh: Zone2 Night set value ×10 with sign.
- 0040h: No function
- 0042h: No function
- 0044h: No function
- 0046h: No function
- 0048h: No function
- 004Ah: No function

- 004Ch: Zone1 Frost protection value  $\times 10$  with sign. This value acts as a low limit for all of the other set values. Factory settings: 0050h 8.0 °C
- 004Eh: Zone2 Frost protection value  $\times 10$  with sign.
- 0050h: No function
- 0052h: No function
- 0054h: No function
- 0056h: No function
- 0058h: No function
- 005Ah: No function
- \*005Ch: Selected application program (P0, P1, P2) number. At this type: 0, 1, 2
- \*005Eh: Number of Zones as configured. Max: 2 at this type.
- \*0060h: Error signals: in case of an error the corresponding bit: 1  
**bit7**: Error relay is on (relay contacts closed)  
**bit6**: Sensor Error  
**bit2**: Button is stuck  
**bit1**: Alarm signal (contact input is opened)  
**bit0**: Battery is low (discharged)
- \*0062h: State of the contact inputs (a closed contact sets the bit to 1):  
**bit4**: Modulation control (only at Plus M1 type)  
**bit2**: „Error” (Störung) contact  
**bit1**: „Alarm” contact  
**bit0**: „Remote” contact
- \*0064h: Outside temperature  $\times 10$  with sign. Example: FFE2h  $-3.0^{\circ}\text{C}$   
When ATF sensor is not connected: F830h (-200.0)
- \*0066h: **bit0**: State of the ventilator relay: 1 if relay is on (relay contacts closed)
- \*0068h: State of Zone 1:  
**bit7**: 1 when Zone 1 is enabled (see: 000Ch)  
**bit6**: 1 when there is a sensor error in Zone 1  
**bit5**: 1 when „Error” (Störung) contact is closed  
**bit4**: 1 when “H1” contact is opened (only at Plus M1 type)  
**bit3**: 1 when Ventilator relay is on,  
**bit2**: 1 when heater is operated with full load (Stage 2) in Zone 1  
**bit1**: 1 when heater is operated with part load (Stage 1) in Zone 1  
**bit0**: 1 when selected control mode is 2-stage in Zone 1;  
0 when selected control mode is 1-stage in Zone 1
- \*006Ah: State of Zone 2: format same as at Zone 1.
- \*006Ch: No function
- \*006Eh: No function
- \*0070h: No function
- \*0072h: No function
- \*0074h: No function
- \*0076h: No function
- \*0078h: Zone1 Actual set value  $\times 10$  with sign; the controller presently tries to keep this temperature value.
- \*007Ah: Zone2 Actual set value  $\times 10$  with sign.
- \*007Ch: No function
- \*007Eh: No function
- \*0080h: No function
- \*0082h: No function
- \*0084h: No function
- \*0086h: No function.
- \*0088h: The measured and displayed room temperature  $\times 10$  with sign in Zone1.
- \*008Ah: The measured and displayed room temperature  $\times 10$  with sign in Zone2.
- \*008Ch: No function

- \*008Eh: No function
- \*0090h: No function
- \*0092h: No function
- \*0094h: No function
- \*0096h: No function
- \*0098h: State of the heater relays in Zone1: 1 if the relay is switched On  
     In case of P0, P1 application programs:  
         **bit7**: State of Relay 1  
     In case of P2 application program:  
         **bit7**: State of Relay 1  
         **bit6**: State of Relay 2
- \*009Ah: State of the heater relay in Zone2:  
     In case of P0, P1 application programs:  
         **bit6**: State of Relay 2
- \*009Ch: No function
- \*009Eh: No function
- \*00A0h: No function
- \*00A2h: No function
- \*00A4h: No function
- \*00A6h: No function
- \*00A8h: 0 if the maintenance warning counter is switched off (not operating).
- \*00AAh: Remaining working hours till the next maintenance time.
- \*00ACh: Selected length of the maintenance period (the number of hours).
- \*00AEh: Heating Power in Zone1 (only at Plus M1 type)
- \*00B0h: Heating Power in Zone2 (only at Plus M1 type)
- \*00C6h: Work-hour counter content for Relay 1; 4byte length
- \*00CAh: Work-hour counter content for Relay 2; 4byte length
- \*00CEh: No function
- \*00D2h: No function
- \*00D6h: No function
- \*00DAh: No function
- \*00DEh: No function
- \*00E2h: No function
- \*01C0h: Serial number of the controller, length: 6 ASCII characters.
- \*01C6h: Name of the controller, max. 10 characters, the end is indicated with „0” (TC Plus1)

