KOSMOS SERIE

CODE: 30727253 EDITION: 20-12-2007

INSTRUCTIONS MANUAL
PROTOCOL MODBUS-RTU

MICRA-M

DITEL

INDEX

1.2. MODBUS RTU PROTOCOL

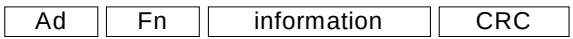
The modbus RTU protocol is a serial data transmission format widely used in communications with PLC's but easy adaptable to other types of remote units thanks to its particular message structure (it doesn't operate with variables but with memory addresses).

The implementation of a universal standard protocol such as modbus allows the instrument be connected to existing systems without need for creating specific communication logic programs.
Moreover, the quantity and variety of processable data can be infinite since it is no more necessary to specify the desired parameter or parameters but only the first memory address and the quantity to transmit.

The following definitions of modbus protocol are given adapted to the KOSMOS SERIES instruments.

The MODBUS RTU protocol has no delimiter character at the beginning and the end of a message.
Each frame must be preceded by a silent interval of at least 3, 5 character times and must be finished with a silent of the same duration.

The first character of a frame is the slave address, followed by the function number, the information field and two bytes of error check code (CRC).



One character is composed of 10 bits: 1 start bit, 8 data bits and 1 stop bit.

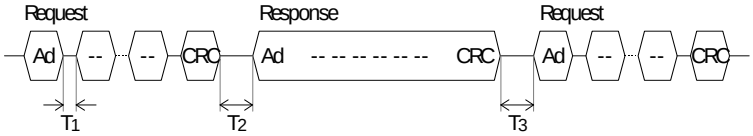
CRC GENERATION *(according to modbus RTU)*

1. Load a 16-bit register with H'FFFF (all '1'). Call it CRC.
2. Exclusive OR the first message byte with the low-order byte of the CRC register and put the result in CRC.
3. Shift the CRC register 1 bit towards the right (to the LSB) writing a '0' in the MSB.
4. If the new LSB is '0' repeat from point 3. If the LSB is '1', exclusive OR the CRC register with the 16-bit value H'A001 (1010 0000 0000 0001).
5. Repeat points 3 and 4 until completing a total of 8 shifts, after what the first byte has been processed.
6. Repeat points 2 to 5 with the second byte and continue these operations until all bytes on the message are processed.
7. Append the obtained CRC to the message so that the low order byte is transmitted first.

CRC Polynome: $2^{15} + 2^{13} + 2^{0}$
CRC Initial value: H'FFF

3. TIMING

The instrument detects the start of a message when receives a valid character (containing its address or the address 00) after a silent interval of at least 3,5 character times.
The end of a message is acknowledged by a silent interval of the same duration.



T1: time between two characters (min 0, max 3,5CT)
T2: time between query and response (min 3,5CT)
T3: time between response and next query (min 3,5CT)

CT = 1 character time

baud (bits/s)	3,5CT
1200	30ms
2400	15ms
4800	8ms
9600	4ms
19200	2ms

4. MODBUS FUNCTIONS

The modbus functions supported by the instrument are the following:

CODE	FUNCTION
03	read n words
05	force state
16	write n words

Function 03 Used to read, in floating point format, dynamic variables such as display value, peak, valley, tare..., depending on model.

Function 05 Used to force the instrument to take an action such as 'make tare', 'reset tare', 'reset peak', 'reset latched setpoints'..., depending on model.

Function 16 Used to write in the instrument memory, programming variable in word format.

5. DATA TYPE AND MEMORY LOCATIONS

The instrument manages different types of data which are accessible to the user by programming, display or through the serial channel RS232C or RS485.
The data is located depending on type, in specific memory areas with unique addresses incrementing in 1 per byte from the zero address.
The figure below shows the user accessible memory areas, with the data types and modbus functions to be used each zone.

PROGRAMMING DATA **IN TABLE 4** *(READ AND WRITE)*	These are the data saved, in binary format in the instrument e2prom memory. The modbus function used to read is 01 and to write is 0F.
RESERVED	
FLOATING POINT VARIABLES **IN TABLE 1** *(READ ONLY)*	These are measurement variables such as display, peak..., in floating point format (IEEE single precision). The modbus function used to read these data is 03.
RESERVED	

CONTROL VARIABLES are not located to specific memory addresses but they consist of internal flags that the instrument interprets to take an action, and then deactivate the flag (see page 12).

