



# ICON 2000

## ELECTRIC ACTUATOR

# Section 618/7

## Operating and configuration

# ICON 2000\_MOD\_RTU

## *MODBUS RTU*

### *Module*

# *MODBUS RTU*

## *Fieldbus*

File : man618-7\_1.doc      rev. 1

***NOTES:***

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

***REVISION LIST***

1	15-11-01	Instruction for rev. 2 ISP, 2.4 base, MOD 12	A. A.	A. A.
0	30-08-01	Instruction for rev. 2 ISP, 2.3 base, MOD 6	A. A.	A.A.
Rev.	Date	DESCRIPTION	Prepared	Approved

## Table of contents

<b>1. INTRODUCTION .....</b>	<b>1</b>
1.1. MODBUS RTU or ASCII.....	1
<b>2. OPERATION AND STORAGE .....</b>	<b>1</b>
<b>3. COMMUNICATION FEATURES.....</b>	<b>1</b>
<b>4. ICON_2000_MOD_RTU MODULE .....</b>	<b>2</b>
<b>5. MODBUS RTU DESCRIPTION .....</b>	<b>3</b>
<b>6. RS485 TRANSMISSION MODE .....</b>	<b>4</b>
<b>7. MODBUS RTU FUNCTIONS.....</b>	<b>7</b>
<b>8. MODBUS FUNCTION DESCRIPTION .....</b>	<b>8</b>
8.1. Read coil status.....	8
8.2. Read input status .....	9
8.3. Read holding registers .....	10
8.4. Read input registers .....	11
8.5. Force single coil.....	12
8.6. Preset single register.....	13
8.7. Diagnostic .....	14
8.8. Force multiple coil.....	15
8.9. Preset multiple register.....	16
8.10. Report slave ID .....	17
8.11. Exception responses .....	17
<b>9. CONFIGURATION VIA LOCAL OPERATOR INTERFACE OF ICON 2000.....</b>	<b>18</b>
9.1. BUS control .....	18
9.2. Positioner function.....	19
9.3. Fail safe function.....	20

**9.4. Viewing transmission info ..... 21**

**9.5. Bus signal failure indication..... 21**

## 1. Introduction

The **ICON\_2000\_MOD\_RTU** is an electronic module that allows to connect the BIFFI electrical actuator ICON 2000 to a MODBUS RTU serial communication line. The module has its microprocessor, it is controlled by a program stored internally, it works as a pure bus interface and does not affect the actuator control integrity. It is installed inside the actuator housing, and takes the electrical power from the actuator power supply module. The RS 485 interface is located on the module board. The data lines are fully isolated from the actuator electronics.

### 1.1. MODBUS RTU OR ASCII

MODBUS protocol is currently supported by most Programmable Logic Controllers (PLCs). Modbus is a very simple Master / Slave serial protocol. It supports both ASCII (rarely used) and Binary (RTU) form of transmission. In ASCII transmission mode each 8-bit byte in a message is sent as 2 ASCII characters. In RTU transmission mode each 8-bit byte is sent as 2 four-bit hexadecimal characters. The main advantage of RTU mode is that it allows greater character density than in ASCII mode at the same baud rate.

The structure of a MODBUS message is shown below.

Address	Function	Data	Checksum
---------	----------	------	----------

## 2. Operation and storage

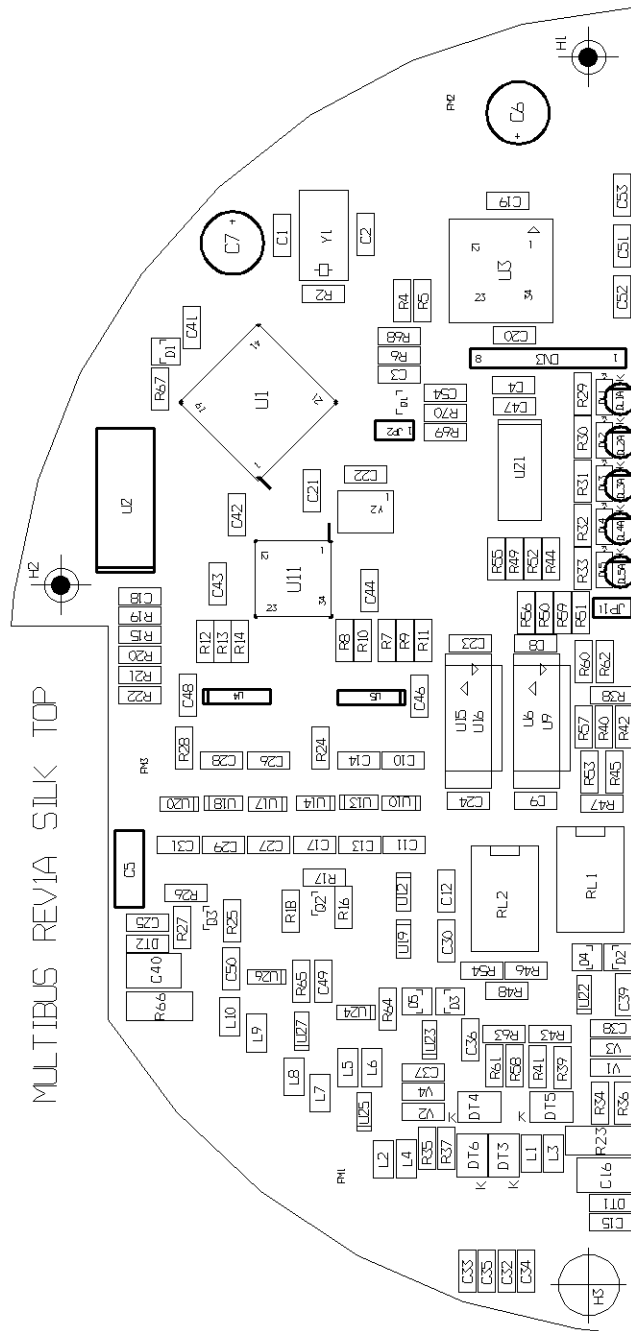
The module is designed to work and to be stored in the same environment of the actuator.