**ThermoControl Plus M2**

The \*-marked addresses are „Read-only” types

- \*0000h: Hi: Controller type code (36h),  
     Lo: Version number.
- 0002h: No function
- 0004h: Disable or enable buttons individually. Detailed in Appendix.
- 0006h: Simulation of pushing a button . Detailed in Appendix
- 0008h: Special button combinations. Detailed in Appendix
- 000Ah: Simulation of releasing a button.
- 000Ch: in Zone 1 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 000Eh: in Zone 2 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 0010h: in Zone 3 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 0012h: in Zone 4 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 0014h: No function
- 0016h: No function
- 0018h: No function
- 001Ah: No function

001Ch: Set value type selection in Zone 1:  
0000h: Based on time program,  
0001h: Frost protection value (continuously)  
0002h: Night value (continuously)  
0003h: Day value (continuously)

001Eh: Set value type selection in Zone 2: same as described at Zone 1  
0020h: Set value type selection in Zone 3: same as described at Zone 1  
0022h: Set value type selection in Zone 4: same as described at Zone 1  
0024h: No function  
0026h: No function  
0028h: No function  
002Ah: No function  
002Ch: Zone1 Day set value ×10 with plus/minus sign.  
Example: 00C7h 19.9°C  
002Eh: Zone2 Day set value ×10 with plus/minus sign  
0030h: Zone3 Day set value ×10 with plus/minus sign  
0032h: Zone4 Day set value ×10 with plus/minus sign  
0034h: No function  
0036h: No function  
0038h: No function  
003Ah: No function  
003Ch: Zone1 Night set value ×10 with plus/minus sign  
003Eh: Zone2 Night set value ×10 with plus/minus sign  
0040h: Zone3 Night set value ×10 with plus/minus sign  
0042h: Zone4 Night set value ×10 with plus/minus sign  
0044h: No function  
0046h: No function  
0048h: No function  
004Ah: No function  
004Ch: Zone1 Frost protection value ×10 with sign. This value acts as a low limit for  
all of the other set values. Factory settings: 0050h 8.0°C  
004Eh: Zone2 Frost protection value ×10 with sign  
0050h: Zone3 Frost protection value ×10 with sign  
0052h: Zone4 Frost protection value ×10 with sign  
0054h: No function  
0056h: No function  
0058h: No function  
005Ah: No function  
\*005Ch: Selected application program (P0, P1...) number. At this type: 0,1,2,3  
\*005Eh: Number of Zones as configured. Max: 4 at this type.  
\*0060h: Error signals: in case of an error the corresponding bit: 1.  
**bit7**: Error relay is on (relay contacts closed)  
**bit6**: Sensor Error  
**bit2**: Button is stuck  
**bit1**: Alarm signal (contact input is opened)  
**bit0**: Battery is low (discharged)

\*0062h: State of the contact inputs (a closed contact sets the bit to 1)  
**bit4**: Modulation control (only at Plus M2 type)  
**bit2**: „Error” (Störung) contact  
**bit1**: „Alarm” contact  
**bit0**: „Remote” contact