5.1. Reading programming variables

FUNCTION 03 *(READ N WORDS))*

Request frame

1 byte	1 byte	2 bytes	2 bytes	2 bytes
slave address	MODBUS function	1st byte address according to table 1	number of words (nº bytes / 2)	CRC

Response frame

1 byte	1 byte	1 byte	n bytes	2 bytes
address	function	nº bytes	information read	CRC

Examples

Request input type of instrument at address 01

H'01	H'03	H'00	H'00	H'00	H'01	H'84	H'0A
Slave address	modbus function	1er byte address according to tables 1		number of words =1		CRC	

Response

H'01	H'03	H'02	H'01	H'00	H'B9	H'D4
address	function	nº bytes	Load cell and Celsius		CRC	

5.2 Write programming variables

Data programming listings in table 4 can be modified writing the wished direction using the function 10.

FUNCTION 10 *(WRITE N WORDS))*

Request frame

1 byte	1 byte	2 bytes	2 bytes	1 byte	n bytes	2 bytes
Slave address	MODBUS function	1st word address according to table 1a and 1b	number of words (nº bytes / 2)	nº of bytes to write	Data to write	CRC

Response frame

1 byte	1 byte	2 bytes	2 bytes	2 bytes
address	function	1st address	number of words written	CRC

Examples

Programming of the brightness and the shipment of date to the printer to the guidance unit 01

H'01	H'10	H'00	H'6D	H'00	H'01	H'02	H'01	H'01
Slave address	function MODBUS	1st byte address (d'109 according to tabla1a)		number of words =1		number of bytes=2	Bright Hi	Print

H'6E	H'BD
CRC	

Response

H'01	H'10	H'00	H'6D	H'00	H'01	H'90	H'14
address	function	1st byte address		number of words =1		CRC	

5.3 Reading dynamic variables

Dynamic variables are those that can vary based on the process without the user has access to modify them directly.
The dynamic variables are normally the values of display, peak, valley.... Their positions in memory specify in the tables 1a and 1b of the annex.
These variables are asked for by means of the function MODBUS like type variables integer of 2 bytes (1 word).

FUNCTION 03 *(WRITE N WORDS))*

Request frame

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Slave address	function MODBUS	1st word address according to the tables 1a and 1b	number of words (nº bytes / 2)	CRC

Response frame

1 byte	1 byte	1 byte	n bytes	2 bytes
address	function	nº bytes	Read Information	CRC

NOTE: In general these variables are transmitted without the decimal point of the display.

Example

Request of the value of display to the instrument unit 01

H'01	H'03	H'00	H'8D	H'00	H'01	H'14	H'21
Slave address	function MODBUS	1st byte address (d'141 as in table)		number of words =1		CRC	

Response (supposing display = +992)

H'01	H'03	H'02	H'03	H'E0	H'B9	H'3C
address	function	nº bytes	data (display value whole format)		CRC	

Request of the values of peak and valley to the guidance unit 01

H'01	H'03	H'00	H'92	H'00	H'02	H'65	H'E6
Slave address	function MODBUS	1st byte address (d'146 in as table1)		number of words =2		CRC	

Response (supposing peak=+1520, valley=-968)

H'01	H'03	H'04	H'5	H'F0	H'FC	H'38
address	function	nº bytes	datas (peak value)		datas (valley value)	

H'BA	H'1E
CRC	

Orders (Control of Variables)

It implies the execution of an order by the instrument. The address of the variable replaces itself by the command indicated in the table 3 of the annex according to the model.

FUNCTION 05 *(TO FORCE STATE)*

Request format

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Slave address	function MODBUS	address word (command)	put bit to '1' (fixed H'FF H'00)	CRC

Response format

1 byte	1 byte	2 bytes	2 bytes	2 bytes
address	function	command	bit to '1' (H'FF H'00)	CRC

6. FORMAT OF ERROR MESAGES

Response Format

1 byte	1 byte	1 byte	2 bytes
Slave address	function +H'80	Code error	CRC

Example

Data Error (code 02) message from slave 01 as response to a function 03.