## 3. Communication features

<b>Communication protocol</b>	: MODBUS RTU
<b>Transmission technology</b>	: RS 485, half duplex
<b>Network topology</b>	: Line (bus) structure.
<b>Transmission medium</b>	: Twisted, screened copper cable
<b>Data rate</b>	: 600 1200 2400 4800 9600 19200 38400 <b>bit/sec</b>
<b>Device number</b>	: max. 32 devices per segment. If more than 32 devices are present on the bus, repeaters should be used
<b>Slave address</b>	: from 1 to 247 (address 0 reserved for broadcast messages), configurable via local operator interface of actuator
<b>Bus access</b>	: polling between masters and slaves
<b>Electrical power</b>	: actuator powered
<b>Bus termination</b>	: available on board, configurable via local operator interface of actuator
<b>Temperature</b>	: -40°C, +85°C
<b>Fieldbus redundancy</b>	: available by setting the MODE function (CH1, CH2, AUTO)
<b>Auto_baud rate</b>	: available by setting "baud rate = AUTO"
<b>EMC protections</b>	: EN 50081-2 and EN 50082-2
<b>Baud rate</b>	: configurable via local operator interface of actuator
<b>Addressing</b>	: configurable via local operator interface of actuator
<b>Coding system</b>	: 8 bit binary, hexadecimal 0-9, A-F. Two hexadecimal characters contained in each 8 bit field of the message
<b>Bits per byte</b>	: 1 start bit 8 data bits, least significant bit sent first 1 bit for even / odd parity; no bit for no parity, configurable via local operator interface of the actuator 1 stop bit
<b>Error check field</b>	: cyclical redundancy check (CRC)

## 4. ICON\_2000\_MOD\_RTU module

The module consists in a single PCB that is installed inside the actuator housing. It is connected to the ICON 2000 base card via strip connector. The internal wiring connects the RS 485 data lines to the actuator terminal board.

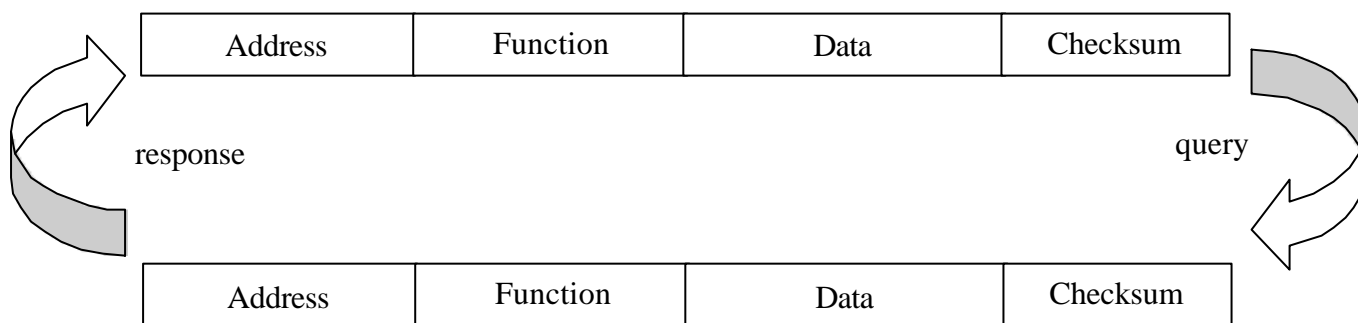


## 5. MODBUS RTU description

MODBUS protocol is a messaging structure, used in a wide range of application in process automation to establish master-slave communication between intelligent devices. Since MODBUS is only a messaging structure, it is independent from the physical layer. It is traditionally implemented using RS 232 or RS 422 or RS 485. The central controllers (as PLC ) communicate via serial connection with field devices (as sensors, actuators). The central controller (called master) reads the input information from the field devices (called slaves) and writes the output information to the slaves. The master initiates the transmission which is called "query". The slave devices answer by sending the requested data (called "response") and performing the action requested in the query. The master can address individual slaves or can send a broadcast message to all slaves.

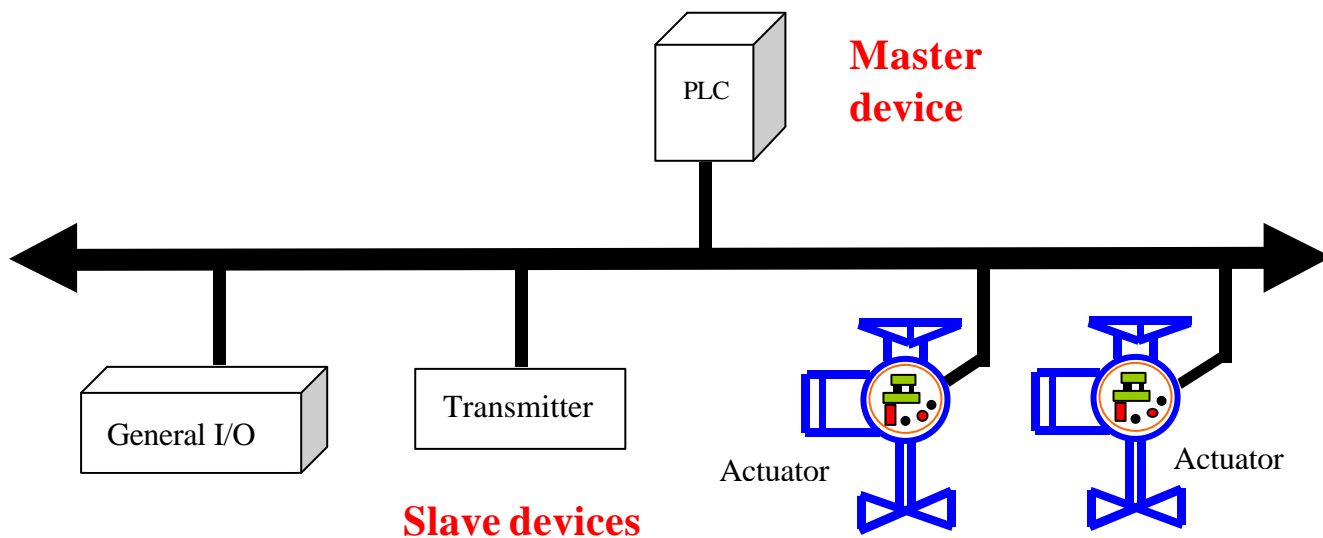
No response from slaves is done in case of broadcast message. Address 0 is used for the broadcast message.