- \*0064h: Outside temperature ×10 with sign. Example: FFE2h –3.0°C  
When ATF sensor is not connected: F830h (-200.0)
- \*0066h: **bit1**: State of the vent. Relay 3-4: 1 if relay is on (relay contacts closed).  
**bit0**: State of the vent. Relay 1-2: 1 if relay is on (relay contacts closed).
- \*0068h: State of Zone 1:  
**bit7**: 1 when Zone 1 is enabled (see: 000Ch)  
**bit6**: 1 when there is a sensor error in Zone 1  
**bit5**: 1 when „Error” (Störung) contact is closed  
**bit3**: 1 when Ventilator relay is on,  
**bit2**: 1 when heater is operated with full load (Stage 2) in Zone 1  
**bit1**: 1 when heater is operated with part load (Stage 1) in Zone 1  
**bit0**: 1 when selected control mode is 2-stage in Zone 1;  
0 when selected control mode is 1-stage in Zone 1
- \*006Ah: State of Zone 2: format same as at Zone 1
- \*006Ch: State of Zone 3: format same as at Zone 1
- \*006Eh: State of Zone 4: format same as at Zone 1
- \*0070h: No function
- \*0072h: No function
- \*0074h: No function
- \*0076h: No function
- \*0078h: Zone1 Actual set value ×10 with sign, the controller presently tries to keep this temperature value
- \*007Ah: Zone2 Actual set value ×10 with sign
- \*007Ch: Zone3 Actual set value ×10 with sign
- \*007Eh: Zone4 Actual set value ×10 with sign
- \*0080h: No function
- \*0082h: No function
- \*0084h: No function
- \*0086h: No function
- \*0088h: The measured and displayed room temperature ×10 with sign in Zone1
- \*008Ah: The measured and displayed room temperature ×10 with sign in Zone2
- \*008Ch: The measured and displayed room temperature ×10 with sign in Zone3
- \*008Eh: The measured and displayed room temperature ×10 with sign in Zone4
- \*0090h: No function
- \*0092h: No function
- \*0094h: No function
- \*0096h: No function
- \*0098h: State of the heater relays in Zone1: 1 if the relay is switched On  
In case of P0, P1 application programs:  
**bit7**: State of Relay 1  
In case of P2, P3 application programs:  
**bit7**: State of Relay 1  
**bit6**: State of Relay 2
- \*009Ah: State of the heater relays in Zone2:  
In case of P0, P1 application programs:  
**bit6**: State of Relay 2  
In case of P2 application program:  
**bit5**: State of Relay 3  
In case of P3 application program:  
**bit5**: State of Relay 3  
**bit4**: State of Relay 4

- \*009Ch: State of the heater relays in Zone3:  
In case of P0, P1 application programs:  
**bit5**: State of Relay 3  
In case of P2 application program:  
**bit4**: State of Relay 4
- \*009Eh: State of the heater relays in Zone4:  
In case of P0, P1 application programs:  
**bit4**: State of Relay 4
- \*00A0h: No function
- \*00A2h: No function
- \*00A4h: No function
- \*00A6h: No function
- \*00A8h: 0 if the maintenance warning counter is switched off (not operating)
- \*00AAh: Remaining working hours till the next maintenance time
- \*00ACh: Selected length of the maintenance period (the number of hours)
- \*00AEh: Heating Power in Zone1 (only at Plus M2 type)
- \*00B0h: Heating Power in Zone2 (only at Plus M2 type)
- \*00B2h: Heating Power in Zone3 (only at Plus M2 type)
- \*00B4h: Heating Power in Zone4 (only at Plus M2 type)
- \*00C6h: Work-hour counter content for Relay 1; 4byte length
- \*00CAh: Work-hour counter content for Relay 2; 4byte length
- \*00CEh: Work-hour counter content for Relay 3; 4byte length
- \*00D2h: Work-hour counter content for Relay 4; 4byte length
- \*00D6h: No function
- \*00DAh: No function
- \*00DEh: No function
- \*00E2h: No function
- \*01C0h: Serial number of the controller, length: 6 ASCII characters
- \*01C6h: Name of the controller, max. 10 characters, the end is indicated with „0” (TC Plus 2)