H'01	H'83	H'02	H'00	H'2C
address	H'03+H'80	code	CRC	

Example:

Request input type of instrument address 01

H'01	H'03	H'00	H'00	H'00	H'01	H'84	H'0B
Slave address	function MODBUS	1st byte address (d'0 According table)		number of words =1		CRC erroneous	

Message Error

H'01	H'83	H'02	H'C0	H'F1
address	H'03+H'80	code	CRC	

7. USING STANDARD COMMUNICATIONS PROGRAMS

Introduction

It is found in the market a wide variety of standard programs that allow collecting data through the serial port of a computer to be used in a virtual control panel on the screen of the PC. The modbus communications protocol is used as a universal tool for connecting these programs with all types of remote instrumentation capable of managing this protocol.

In general, such programs continuously scan, at a fixed rate, the setup addresses according to the type of data inside. The type of the data depends on the modbus function used to request it.
The information is continuously updated in the data bus according to the frame format described in section 2. The program user only has to extract each variable from the frame to be represented on the PC screen in the desired format.
Due to that not always the standard modbus data formats match with those of most instrumentation units, in the case of KOSMOS instruments, the data extraction must be made under the following considerations.

3.2. COLLECTING FLOATING POINT VARIABLES USING FUNCTION 03

In modbus format, the variables by function 03 are represented by words (1 word=2 bytes) and they are contained in addresses that increment in 1 per word, that is, each address contains 2 bytes.
In the instrument, the addresses are incremented in 1 per byte that means that each address contains one byte.
The result of this is that a floating point variable takes 4 addresses in the instrument and 2 addresses in modbus representation.

8. ADRESS OF VARIABLES MEMORY

Programming Data (Reading / Writing)

BYTE	MODBUS	Variable	Explanation
0	0	INPUT	0=Process, 1=Load cell, 2= Temperature
1		UNIDAD	0=Celsius, 1=Fahrenheit
2	1	TYPE PROCESS	0=±10V, 1=±20mA
3		TYPE_TEMP	0=Pt100, 1=Thermocouple
4	2	TYPE_LOAD	0=15mV, 1=30mV, 2=150mV
5		RES	0=1º, 1=0.1º
6	3	TYPE_ THERMOCOUPLE	0=J, 1=K, 2=T
7		INPUT 1	0..9, 10=-1, 11="-"
8	4	[6]	Digit 4
9			Digit 3
10	5		Digit 2
11			Digit 1
12	6		Digit 0 (Digit for maximal internal resolution)
13		INPUT 2	0..9, 10=-1, 11="-"
14	7	[6]	Digit 4
15			Digit 3
16	8		Digit 2
17			Digit 1
18	9		Digit 0 (Digit for maximal internal resolution)
19		INPUT 3	0..9, 10=-1, 11="-"
20	10	[6]	Digit 4
21			Digit 3
22	11		Digit 2
23			Digit 1
24	12		Digit 0 (Digit for maximal internal resolution)

25		INPUT 4 [6]	0..9, 10=-1, 11="-"
26	13		Digit 4
27			Digit 3
28	14		Digit 2
29			Digit 1
30	15		Digit 0 (Digit for maximal internal resolution)
31		INPUT 5 [6]	0..9, 10=-1, 11="-"
32	16		Digit 4
33			Digit 3
34	17		Digit 2
35			Digit 1
36	18		Digit 0 (Digit for maximal internal resolution)
37		INPUT 6 [6]	0..9, 10=-1, 11="-"
38	19		Digit 4
39			Digit 3
40	20		Digit 2
41			Digit 1
42	21		Digit 0 (Digit for maximal internal resolution)
43		INPUT 7 [6]	0..9, 10=-1, 11="-"
44	22		Digit 4
45			Digit 3
46	23		Digit 2
47			Digit 1
48	24		Digit 0 (Digit for maximal internal resolution)
49		INPUT 8 [6]	0..9, 10=-1, 11="-"
50	25		Digit 4
51			Digit 3
52	26		Digit 2
53			Digit 1
54	27		Digit 0 (Digit for maximal internal resolution)

55		INPUT 9	0..9, 10=-1, 11=”-“
56	28	[6]	Digit 4
57			Digit 3
58	29		Digit 2
59			Digit 1
60	30		Digit 0 (Digit for maximal internal resolution)
61		INPUT 10	0..9, 10=-1, 11=”-“
62	31	[6]	Digit 4
63			Digit 3
64	32		Digit 2
65			Digit 1
66	33		Digit 0 (Digit for maximal internal resolution)
67		INPUT 11	0..9, 10=-1, 11=”-“
68	34	[6]	Digit 4
69			Digit 3
70	35		Digit 2
71			Digit 1
72	36		Digit 0 (Digit for maximal internal resolution)
73		DISPLAY 1	0, 1, 10=-1, 11=”-“
74	37	[5]	Digit 3
75			Digit 2
76	38		Digit 1
77			Digit 0
78	39	DISPLAY 2	0, 1, 10=-1, 11=”-“
79		[5]	Digit 3
80	40		Digit 2
81			Digit 1
82	41		Digit 0

