EWCM 400
Modbus Serial Communication Protocol

1 CONTENTS

2 HOW TO USE THIS MANUAL

This manual is designed to permit quick, easy reference with the following features:

References

References column**:**
A column to the left of the text contains *references* to subjects discussed in the text to help you locate the information you need quickly and easily.

Cross references

Cross references:
All words written in *italics* are referenced in the subject index to help you find the page containing details on this subject; supposing you read the following text:
”when the alarm is triggered, the compressors will be shut down”
The italics mean that you will find a reference to the page on the topic of compressors listed under the item compressors in the index.
If you are consulting the manual “on-line” (using a computer), words which appear in italics are hyperlinks: just click on a word in italics with the mouse to go directly to the part of the manual that discusses this topic.

Icons for emphasis

Some segments of text are marked by icons appearing in the *references* column with the meanings specified below:

☞

Take *note*: information on the topic under discussion which the user ought to keep in mind

💡

Tip: a recommendation which may help the user to understand and make use of the information supplied on the topic under discussion.

⚠

Warning! : **information which is essential for preventing negative consequences for the system or a hazard to personnel, instruments, data, etc., and which users MUST read with care.**

3 FUNCTIONS AND IMPLEMENTED AREAS

This section describes the EWCM 400 communication and interface modes, using Modbus protocol.
Knowledge of specific protocol principles is required in order to be able to read and understand this section, therefore the reader is advised to refer to the “Modbus Protocol” general manual.

3.1 Transmission Format

Configuration of the serial port

The protocol uses the binary method (RTU) with byte composed as follows:
8 data bit, 1 even parity bit, 1 stop bit.

Note: The communication speed must be set to 9600 baud

3.2 Configuration parameters

To operate on prameters listed below. Please look at EWCM 400 Usrer Manual to see complete list of *parameters*. If one or more *parameters* in this category are modified, the controller must be switched off after the modification and switched on again to ensure correct operation.
Note: If *parameters* in this category are modified, the controller must be turned off after the modification and turned on again to guarantee correct functioning.

Pa H22

Serial protocol configuration (not used)
0= Standard
1= enabling Modbus protocol

Pa H26
Pa H27

Family serial address,
Device serial address
May be used to select serial address.
***Note*:**

- **at least one of *parameters* PA H44-45 should be <> 0**
- **Maximum TX & RX buffer dimension is 38 byte**

3.2.1 Table of parameters

EWCM 400 communication and interface modes, using Modbus protocol.*parameters* are listed in the table below.

Configuration parameters

CONFIGURATION PARAMETERS*				
Par.	Description	Value	Limits	Unit of measurement
Pa H22	Configuration of serial protocol		0 ÷ 1	Num
Pa H26	Family serial address		0 ÷ 14	Num.
Pa H27	Device serial address		0 ÷ 14	Num.

* If *parameters* in this category are modified, the controller must be turned off after the modification and turned on again to guarantee correct functioning.

3.3 Logic areas

EWCM 400 provides 4 *logic areas* where Modbus functions can be used:

Area no.	**Area**
2	*PARAMETERS*
3	EEPROM
1	*RAM*
4	*VER*
5	VISIBILITY OF *PARAMETERS* AND SUB-MENUS

3.3.1 Parameters

Area Index: 2
In this area the EWCM 400 characteristic *parameters* can be read and written, using Modbus functions 03 (hex) and 10 (hex).

The first parameter (index 0 in the EWCM 400 manual and Param manager) has a modbus index of 1; the various *parameters* will be addressed in the following manner (refer to the "data field" in the Modbus general manual):