#### ThermoControl Plus M4

The \*-marked addresses are „Read-only” types

- \*0000h: Hi: Controller type code (37h),  
Lo: Version number.
- 0002h: No function
- 0004h: Disable or enable buttons individually. Detailed in Appendix.
- 0006h: Simulation of pushing a button. Detailed in Appendix
- 0008h: Special button combinations. Detailed in Appendix
- 000Ah: Simulation of releasing a button.
- 000Ch: in Zone 1 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 000Eh: in Zone 2 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 0010h: in Zone 3 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 0012h: in Zone 4 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 0014h: in Zone 5 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 0016h: in Zone 6 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 0018h: in Zone 7 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 001Ah: in Zone 8 control is enabled (On) or disabled (Off), 1: On, 0: Off
- 001Ch: Set value type selection in Zone 1:
  - 0000h: Based on time program,
  - 0001h: Frost protection value (continuously)
  - 0002h: Night value (continuously)
  - 0003h: Day value (continuously)

- 001Eh: Set value type selection in Zone 2: same as described at Zone 1
- 0020h: Set value type selection in Zone 3: same as described at Zone 1
- 0022h: Set value type selection in Zone 4: same as described at Zone 1
- 0024h: Set value type selection in Zone 5: same as described at Zone 1
- 0026h: Set value type selection in Zone 6: same as described at Zone 1
- 0028h: Set value type selection in Zone 7: same as described at Zone 1
- 002Ah: Set value type selection in Zone 8: same as described at Zone 1
- 002Ch: Zone1 Day set value ×10 with plus/minus sign.  
Example: 00C7h 19.9°C
- 002Eh: Zone2 Day set value ×10 with plus/minus sign
- 0030h: Zone3 Day set value ×10 with plus/minus sign
- 0032h: Zone4 Day set value ×10 with plus/minus sign
- 0034h: Zone5 Day set value ×10 with plus/minus sign
- 0036h: Zone6 Day set value ×10 with plus/minus sign
- 0038h: Zone7 Day set value ×10 with plus/minus sign
- 003Ah: Zone8 Day set value ×10 with plus/minus sign
- 003Ch: Zone1 Night set value ×10 with plus/minus sign
- 003Eh: Zone2 Night set value ×10 with plus/minus sign
- 0040h: Zone3 Night set value ×10 with plus/minus sign
- 0042h: Zone4 Night set value ×10 with plus/minus sign
- 0044h: Zone5 Night set value ×10 with plus/minus sign
- 0046h: Zone6 Night set value ×10 with plus/minus sign
- 0048h: Zone7 Night set value ×10 with plus/minus sign
- 004Ah: Zone8 Night set value ×10 with plus/minus sign
- 004Ch: Zone1 Frost protection value ×10 with sign. This value acts as a low limit for all of the other set values. Factory settings: 0050h 8.0°C
- 004Eh: Zone2 Frost protection value ×10 with sign
- 0050h: Zone3 Frost protection value ×10 with sign
- 0052h: Zone4 Frost protection value ×10 with sign
- 0054h: Zone5 Frost protection value ×10 with sign
- 0056h: Zone6 Frost protection value ×10 with sign
- 0058h: Zone7 Frost protection value ×10 with sign
- 005Ah: Zone8 Frost protection value ×10 with sign
- \*005Ch: Selected application program (P0, P1...) number. Here: 0,1,2,3,4,5
- \*005Eh: Number of Zones as configured. Max: 8 at this type.
- \*0060h: Error signals: in case of an error the corresponding bit: 1.  
**bit7:** Error relay is on (relay contacts closed)  
**bit6:** Sensor Error  
**bit2:** Button is stuck  
**bit1:** Alarm signal (contact input is opened)  
**bit0:** Battery is low (discharged)
- \*0062h: State of the contact inputs (a closed contact sets the bit to 1)  
**bit4:** Modulation control (only at Plus M4 type)  
**bit2:** „Error” (Störung) contact  
**bit1:** „Alarm” contact  
**bit0:** „Remote” contact
- \*0064h: Outside temperature ×10 with sign. Example: FFE2h -3.0°C  
When ATF sensor is not connected: F830h (-200.0)
- \*0066h: **bit3:** State of the vent. Relay 7-8: 1 if relay is on (relay contacts closed).  
**bit2:** State of the vent. Relay 5-6: 1 if relay is on (relay contacts closed).  
**bit1:** State of the vent. Relay 3-4: 1 if relay is on (relay contacts closed).  
**bit0:** State of the vent. Relay 1-2: 1 if relay is on (relay contacts closed).