The figure below shows a query-response cycle.



The max. number of master and slave devices in a bus segment is 32 without repeaters. The max. cable length depends on the speed of transmission. Higher is the speed shorter should be the cable length

The figure below shows a MODBUS RTU configuration with 1 master device and different slave devices



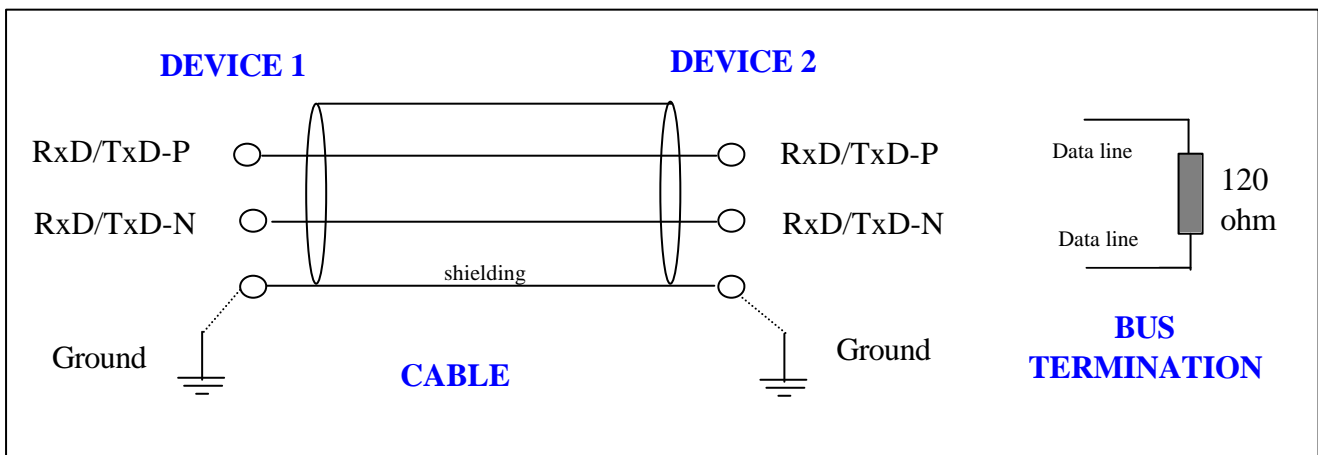
## 6. RS485 transmission mode

The ICON\_2000\_MOD\_RTU module uses a half duplex, multidrop, serial communication line RS485. The module communicates with the masters via RS485 interface and the transmission media consists in a shielded twisted pair cable. Transmission speed from 600 bit/sec to 38400 bit/sec are available. One unique transmission speed is allowed for all devices on the bus when the system works.

All devices are connected in a bus structure. Up to 32 station (master and slaves) can be connected in one segment. Repeaters should be used to extend the number of devices on one bus. Addresses range is from 1 to 247. **Address 0 is reserved for broadcast messages.**

- The bus must be terminated by a resistor of 120 ohm **at the beginning and at the end of each segment**. Only two terminations in one bus segment must be provided. To ensure error-free operation, and to increase driving capability, pull-up and pull-down resistors should be provided on the termination network. The maximum cable length depends on the transmission speed.

**The data lines must not be reversed.** To avoid polarity inversion, it is suggested to use **different colours for each data line**. Use of shielded cable is mandatory for having high system immunity against electromagnetic disturbs. The data lines should be kept separate from all other cables. It should be laid in separate, conductive and earthed cable trunking. It must be ensured that there are not voltage difference between individual nodes of bus.