EWCM 4xx Models

Index	Parameter	Address
0	Pa H01	0000 1000 0000 0001 (08 01 hex)
.....	.....	.....
37	Pa H38	0000 1000 0010 0110 (08 26 hex)
38	Pa A01	0000 1000 0010 0111 (08 27 hex)
.....	.....	.....
50	Pa A13	0000 1000 0011 0011 (08 33 hex)
51	Pa C01	0000 1000 0011 0100 (08 34 hex)
....	....	....
54	Pa C04	0000 1000 0011 0111 (08 37 hex)
55	Pa F01	0000 1000 0011 1000 (08 38 hex)
....	....	....
69	Pa F15	0000 1000 0100 0110 (08 46 hex)
70	*LABEL* SeT	0000 1000 0100 0111 (08 47 hex)
71	*LABEL* bnd	....
72	*LABEL* tP	....
73	*LABEL* Err	....
74	*LABEL* Id	....
75	*LABEL* PAr	....
76	*LABEL* SCF	....
77	*LABEL* StA	....
78	*LABEL* PSS	....
79	*LABEL* OHr	....
80	*LABEL* rES	....
81	*LABEL* CnF	....
82	*LABEL* CP	....
83	*LABEL* FAn	....
84	*LABEL* ALL	....
85	*LABEL* UL	....
86	*LABEL* dL	....
87	*LABEL* On	....
88	*LABEL* OFF	....
89	*LABEL* H01	....
90	*LABEL* H02	....
91	*LABEL* H03	....
92	*LABEL* H04	....
93	*LABEL* CH1	....
94	*LABEL* CH2	....
95	*LABEL* CH3	....
96	*LABEL* CH4	....
97	*LABEL* y	....
98	*LABEL* n	0000 1000 0110 0011 (08 63 hex)

3.3.2 Ram

Area Index: 1
In this area the *ram* memory values (organised by bytes) can be read and written, using Modbus functions 03 (hex) and 10 (hex).

The following tables show the addresses of the *Ram* memory principal areas of interest:

3.3.3 Display and LED status

Test Bit

***Add. :* 5B (hex)**		**States**	
n_bit	***Label***	***meaning***	***active at***
Bit_0			
Bit_1	COLL	Puts the machine in test mode and stops all regulators	1
Bit_2			
Bit_3			
Bit_4			
Bit_5			
Bit_6			
Bit_7			

To modify the output state (digital or analogue) and the state of display, locate this bit at 1; in this event all regulators are disabled.

***Add. :* 9E (hex)**		**Machine status**	
n_bit	*Label*	***meaning***	***active at***
Bit_0	State		See *note*
Bit_1	State		See *note*
Bit_2	On off		1
Bit_3	DP_1	Decimal visualization	1
Bit_4	DP_2	Centesimal visualization	1
Bit_5			
Bit_6			
Bit_7			

Note

this byte allows to modify controller's state
Bit 0 –1:

- 00 = OFF
- 01 = cooling
- 10 = heating
-

Bit 2:

- 1 = ON
- 0 = OFF

Bit 5 – 7:
not used
EEPROM data saving is automatic

Display

***Add. :* 72-74 (hex)**	**KPS_RAMD display**

LED

***Add. :* 75 (hex)**		**KPS_RAMD LED:**	
n_bit	***Label***	***meaning***	***active at***
Bit_0	LED_COMP3	Compressor LED 3 ON	1
Bit_1	LED_MENU	Menu LED	1
Bit_2	LED_COMP	Compressor LED 1 ON	1
Bit_3	SEGNO_MENO	Minus sign LED	1
Bit_4	LED_COMP4	Compressor LED 4 ON	1
Bit_5	LED_COMP2	Compressor LED 2 ON	1
Bit_6			
Bit_7			

3.3.4 Analogue Inputs

***Add. :* 6D (hex)**	ST1 Input more significative byte

***Add. :* 6E (hex)**	ST1 Input less significative byte

***Add. :* 6F (hex)**	ST2 Input more significative byte

***Add. :* 70 (hex)**	ST2 Input less significative byte

In order to calculate the probe's value multiply the more significative byte x 256 and add the less significative byte.