- \*0068h: State of Zone 1:
  - bit7**: 1 when Zone 1 is enabled (see: 000Ch)
  - bit6**: 1 when there is a sensor error in Zone 1
  - bit5**: 1 when „Error” (Störung) contact is closed
  - bit3**: 1 when Ventilator relay is on,
  - bit2**: 1 when heater is operated with full load (Stage 2) in Zone 1
  - bit1**: 1 when heater is operated with part load (Stage 1) in Zone 1
  - bit0**: 1 when selected control mode is 2-stage in Zone 1;  
0 when selected control mode is 1-stage in Zone 1
- \*006Ah: State of Zone 2: format same as at Zone 1
- \*006Ch: State of Zone 3: format same as at Zone 1
- \*006Eh: State of Zone 4: format same as at Zone 1
- \*0070h: State of Zone 5: format same as at Zone 1
- \*0072h: State of Zone 6: format same as at Zone 1
- \*0074h: State of Zone 7: format same as at Zone 1
- \*0076h: State of Zone 8: format same as at Zone 1
- \*0078h: Zone1 Actual set value ×10 with sign; the controller presently tries to keep this temperature value
- \*007Ah: Zone2 Actual set value ×10 with sign
- \*007Ch: Zone3 Actual set value ×10 with sign
- \*007Eh: Zone4 Actual set value ×10 with sign
- \*0080h: Zone5 Actual set value ×10 with sign
- \*0082h: Zone6 Actual set value ×10 with sign
- \*0084h: Zone7 Actual set value ×10 with sign
- \*0086h: Zone8 Actual set value ×10 with sign
- \*0088h: The measured and displayed room temperature ×10 with sign in Zone1
- \*008Ah: The measured and displayed room temperature ×10 with sign in Zone2
- \*008Ch: The measured and displayed room temperature ×10 with sign in Zone3
- \*008Eh: The measured and displayed room temperature ×10 with sign in Zone4
- \*0090h: The measured and displayed room temperature ×10 with sign in Zone5
- \*0092h: The measured and displayed room temperature ×10 with sign in Zone6
- \*0094h: The measured and displayed room temperature ×10 with sign in Zone7
- \*0096h: The measured and displayed room temperature ×10 with sign in Zone8
- \*0098h: State of the heater relays in Zone1: 1 if the relay is switched On
  - In case of P0, P1 application programs:
    - bit7**: State of Relay 1
  - In case of P2, P3, P4, P5 application programs:
    - bit7**: State of Relay 1
    - bit6**: State of Relay 2
- \*009Ah: State of the heater relays in Zone2:
  - In case of P0, P1 application programs:
    - bit6**: State of Relay 2
  - In case of P2 application programs:
    - bit5**: State of Relay 3
  - In case of P3, P4, P5 application programs:
    - bit5**: State of Relay 3
    - bit4**: In case of P2 application programs:
- \*009Ch: State of the heater relays in Zone3:
  - In case of P0, P1 application programs:
    - bit5**: State of Relay 3
  - In case of P2 application programs:
    - bit4**: State of Relay 4
  - In case of P3 application programs:
    - bit3**: State of Relay 5