83		DISPLAY 3 [5]	0, 1, 10=-1, 11=”-“
84	42		Digit 3
85			Digit 2
86	43		Digit 1
87			Digit 0
88	44	DISPLAY 4 [5]	0, 1, 10=-1, 11=”-“
89			Digit 3
90	45		Digit 2
91			Digit 1
92	46		Digit 0
93		DISPLAY 5 [5]	0, 1, 10=-1, 11=”-“
94	47		Digit 3
95			Digit 2
96	48		Digit 1
97			Digit 0
98	49	DISPLAY 6 [5]	0, 1, 10=-1, 11=”-“
99			Digit 3
100	50		Digit 2
101			Digit 1
102	51		Digit 0
103		DISPLAY 7 [5]	0, 1, 10=-1, 11=”-“
104	52		Digit 3
105			Digit 2
106	53		Digit 1
107			Digit 0
108	54	DISPLAY 8 [5]	0, 1, 10=-1, 11=”-“
109			Digit 3
110	55		Digit 2
111			Digit 1
112	56		Digit 0

113		DISPLAY 9 [5]	0, 1, 10=-1, 11=”-“
114	57		Digit 3
115			Digit 2
116	58		Digit 1
117			Digit 0
118	59	DISPLAY 10 [5]	0, 1, 10=-1, 11=”-“
119			Digit 3
120	60		Digit 2
121			Digit 1
122	61		Digit 0
123		DISPLAY 11 [5]	0, 1, 10=-1, 11=”-“
124	62		Digit 3
125			Digit 2
126	63		Digit 1
127			Digit 0
128	64	Offset [3]	0=”+”, 11=”-“
129			Digit 1
130	65		Digit 0
131		AnaHi [5]	0, 1, 10=-1, 11=”-“
132	66		Digit 3
133			Digit 2
134	67		Digit 1
135			Digit 0
136	68	AnaLo [5]	0, 1, 10=-1, 11=”-“
137			Digit 3
138	69		Digit 2
139			Digit 1
140	70		Digit 0

141		DisplayPointDecimal	0=18888, 1=1888.8, 2=188.88, 3=18.888, 4=1.8888
142	71	Point Decimal Input	0=88888, 1=8888.8, 2=888.88, 3=88.888, 4=8.8888
143		Linealization Points	1 = 2 ptos, ... 10 = 11 ptos
144	72	Order P Filter	0=off, 1..9 on
145		SetpointOnOff	State Setpoint 1 -> 0=off, 1=on
146	73	[4]	State Setpoint 2 -> 0=off, 1=on
147			State Setpoint 3 -> 0=off, 1=on
148	74		State Setpoint 4 -> 0=off, 1=on
149		SetpointHiLo	Mode Setpoint 1 -> 0=Hi, 1=Lo
150	75	[4]	Mode Setpoint 2 -> 0=Hi, 1=Lo
151			Mode Setpoint 3 -> 0=Hi, 1=Lo
152	76		Mode Setpoint 4 -> 0=Hi, 1=Lo
153		SetpointAct	Activation Setpoint 1 -> 0=Hysteresis, 1=Dly
154	77	[4]	Activation Setpoint 1 -> 0=Hysteresis, 1=Dly
155			Activation Setpoint 1 -> 0=Hysteresis, 1=Dly
156	78		Activation Setpoint 1 -> 0=Hysteresis, 1=Dly
157		ValueSet1	0, 1, 10=-1, 11="-"
158	79	[5]	Diigt 3
159			Digit 2
160	80		Digit1
161			Digit 0
162	81	ValueSet 2	0, 1, 10=-1, 11="-"
163		[5]	Digit 3
164	82		Digit 2
165			Digit 1
166	83		Digit 0
167		ValueSet 3	0, 1, 10=-1, 11="-"
168	84	[5]	Digit 3
169			Digit 2
170	85		Digit 1
171			Digit 0