3.3.5 Analogue Output

Add. : **B0 (hex)**	**VELPER** fan speed proportional value

3.3.6 Digital inputs

Add. : **62 (hex)**		***Digital inputs***	
n_bit	***Label***	***Meaning***	***active at***
Bit_0	M_DIG_ID7	Digital input ID7	1
Bit_1	M_DIG_ID6	Digital input ID6	1
Bit_2	M_DIG_ID3	Digital input ID3	1
Bit_3	M_DIG_ID4	Digital input ID5	1
Bit_4	M_DIG_ID5	Digital input ID1	1
Bit_5	M_DIG_ID2	Digital input ID2	1
Bit_6	M_DIG_ID1	Digital input ID1	1
Bit_7	M_DIG_ID2	Digital input ID2	1

3.3.7 Digital outputs

Add. : **9F (hex)**		***Digital outputs***	
n_bit	***Label***	***Meaning***	***active at***
Bit_0	RL1	Relay 1	1
Bit_1	RL2	Relay 2	1
Bit_2	RL3	Relay 3	1
Bit_3	RL4	Relay 4	1
Bit_4	RLALL	24 V alarm output	0
Bit_5			
Bit_6			
Bit_7			

3.3.8 Alarms and resource blocks

AUTOMATICALLY RESET ALARMS

The alarm bit remains active while alarm conditions are active

Auto reset:
Alarms 1

Add. : **B8 (hex)**		**Automatic alarms 1**	
n_bit	***Label***	***Meaning***	***active at***
Bit_0	M_AONOFF	Remote ON-OFF	1
Bit_1	M_ADALTA	high pressure digital alarm	1
Bit_2	M_ADBASSA	low pressure digital alarm	1
Bit_3	M_ ANALTA	high pressure analogue alarm	1
Bit_4	M_ANBASSA	low pressure anlaogue alarm	1
Bit_5	M_AST1	Probe 1 alarm	1
Bit_6	M_AST2	Probe 2 alarm	1
Bit_7	M_ATCOMP	thermal switch compressor 1	1

Auto reset:
Alarms 2

Add. : **B9 (hex)**		**Automatic alarms 2**	
n_bit	***Label***	***Meaning***	***active at***
Bit_0	M_ATCOMP2	thermal switch compressor 2	1
Bit_1	M_ATCOMP3	thermal switch compressor 3	1
Bit_2	M_ATCOMP4	thermal switch compressor 4	1
Bit_3			
Bit_4			
Bit_5			
Bit_6			
Bit_7			

MANUALLY RESET ALARMS

The alarm bit activates immediately when the alarm switches from auto reset to manual.

Manual reset:
Alarms 1

Add. : **BA (hex)**		**Automatic alarms 1**	
n_bit	***Label***	***Meaning***	***active at***
Bit_0	M_AONOFF	Remote ON-OFF	1
Bit_1	M_ADALTA	high pressure digital alarm	1
Bit_2	M_ADBASSA	low pressure digital alarm	1
Bit_3	M_ ANALTA	high pressure analogue alarm	1
Bit_4	M_ANBASSA	low pressure anlaogue alarm	1
Bit_5	M_AST1	Probe 1 alarm	1
Bit_6	M_AST2	Probe 2 alarm	1
Bit_7	M_ATCOMP	thermal switch compressor 1	1

Manual reset: Alarms 2

Add. : **BB (hex)**		**Automatic alarms 2**	
n_bit	***Label***	***Meaning***	***active at***
Bit_0	M_ATCOMP2	thermal switch compressor 2	1
Bit_1	M_ATCOMP3	thermal switch compressor 3	1
Bit_2	M_ATCOMP4	thermal switch compressor 4	1
Bit_3			
Bit_4			
Bit_5			
Bit_6			
Bit_7			

3.3.9 Ver

Area Index: 4
In this area the mask values and device version can be read, using Modbus function 04 (hex).

These values can be read using the following logic addresses (refer to the "data field" in the Modbus general manual):

Index	***Ver***	**Address**
1	Maschera e versione (Mask & Version)	0100 1000 0000 0001 (48 01 hex)

Answer word includes in lower byte the version and in upper byte the family address.

3.3.10 Visibility of parameters and sub-menus

Area Index: 5
In this area, a “visibility value” may be assigned to each parameter, or *label*, as described below:

Label

Value	**Meaning**
0003	Parameter or *label* visible at all times
0258	Parameter or *label* visible if user password entered correctly (password = Pa H28)
0770	Parameter or *label* visible if user password entered correctly (password = Pa H28). Parameter cannot be modified.
0768	Parameter visible from PC only.

Some visibility settings are factory set.