- In case of P4, P5 application programs:
  - bit3**: State of Relay 5
  - bit2**: State of Relay 6
- \*009Eh: State of the heater relays in Zone4:
  - In case of P0, P1 application programs:
    - bit4**: State of Relay 4
  - In case of P2 application programs:
    - bit3**: State of Relay 5
  - In case of P3 application programs:
    - bit2**: State of Relay 6
  - In case of P4 application programs:
    - bit1**: State of Relay 7
  - In case of P5 application programs:
    - bit1**: State of Relay 7
    - bit0**: State of Relay 8
- \*00A0h: State of the heater relays in Zone5:
  - In case of P0, P1 application programs:
    - bit3**: State of Relay 5
  - In case of P2 application programs:
    - bit2**: State of Relay 6
  - In case of P3 application programs:
    - bit1**: State of Relay 7
  - In case of P4 application programs:
    - bit0**: State of Relay 8
- \*00A2h: State of the heater relays in Zone6:
  - In case of P0, P1 application programs:
    - bit2**: State of Relay 6
  - In case of P2 application programs:
    - bit1**: State of Relay 7
  - In case of P3 application programs:
    - bit0**: State of Relay 8
- \*00A4h: State of the heater relays in Zone7:
  - In case of P0, P1 application programs:
    - bit1**: State of Relay 7
  - In case of P2 application programs:
    - bit0**: State of Relay 8
- \*00A6h: State of the heater relays in Zone8:
  - In case of P0, P1 application programs:
    - bit0**: State of Relay 8
- \*00A8h: 0 if the maintenance warning counter is switched off (not operating)
- \*00AAh: Remaining working hours till the next maintenance time
- \*00ACh: Selected length of the maintenance period (the number of hours)
- \*00AEh: Heating Power in Zone1 (only at Plus M4 type)
- \*00B0h: Heating Power in Zone2 (only at Plus M4 type)
- \*00B2h: Heating Power in Zone3 (only at Plus M4 type)
- \*00B4h: Heating Power in Zone4 (only at Plus M4 type)
- \*00B6h: Heating Power in Zone5 (only at Plus M4 type)
- \*00B8h: Heating Power in Zone6 (only at Plus M4 type)
- \*00BAh: Heating Power in Zone7 (only at Plus M4 type)
- \*00BCh: Heating Power in Zone8 (only at Plus M4 type)
- \*00C6h: Work-hour counter content for Relay 1; 4byte length
- \*00CAh: Work-hour counter content for Relay 2; 4byte length
- \*00CEh: Work-hour counter content for Relay 3; 4byte length
- \*00D2h: Work-hour counter content for Relay 4; 4byte length

- \*00D6h: Work-hour counter content for Relay 5; 4byte length
- \*00DAh: Work-hour counter content for Relay 6; 4byte length
- \*00DEh: Work-hour counter content for Relay 7; 4byte length
- \*00E2h: Work-hour counter content for Relay 8; 4byte length
- \*01C0h: Serial number of the controller, length: 6 ASCII characters
- \*01C6h: Name of the controller, max. 10 characters, the end is indicated with „0”  
(TC Plus 4)

### **Common Register Addresses:**

The following addresses have the same functions in every controller types:

- 00F0h-00F5h: Data of the Clock (Real Time Clock)
- 00F6h-0143h: Data of the time program in a zone
- 0144h-0174h: Data of the holiday program
- \*01E0h-01EFh: Data of the display segments (16x8bit)
- \*01F0h-01FFh: Data of the blinking segments (16x8bit)
- \*0200h: state of Error(bit 7) and Service (bit 6) LED when 0201h=D2h.
- \*0201h: D2h, when built-in a second generation LCD.
- \*0202h: 10h, at “Plus M” types
- \*0FFDh: Code to identify the type of the controller (without version information): on the lower 8 bits (Example: TCP-1: 0031h or 0035h).
- \*0FFEh: ID number of the CM-485 module (RevC: 0013h, RevB: 0012h).

} *Available only in  
controllers produced  
after 01.05.2008.*

The following Clock, Time program and Holiday program reading and writing possibility is available only in controllers produced after 01.05.2008.

### Description of Clock (RTC) registers

Address	00F0h		00F2h		00F4h	
	Hi	Lo	Hi	Lo	Hi	Lo
Content	Hour	Minute	Month	Day	Year	Command / Error

Commands: 22h: Load momentary RTC data into the registers (load as a snapshot).  
 41h: Write registers content into RTC (rewrite and save Clock data).  
 Error: See at chapter „Setting the Clock”

### Reading out the Clock in two steps using 22h command:

Load 0022h into register address 00F4h : 020600F40022E2↓  
 Read out data from addresses 00F0h – 00F5h : 020300F0000308↓  
 The result will contain the snapshot of the momentary Clock data: Hour, Minute, Month, Day, Year, Error.

: 020306090F03130800BF↓  
 Hour → ↑ ↑ ↑ ↑ ↑ Error (00h)  
 Minute → ↑ ↑ ↑ ↑ ↑ Year (08: 2008)  
 Month → ↑ ↑ ↑ ↑ ↑  
 Day → ↑ ↑ ↑ ↑ ↑

To get actual fresh Clock data repeat the above two steps.