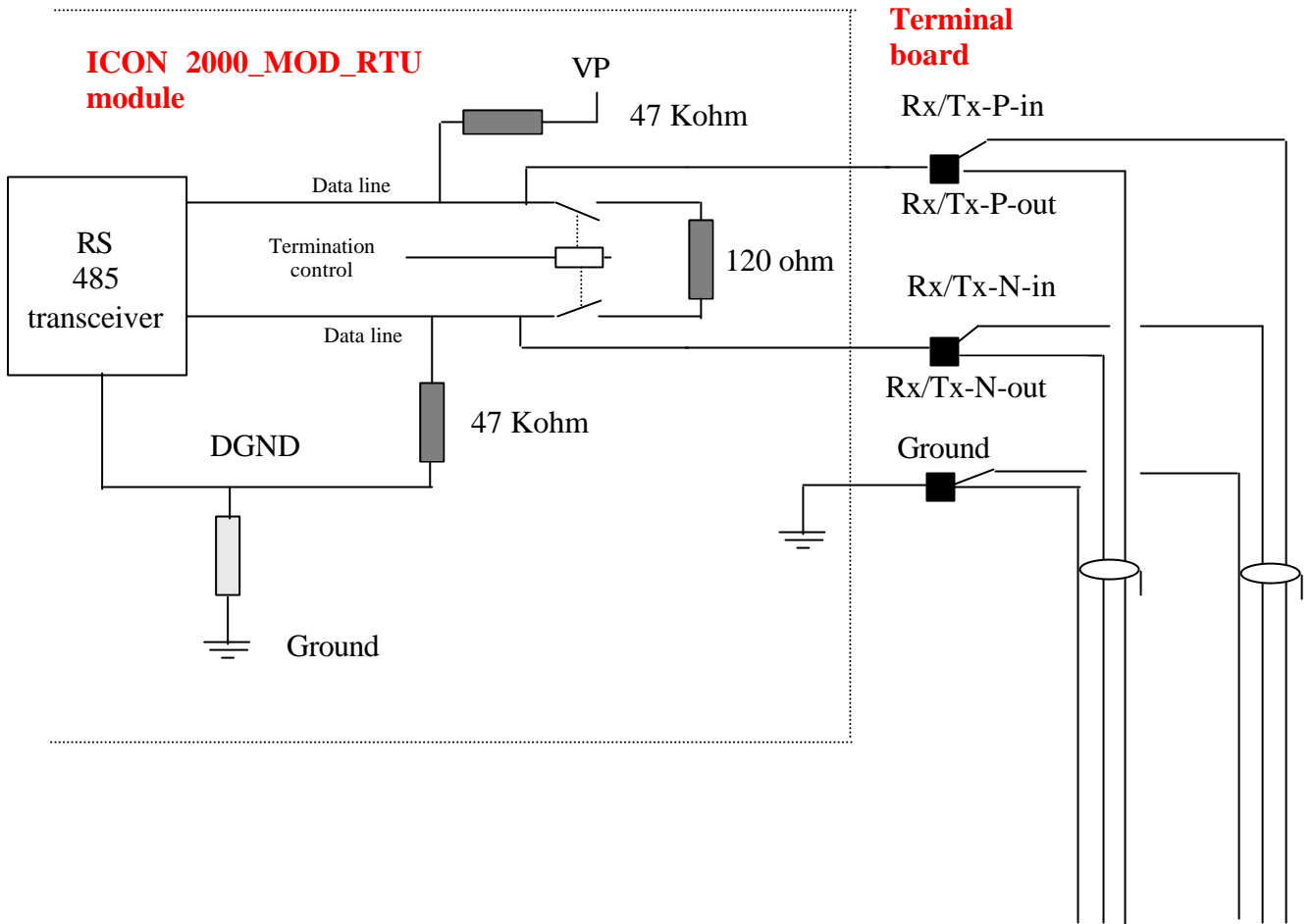


The ICON 2000\_MOD\_RTU module takes its electrical supply from the actuator power supply module. The RS485 bus transceiver is isolated from the actuator electronics. The bus termination, located inside the actuator, should be used only if the actuator is at the beginning or at the end of the bus segment and if there is no external termination.

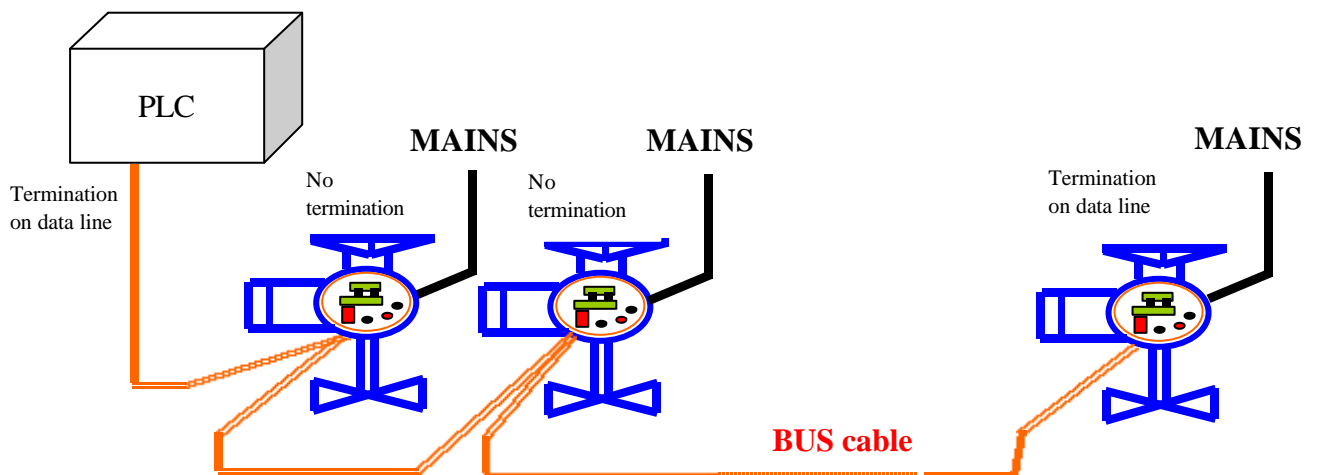
The bus terminations can be switched on the data lines by means of 2 links, configurable via local operator interface.

The figure below shows the wiring in case of **single bus cable**.