172	86	ValueSet 4 [5]	0, 1, 10=-1, 11=”-“
173			Digit 3
174	87		Digit 2
175			Digit 1
176	88		Digit 0
177		DlyRetSet 1	Digit 4
178	89		Digit 3
179			Digit 2
180	90		Digit 1
181			Digit 0
182	91	DlyRetSet 2	Digit 4
183			Digit 3
184	92		Digit 2
185			Digit 1
186	93		Digit 0
187		DlyRetSet 3	Digit 4
188	94		Digit 3
189			Digit 2
190	95		Digit 1
191			Digit 0
192	96	DlyRetSet 4	Digit 4
193			Digit 3
194	97		Digit 2
195			Digit 1
196	98		Digit 0
197		SetComp [4]	Comp. Setpoint 1 -> 0=Net, 1=Gross
198	99		Comp. Setpoint 2 -> 0=Net, 1=Gross
199			Comp. Setpoint 3 -> 0=Net, 1=Gross
200	100		Comp. Setpoint 4 -> 0=Net, 1=Gross

201		AlarmColor	Alarm Set1 -> 0=No Change, 1=Red, 2=Green, 3=Orange
202	101	[4]	Alarm Set2 -> 0=No Change, 1= Red, 2=Green, 3=Orange
203			Alarm Set3 -> 0=No Change, 1= Red, 2=Green, 3=Orange
204	102		Alarm Set4 -> 0=No Change, 1= Red, 2=Green, 3=Orange
205		Lock 1	bit 7 = - bit 6 = - bit 5 = Scal bit 4 = Input bit 3 = Setpoint 4 bit 2 = Setpoint 3 bit 1= Setpoint 2 bit 0 = Setpoint 1
206	103	Lock 2	bit 7 = - bit 6 = - bit 5 = Analog output bit 4 = Total Lock bit 3 = Tare Key bit 2 = Logical Functions bit 1 = Serial Output bit 0 = Direct access to Setpoint
207		Code	Digit 3
208	104	[4]	Digit 2
209			Digit 1
210	105		Digit 0
211		Logical Functions	Flog 1
212	106	[3]	Flog 2
213			Flog 3
214	107	SetpointFLog12	0=Set 1, 1=Set 2, 2=Set 3, 3=Set 4
215		Round	0=”01”, 1=”05”, 2=”10”
216	108	ColorRun	0=Red, 1=Green, 2=Orange
217		ColorProg	0=Red, 1=Green, 2=Orange

218	109	Bright	0=High, 1=Low
219		Print Date	0=No, 1=yesí
220	110	ModeTare	0=Tara1, 1=Tare2, 2=Tare3
221		Special value Tare	0, 1, 10=-1, 11="-"
222	111		Digit 3
223			Digit 2
224	112		Digit 1
225			Digit 0

Programming (Only Reading)

BYTE	MODBUS	Variable	Explanation
226	113		
227		RSBaudRate	0=1200 baud, 1=2400, 2=4800, 3=9600, 4=19200
228	114	RSAdr	Digit 1
229		[2]	Digit 0
230	115	RSPrt	0=ASCII, 1=ISO1745, 2=MODBUS
231		RSDly	0=30 ms, 1=60 ms, 2=100 ms

Commands

Command	Order
110	Reset Max
111	Reset Min
112	Tare
113	Reset Tare

Dinamic Values (Only Reading)

BYTE	MODBUS	Variable	Explanation
280	140	ValueTare	
281			
282	141	ValueDisplayInteger	
283			
284	142	ValorInteger Set 1	
285			
286	143	ValorInteger Set 2	
287			
288	144	ValorInteger Set 3	
289			
290	145	ValorInteger Set 4	
291			
292	146	Value Pic	
293			
294	147	Value Valley	
295			
296	148	StateReley	Set 1
297		[4]	Set 2
298	149		Set 3
299			Set 4
300	150	Ana High Integer	
301			
302	151	Ana Low Integer	
303			
304	152	ValueDisplayLong	
305			
306	153		
307			

BYTE	MODBUS	Variable	Explanation
308	154	SignoOver	
309		Over	
310	155	Version	
311			

DISEÑOS Y TECNOLOGIA, S.A.
Travessera de Les Corts, 180
08028 BARCELONA - Spain
Tel: +34 - 93 339 47 58
Fax: +34 - 93 490 31 45
E-mail: dtl@ditel.es

www.ditel.es