### Setting the Clock - example:

Set the following Clock data: Hour:14, Minute:05, Month: 03, Day:22, Year: 2008  
 Load Hour and Minute into register 00F0h : 020600F00E05F5↓  
 Load Month and Day into register 00F2h. : 020600F20316ED↓  
 Load Year (08h) and command 41h into register 00F4h : 020600F40841BB↓  
 Loading 41h will start the Clock-set procedure.  
 Then read out data from addresses 00F4h – 00F5h : 020300F40001FE↓  
 The result will contain the Year and Error informing about the successful loading.  
 : 0203020800F1

↑ Error information

If the Error is 0 all Clock data is loaded.  
 Possible Error types and codes are listed here:  
 In case of an error all Clock data loading will be prohibited.

Error	Caused by
81h	Hour > 23
82h	Minute > 59
83h	Month > 12
84h	Too many Days
85h	Year<8, Year>50

## Time program registers overview

The table below contains the register addresses of the different time programs. All the data of a time program is stored in two 16-bit registers.

Program Nr.	Address	Program Nr.	Address
1	00F6h	11	011Eh
2	00FAh	12	0122h
3	00FEh	13	0126h
4	0102h	14	012Ah
5	0106h	15	012Eh
6	010Ah	16	0132h
7	010Eh	17	0136h
8	0112h	18	013Ah
9	0116h	19	013Eh
10	011Ah		

0142h	0143h
Zone Nr.	Command / Error

Zone No.: Between 1...8 depending on the type of the controller  
 Commands: 22h: Load data of a certain Zone into the registers.  
 41h: Write content of the registers into a certain Zone.  
 Error: See at chapter „Writing time program into a Zone”

## Reading out a time program from a certain Zone

Load 0122h into register 0142h. : 02060142012293↓  
 ↑ Zone number (here:1) ↑

After this the data of the 19 time programs on Zone 1 will be loaded to registers 00F6h-0141h. Then the time programs can be read out one by one.

For example:

Reading out time program Nr.2 from address 00FAh : 020300FA0002FF↓

In the answer we will get all data of time program Nr.2:

: 020304067C03CFA3↓

Hi	Lo	Hi	Lo
Status + Value type	Group of days	Minute (0...1439)	
06h	7Ch	03CFh	
00000110b	01111100b	975	
Active + Night value	Workdays	16 hour 15 minute	

bit2=1 Active  
 bit2=0 Inactive

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
1	1	1	1	1	0	0

[bit1-bit0]

01: Frost protection value  
 10: Night value  
 11 or 00: Day value

## Writing time program into a Zone

The switching parameters must be loaded into 00F6h-0141h.

After this the Zone Nr. and 41h must be loaded into address 0142h

Command 41h will start the writing procedure. : 02060142014174↓

To check result read out 0142h-0143h: : 020301420001B8↓

The answer will indicate Zone Nr. and Error state

: 0203020100F8

↑ Error information

Writing is completed if Error is 0.

Possible Error types and codes are listed here:

In case of any error the writing of the good data will also be prohibited. Data checking will be carried out only at active-state programs.

The same program can be writing into different zones. To do this write new data only into 0142h-0143h registers.

Error	Caused by
82h	Minute>1439
84h	Group of days=0
86h	Inactive program is followed by an active one
87h	Two different programs at the same time
88h	The given zone number is not existing in the unit
89h	A manual setting is presently going on at the unit

## Holiday program registers overview

The table below contains the register addresses of the different holiday programs..

All the data of a holiday program is stored in three 16-bit registers.

Program Nr	Address	Program Nr.	Address
1	0144h	5	015Ch
2	014Ah	6	0162h
3	0150h	7	0168h
4	0156h	8	016Eh

0174h	0175h
On / Off	Command / Error

On / Off: The switched-on program is indicated by a bit 1

Commands: 22h: Load the eight holiday programs into the registers.

41h: Write content of the registers into the eight holiday programs.