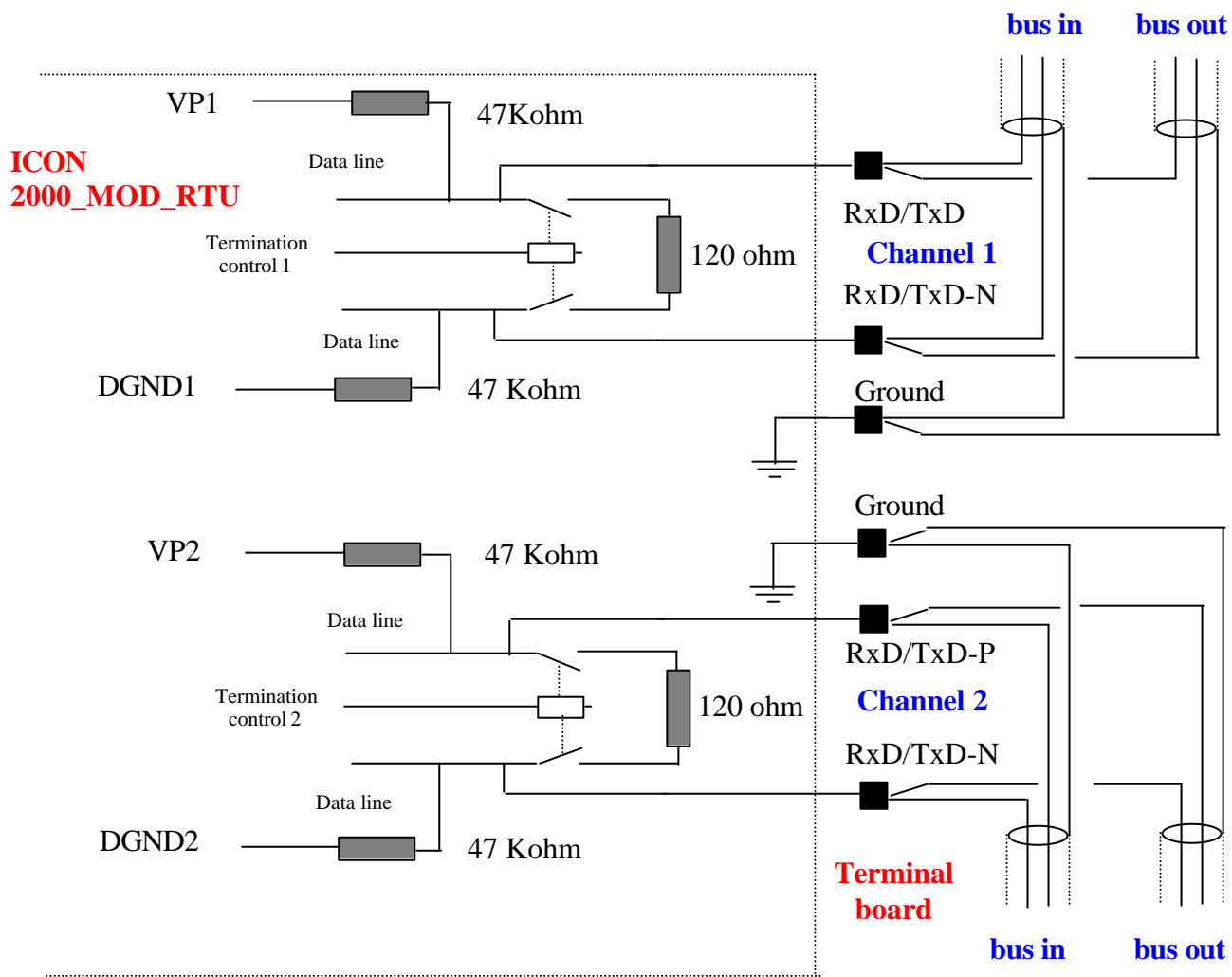




The figure below shows the bus with an actuator at the end of the bus segment.



The ICON 2000\_MOD\_RTU is provided with 2 RS 485 transceivers and 2 bus terminations for application with **redundant cable**. The bus terminations can be switched on the data lines by means of 2 links, configurable via local operator interface. One channel at time works, according to the configuration received from bus or via local operator interface. The figure below shows the wiring necessary in case of **redundant bus cable**. The termination must be linked to the data lines only if the actuator is at the beginning or at the end of the bus segment.



## 7. MODBUS RTU functions

<b>Transmission technology</b>	: RS 485 , 2 wires , half duplex
<b>Data rate</b>	: 600 1200 4800 9600 19200 38400 <b>bit/sec</b>
<b>Slave address</b>	: from 1 to 247 (address 0 reserved for broadcast messages)
<b>Coding system</b>	: 8 bit binary, exadecimal 0-9, A-F. Two exadecimal characters contained in each 8 bit field of the message
<b>Bits per byte</b>	: 1 start bit 8 data bits, least significant bit sent first 1 bit for even / odd parity; no bit for no parity 1 stop bit
<b>Error check field</b>	: cyclical redundancy check (CRC)

The following functions are supported:

Function Code	Modbus Name
01	Read coil status
02	Read input status
03	Read holding registers
04	Read input registers
05	Force single coil
06	Preset single register
08	Diagnostic
15 (0F Hex)	Force multiple coil
16 (10 Hex)	Preset multiple registers
17 (11 Hex)	Report slave ID

## 8. MODBUS function description

The following paragraph describes the input and output messages of ICON 2000\_MOD\_RTU. In all cases it is called **"input signal"** a data flowing from actuator to bus, viceversa it is called **"output signal"** a data flowing from bus to slave.

The information relevant to the MODBUS protocol are based on the MODICON MODBUS PROTOCOL REFERENCE GUIDE PI-MBUS-300 Rev. J June 1996

### 8.1. READ COIL STATUS

- function code 01
- references 0x

The function allows the host (PLC, DCS, etc.) to know the status of discrete outputs called "coils" in the slave. Broadcast is not allowed.

1. Coil n°	function	Description
Bit 0 = 1	: open	open command active
Bit 1 = 1	: close	close command active
Bit 2 = 1	: stop	stop command active
Bit 3 = 1	: ESD on	ESD command active
Bit 4 = 1	: positioner mode on	positioner mode enabled
Bit 5	: reserved	
Bit 6	: reserved	
Bit 7	: reserved	
Bit 8	: reserved	
Bit 9	: reserved	
Bit 10	: reserved	

Example: read status of coils 0 – 3 from device 18 (DEC)

#### Query

Name in the message	Value (HEX)
address	12
code	01
Start addr. Hi	00
Start addr. Lo	00
N° points Hi	00
N° points Lo	04
CRC	-

#### Response

Name in the message	Value (HEX)
address	12
code	01
Byte count	01
Data (coil 7-0)	--
CRC	-

Bits 7-4 will be 0

## 8.2 READ INPUT STATUS

- function code 02
- references 1x

This function allows to read the discrete status of the slave. The following table shows the map of the actuator status. The information associated to bits 8 – 13 can be configured via local operator interface of the actuator. Broadcast is not supported.