3.4 Example

Let us suppose that we want to write a value of 30 in the EWCM 400 H03 parameter, with a network (slave) address of 1:

- First of all we must determine the parameter area index: 2
- This index is expressed with 6 bits: 000001
- Then we must determine the function code necessary to write in the index 2 (*parameters*) area: 10 (hex)
- Determine the index of parameter H03: 5
- This index is expressed with 10 bits: : 0000000101
- We combine the area index and the parameter index to form 2 bytes: 0000 1000 0000 0101 (08 05 hex)

The function frame is composed according to the modbus function (10 hex) (see the modbus manual):

Field	**Query (hex)**	**RTU 8-bit (binary)**	**Description**
Slave address	01	0000 0001	Address of the slave in the network
Function code	10	0000 1111	Function code
First bit address	08 05	0000 1000 0000 0101	Hexadecimal value to indicate the logic area 1 and the index 05
Number of variables	00 01	0000 0000 0000 0001	Hexadecimal value to indicate the number of variables to write
Byte count	02	0000 0010	No. of bytes contained in the data field
Force data	00	0000 0000	If the area is a set of bytes this field is always on zero
Data	1E	0001 1110	Value to write: 30
CRC	xx xx	xxxx xxxx xxxx xxxx	CRC value calculated using appropriate algorithm (see general modbus manual)

Therefore the function to send will be (in hexadecimals):
01 10 08 05 00 01 02 00 1E xx xx

Whereas the response will be:

Field	Query (hex)	RTU 8-bit (binary)	Description
Slave address	01	0000 0001	Address of the slave in the network
Function code	10	0000 1111	Function code (echo)
First bit address	08 35	0000 1000 0000 0101	Hexadecimal value to indicate the logic area 1 and the index 05 (echo)
Number of variables written	00 01	0000 0000 0000 0001	Hexadecimal value to indicate the number of variables written
CRC	xx xx	xxxx xxxx xxxx xxxx	CRC codified by the device

With the following frame:
01 10 08 05 00 01 xx xx

3.4.1 Additional Examples

Other examples of function frames and response:

byte reading with $A3 on the slave with address 1(xx=data)

- 01 03 04 A3 00 01 query
- 01 03 02 00 xx response

byte writing with $A3 on the slave with address 1(xx=data)

- 01 10 04 A3 00 01 02 00 xx query
- 01 10 04 A3 00 01 response

3.5 Error codes

If an error occurs, the EWCM 400 device provides one of the following *error codes* in the data field:

error code	error	description
1	function error	The function in question has not been recognised; the response is only given for packets formed in the same way as for function 16
2	address error	The area does not correspond with the function, index 0, index non-existent in the requested area
3	data error	Too many data requested for the area content, tx buffer exceeded in the response. data counter incompatible with data number to be written

3.6 Timing

the minimum time between a packet sent by the master and the slave response is 4 msec.
the DTR is raised by approx. 800µsec before slave transmission starts.

3.7 Transmission Format

Configuration of the serial port

It is important to underline that:
The protocol uses the binary method (RTU) with byte composed as follows:
8 data bit, 1 even parity bit, 1 stop bit.

Note: The communication speed must be set to 9600 baud

4 DISCLAIMER

This manual and its *contents* remain the sole property of **Eliwell & Controlli s.r.l.** and shall not be reproduced or distributed without authorization. Although great care has been exercised in the preparation of this document, **Eliwell & Controlli s.r.l.**, its employees or its vendors, cannot accept any liability whatsoever connected with its use **Eliwell & Controlli s.r.l.** reserves the right to make any changes or improvements without prior notice.

5 ANALITIC INDEX

Eliwell & Controlli s.r.l.
Via dell'Industria, 15 Zona Industriale Paludi
32010 Pieve d'Alpago (BL) ITALY
Telephone +39 0437 986111
Facsimile +39 0437 989066
Internet http://www.eliwell.it

Technical Customer Support:
Email: techsuppeliwell@invensys.com
Telephone +39 0437 986300

Invensys Controls Europe
Part of the Invensys Group

EWCM 400 Modbus
2005/1/0
Cod: 8MA10020