Error: See at chapter „Writing the holiday program”

## Reading out the holiday program

Load 0022h into register 0174h. :02060174002261↓

As a result the data of the eight holiday programs will be loaded to register area 0144h-0173h.

To read out the data of holiday program 1 from address 0144h :020301440003B4↓

In the answer we will get all data of holiday program 1:

Content	Value
Start day	18
End day	18
Start month	03
End month	03
Start year	2008
End year	2008

:020306121203030808BB↓

Read out address 0174h :02030174000181↓

It will show which ones of the 8 programs are presently switched on (On/Off Mask):

:020302A00059↓

HOL.	1	2	3	4	5	6	7	8
	80h	40h	20h	10h	8h	4h	2h	1h
On	x		x					

## Writing the holiday program

Load the table (0144h – 0173h) with the necessary data in the following order: start day, end day, start month, end month, start year, end year. Year must be given as 08 not 2008.

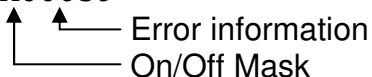
After this write into address 0174h the "On/Off Mask" (see above) and 41h.

Command 41h will start the writing procedure. :02060174A041A3↓

Then read out area 0174h – 0175h. :02030174000186↓

In the answer we will get the On/Off Mask and the possible errors.

:020302A00059



Error	Caused by
83h	Month>12
84h	Too many days
85h	Year<8, Year>50
89h	A manual setting is presently going on at the unit
8Ah	Ending date is earlier than starting date

If the value of the error is 0 the writing is carried out.

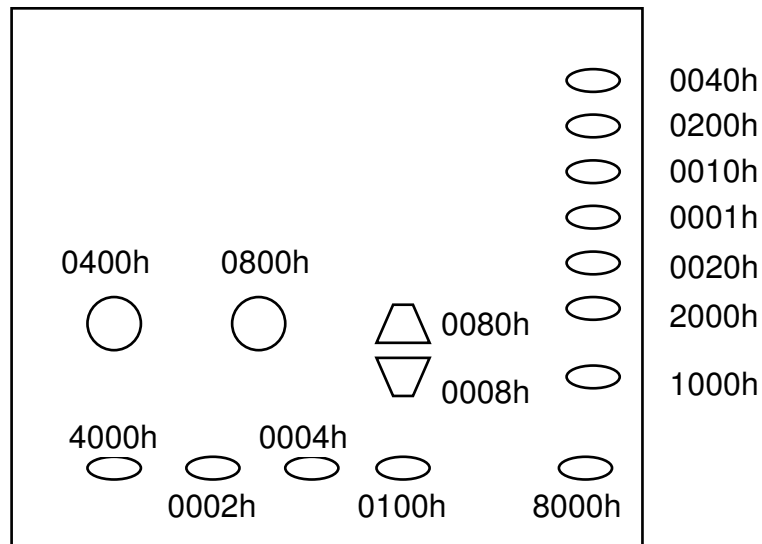
Possible Error types and codes are listed here.

In case of any error the writing of the good data will also be prohibited. Data checking will be carried out only at the on-state programs.

## Appendix 1

### Codes of the buttons

On the schematic picture of a certain controller the buttons can be assigned by the codes given below:



When switching on the power the register 0004h will be loaded with FFFFh. This ensures that all the buttons on the control panel will be enabled.

If we want to disable any of the buttons, the code of that button must be subtracted from FFFFh and the resulting value must be written back to register 0004h. After this procedure the disabled buttons will not work. If we write 0000h into register 0004h: all of the buttons will be disabled.

After switching the power Off/On the buttons (all of them) will be enabled again.

If we write a button code into register 0006h: this will simulate a press on that button (even if the button is presently disabled).

Special combination of buttons:

Values can be written into 0008h register, the result:

- 0001h: Simulates that **DISP** button is kept pressed to start "Par" menu.
- 0002h: Simulates that **SEL+Clr** buttons are pressed to start „LOAd” menu.
- 0003h: Simulates that **++Clr** buttons are pressed to start „SAVE” menu.
- 0004h: Simulates that **ZONE** button is kept pressed.
- 0005h: Simulates that **SEL** button is kept pressed.