N°	function	description
Bit 0 = 1	: close limit	end of stroke in open direction
Bit 1 = 1	: open limit	end of stroke in close direction
Bit 2 = 1	: moving	actuator moving (by action of motor)
Bit 3 = 1	: monitor relay	monitor relay tripped
Bit 4 = 1	: selector in local	3-position selector in LOCAL
Bit 5 = 1	: selector in remote	3-position selector in REMOTE
Bit 6 = 1	: alarm	alarm condition
Bit 7 = 1	: warning	warning condition
Bit 8 = 1	: DIN 1	see note 1
Bit 9 = 1	: DIN 2	see note 1
Bit 10 = 1	: DIN 3	see note 1
Bit 11 = 1	: DIN 4	see note 1
Bit 12 = 1	: DIN 5	see note 1
Bit 13 = 1	: DIN 6	see note 1
Bit 14 = 1	: interlock open	interlock in open direction active
Bit 15 = 1	: interlock close	interlock in close direction active
Bit 16 = 1	: fail safe action	fail safe action active
Bit 17 = 1	: opening	actuator opening (by action of motor)
Bit 18 = 1	: closing	actuator closing (by action of motor)
Bit 19 = 1	: selector in OFF	3-position selector in OFF
Bit 20 = 1	: ESD active	ESD control active
Bit 21 = 1	: HW remote mode	control available via H-wired remote inputs
Bit 22 = 1	: positioner mode	positioner mode enabled
Bit 23 = 1	: motion inhibited	motor operation inhibited (due to 2-speed timer or positioner)
Bit 24 = 1	: channel 1 active	channel 1 active=1, channel 2 active=0
Bit 25 = 1	: aux_in_open	H-wired open active
Bit 26 = 1	: aux_in_close	H-wired close active
Bit 27 = 1	: aux_in_stop	H-wired stop active
Bit 28 = 1	: aux_in_bus-on	H-wired bus-on active
Bit 29 = 1	: act_base_com_fault	reserved (failure on parallel protocol)
Bit 30 = 1	: * maintenance operation	see set/clear maintenance operation

\* available on SW version:  
base >=2.4 , modbus >= 12.0

### Note 1:

via local operator interface of the actuator, the bits DIN 1, ..., DIN 6 can be individually configured to be set to 1 if one of the following condition occurs: open limit, closed limit, position >=xx%, position <=xx%, closing, opening, motor running, blinker, mid-travel position, local selected, remote selected, local stop active, ESD signal on, manual operation, motor over-temperature, high\_high torque, high\_high torque in OP, high\_high torque in CL, valve jammed, valve jammed in OP, valve jammed in CL, low alkaline battery (if present), mid travel alarm in Op or CL, warning.

The following setting is supplied **as standard:**

DIN 1: mid-travel position  
DIN 2: local stop active  
DIN 3: motor over temper. (motor thermostat alarm)  
DIN 4: over-torque (hi\_hi torque alarm)  
DIN 5: valve jammed alarm  
DIN 6: mid-travel alarm in OP/CL

Example: read inputs 10000 – 1013 from device 18 (DEC)

### Query

Name in the message	Value (HEX)
address	12
code	02
Start addr. Hi	00
Start addr. Lo	00
N° points Hi	00
N° points Lo	14
CRC	-

### Response

Name in the message	Value (HEX)
address	12
code	02
Byte count	03
Data (inputs 10007-10000)	--
Data (inputs 10013-10008)	--
CRC	--

### 8.3. READ HOLDING REGISTERS

- function code 03
- references 4x

The function allows to read the contents of the holding (output) registers in the slave. One or many registers can be read. Broadcast is not supported.

Address	Function	Range DEC	
0	actuator commands	0-4	<b>Actuator commands</b> 00 Clear commands 01 close 02 open 03 ESD 04 Positioner mode 05 * Test OP (available with Log Analyser SW) 06 * Test CL (available with Log Analyser SW) 07 * Reserved 08 * Clear recent data log 09 * Set torque reference 10 0A * Set maintenance operation 11 0B * Clear maintenance operation * available on SW version: base >=2.4 , modbus >= 12.0
1	position request	0=0.0% , 1000=100.0%	
2	deadband	0=0.0%, 255=25.5%	
3	motion inhibit time	0-255 sec	
4	reserved		
5	reserved		
6	reserved		
7	safety behavior	0=off, 1=close, 2=open, 3=stay-put, 4=go to xx%	
8	delay before initiating safe action	0=0 sec, 255=255 sec	
9	safe position	0=0%, 100=100%	
10	reserved		
11/12	next maintenance date	4 BCD	
13/14	last maintenance date	4 BCD	
15/16	reserved (date sync)		
17/18	reserved (time sync)		
19/20	start-up date	4 BCD	
21	channel mode	CH1=0, CH2=1, AUTO=2	

**Date format**

dd	mm	20	yy
----	----	----	----

reg.n                  reg.n+1

AUTO: the slave places the response on the same channel of the query

Example: read registers 40000 – 40003 from device 18 (DEC)

#### Query

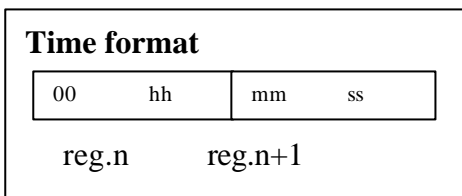
Name in the message	Value (HEX)
address	12
code	03
Start addr. Hi	00
Start addr. Lo	05
N° points Hi	00
N° points Lo	04
CRC	--

#### Response

Name in the message	Value (HEX)
address	12
code	02
Byte count	08
Data Hi (reg. 40000)	--
Data Lo (reg. 40000)	--
Data Hi (reg. 40001)	--
Data Lo (reg. 40001)	--
Data Hi (reg. 40002)	--
Data Lo (reg. 40002)	--
Data Hi (reg. 40003)	--
Data Lo (reg. 40003)	--
CRC	-

### 8.4. READ INPUT REGISTERS

- function code 04
- references 3x



**Warning register**

- Bit 0 = 1 : hi torque in op.
- Bit 1 = 1 : hi torque in cl.
- Bit 2 = 1 : hi temperature
- Bit 3 = 1 : main voltage
- Bit 4 = 1 : contactor cycles
- Bit 5 = 1 : maintenance request
- Bit 6 = 1 : motor current
- Bit 7 = 1 : wrong stroke limit
- Bit 8,...,15 : spare

The function allows to read the contents of the input registers in the slave.

One or many registers can be read. Broadcast is not supported.

Address	Function	Range DEC
0	actuator position	0=0.0%, 1000=100.0
1	output torque	0=0%, 100=100%
2	warning	16 boolean
3	alarms	16 boolean
4	reserved	
5	opening time	0-65535 sec
6	closing time	0-65535 sec
7/8	contactor cycles (resol. 100cycles)	unsigned 4 bytes
9/10	motor run time	unsigned 4bytes (hours)
11/12	time without electrical power	unsigned 4 bytes (hours)
13	reserved (ut.r.%)	0=0%, 100=100%
14/15	test date	4 BCD
16/17	*recent log date	4 BCD
18/19	*recent contactor cycles (resol. 100cycles)	unsigned 4 bytes
20/21	*recent motor run time	unsigned 4bytes (hours)
22/23	*recent time without electrical power	unsigned 4 bytes (hours)
24	*reserved (recent ut.r.%)	0=0%, 100=100%
25/26/27	*torque profile in OP (BRK, PKR, END)	3*integer_16
28/29	*date of last torque profile in OP	4 BCD
30/31/32	*torque profile in CL (BRK, PKR, END)	3*integer_16
33/34	*date of last torque profile in CL	4 BCD
35/36/37	*torque reference in OP (BRK, PKR, END)	3*integer_16
38/39	*date of torque reference in OP	4 BCD
40/41/42	*torque reference in CL (BRK, PKR, END)	3*integer_16
43/44	*date of torque reference in CL	4 BCD
45/46/47/48/49	*alarm code, date, time	unsigned_16+2*4 BCD
50/51/52/53/54	*alarm code, date, time	unsigned_16+2*4 BCD
55/56/57/58/59	*alarm code, date, time	unsigned_16+2*4 BCD
60/61/62/63/64	*alarm code, date, time	unsigned_16+2*4 BCD
65/66/67/68/69	*alarm code, date, time	unsigned_16+2*4 BCD
70/71/72/73/74	*warning code, date, time	unsigned_16+2*4 BCD
75/76/77/78/79	*warning code, date, time	unsigned_16+2*4 BCD
80/81/82/83/84	*warning code, date, time	unsigned_16+2*4 BCD
85/86/87/88/89	*warning code, date, time	unsigned_16+2*4 BCD
90/91/92/93/94	*warning code, date, time	unsigned_16+2*4 BCD
95/96/97/98	*nominal torque/thrust (E.U.+7char)	1 byte (E.U.) + 7 char (0/9/.)

\* available on SW version:  
base >=2.4 , modbus >= 12.0

Output torque: integer 16, closing torque >0

**Alarm register**

- Bit 0 = 1 : motor thermostat
- Bit 1 = 1 : hi\_hi torque in op.
- Bit 2 = 1 : hi\_hi torque in cl.
- Bit 3 = 1 : blocked in op
- Bit 4 = 1 : blocked in cl
- Bit 5 = 1 : hi\_hi temperature
- Bit 6 = 1 : position sensor
- Bit 7 = 1 : speed sensor
- Bit 8 = 1 : main voltage
- Bit 9 = 1 : K1 contactor
- Bit 10=1 : K2 contactor
- Bit 11=1 : configuration error
- Bit 12=1 : HW error
- Bit 13=1 : low battery
- Bit 14=1 : lost phase
- Bit 15=1 : \*request signal

\* available on SW version:  
base >=2.4 , modbus >= 12.0

**Date format**

dd	mm	20	yy
----	----	----	----

reg.n                  reg.n+1

Alarm log code	Warning log code
0 : none	0 : none
1 : configuration error	1 : hi torque in op.
2 : motor thermostat	2 : hi torque in cl
3 : hi_hi temperature	3 : hi temperature
4 : low battery	4 : main supply
5 : hi_hi torque in cl	5 : contactor cycles
6 : hi_hi torque in op.	6 : maintenance request
7 : valve jammed in cl	7 : motor current
8 : valve jammed in op	8 : wrong stroke limits
9 : position sensor	9 : bus fail
10: speed sensor or direction	
11: main voltage	
12: lost phase	
13: K1 contactor	
14: K2 contactor	
15: mid-travel alarm in cl	
16: mid-travel alarm in op	
17: HW error	
18: request signal	

**Torque E.U.** : 0=lb, 2=Nm  
**Thrust E.U.** : 2=lb, 3=KN  
**ASCII tab:** 48=0, 49=1, 50=2, 51=3, 52=4, 53=5, 54=6, 55=7, 56=8, 57=9, 46="."

Example: read registers 30000 – 30001 from device 18 (DEC)

**Query**

Name in the message	Value (HEX)
address	12
code	03
Start addr. Hi	00
Start addr. Lo	00
N° points Hi	00
N° points Lo	02
CRC	--

**Response**

Name in the message	Value (HEX)
address	12
code	02
Byte count	08
Data Hi (reg. 30000)	--
Data Lo (reg. 30000)	--
Data Hi (reg. 30001)	--
Data Lo (reg. 30001)	--
CRC	-

## 8.5. FORCE SINGLE COIL

- function code 05
- references 0x

This function forces a single coil to ON or OFF. Broadcast is supported.

By this function the actuator receives the commands to open, close, stop , or ESD. Bit 4 is used to enable positioner mode. It is available only in **modulating actuators**. The commands "open, close, ESD and enable positioner" are **self-maintained**. Open and Close can be cleared only by the MODBUS command "Stop". Open and Close commands will be ignored until the Stop command is at 1. The local STOP pressed or the selector in OFF or LOCAL do not reset the commands to actuator. When the local STOP is depressed or the selector returns in REMOTE, the previous command is restored.

Coil n° function	Description
Bit 0 : open	Open : on transition from 0 to 1 of the signal an open command to the actuator is generated. If the close command is received the movement is reversed
Bit 1 : close	Close : on transition from 0 to 1 of the signal a close command to the actuator is generated. If the open command is received the movement is reversed.
Bit 2 : stop	Stop : if the signal is 1 , a stop command to the actuator is generated. The open and close command will be cleared. When the signal returns to 0, the commands Open or Close can be carried out.
Bit 3 : ESD	ESD : irf the signal is 1, an ESD command to the actuator is generated. When the signal returns to 0, the ESD control is cleared
Bit 4 : enable positioner	Enable positioner : if the signal is 1 , the positioner function is enabled. The actuator will position according to content of position request register. The open, close, stop commands will be ignored. When the signal returns to 0 the positioner function is disabled
Bit 5 : reserved	
Bit 6 : reserved	
Bit 7 : reserved	
Bit 8 : reserved	
Bit 9 : reserved	
Bit 10 : reserved	

Example: force coil 3 (set ESD control) of device 18 (DEC) .

The value FF 00 force the coil to ON. The value 00 00 force the coil to OFF

The response is the echo of the query

### Query

Name in the message	Value (HEX)
address	12
code	05
Coil addr. Hi	00
Coil addr. Lo	03
Data Hi	FF
Data Lo	00
CRC	-

### Response

Name in the message	Value (HEX)
address	12
code	05
Coil addr. Hi	00
Coil addr. Lo	03
Data Hi	FF
Data Lo	00
CRC	-



## 8.6. PRESET SINGLE REGISTER

- function code 06
- references 4x

This function sets a value in a holding register of the slave. Broadcast is supported.

The content of register 0 is used to generate the commands “open, close, stop, ESD, enable positioner”.

The command is active until a new value of register 0 is received from host. The commands “open and close” are also cleared in the following situations:

- local STOP pressed
- local selector in OFF or LOCAL
- ESD command active
- remote hardwired controls active
- alarm
- end of travel

The command “enable positioner” and the registers 1,...,3 are available only in the modulating actuators.

The content of register 1 is used to update the **position request signal**.

### Actuator commands

00	Clear commands
01	close
02	open
03	ESD
04	Positioner mode
05	* Test OP (available with Data Log Analyser SW pkg)
06	* Test CL (available with Data Log Analyser SW pkg)
07	* Reserved
08	* Clear recent data log
09	* Set torque reference
0A	* Set maintenance operation
0B	* Clear maintenance operation

\* available on SW version:  
base >=2.4 , modbus >= 12.0

Address	Function	Range DEC
0	actuator commands	0-4
1	position request	0=0.0% , 1000=100.0%
2	deadband	0=0.0% , 255=25.5%
3	motion inhibit time	0-255 sec
4	reserved	
5	reserved	
6	reserved	
7	safety behaviour	0=off, 1=close, 2=open, 3=stay-put, 4=go to xx%
8	delay before initiating safe action	0=0 sec, 255=255 sec
9	safe position	0=0% , 100=100%
10	reserved	
11/12	reserved	
13/14	reserved	
15/16	reserved	
17/18	reserved	
19/20	reserved	
21	channel mode	CH1=0, CH2=1, AUTO=2

AUTO: the slave place the response on the same channel of the query

### Clear:

when a code 00 is received, open/close/ESD commands are cleared, and positioner function is disabled

### Close:

when a 01 code is received, a close command to the actuator is generated.

### Open:

when a 02 code is received, an open command to the actuator is generated.

### ESD:

when a 03 code is received, an ESD command to the actuator is generated.

### Enable Positioner:

it enables positioner function. The actuator will position according to content of position demand register. The open and close commands will be ignored.

### Test OP:

Record position / torque in OP. It requires “data log analyser” SW package for PC.

### Test CL:

Record position / torque in CL. It requires “data log analyser” SW package for PC.

### Clear recent data log:

it clears the registers of the “recent data log”

### Set torque reference:

it sets the torque reference registers

### Set maintenance operation:

it inhibits fail safe function and warning detection due to bus failure and enables commands by bus

### Clear maintenance operation:

it enables fail safe function and warning detection due to bus failure

Example: preset register 40001 (set position request to 50.0% = 500 DEC) of device 18 (DEC) .

The response is the echo of the query

### Query

Name in the message	Value (HEX)
address	12
code	06
Register addr. Hi	00
Register addr. Lo	01
Data Hi	01
Data Lo	F4
CRC	-

### Response

Name in the message	Value (HEX)
address	12
code	06
Register addr. Hi	00
Register addr. Lo	01
Data Hi	01
Data Lo	F4
CRC	-

## 8.7. DIAGNOSTIC

- **function code 08**

The function provides a certain number of test to check the communication between master and slave and the presence of error conditions . Broadcast is not supported. The function is divided in sub-functions to select the type of test to be done.

Sub-function Code	Modbus Name	Description
00	Return query data	The data passed in the query data field should be returned in the response
11	Return bus message count	The slave should return the number of messages from restart, clear counter or power-up
12	Return bus communication error count	The slave should return the number of CRC error from restart, clear counter or power-up
13	Return bus exception error count	The slave should return the number of exception responses from restart, clear counter or power-up
14	Return slave message count	The number of messages to the slave from restart, clear counter or power-up should be returned
17	Return slave busy count	The number of slave busy messages from restart, clear counter or power-up should be returned
18	Return bus character overrun count	The number of overrun messages from restart, clear counter or power-up should be returned. An overrun condition is due to data coming at the port slave faster than they can be stored. or by the loss of a

Example: diagnostic “return query data” to slave device 18 (DEC) .The data to be returned is A9 F4

### QUERY

Name in the message	Value (HEX)
address	12
code	08
Sub-function Hi	00
Sub-function Lo	00
Data Hi	A9
Data Lo	F4
CRC	-

### RESPONSE

Name in the message	Value (HEX)
address	12
code	08
Sub-function Hi	00
Sub-function Lo	00
Data Hi	A9
Data Lo	F4
CRC	-

## 8.8. FORCE MULTIPLE COIL

- **function code 15 (DEC)**
- **references 0x**

This function allows to force the coils indicated in the query to ON or OFF state. Broadcast is supported

See function code 05.

Coil n° function	Description
Bit 0 : open	Open : on transition from 0 to 1 of the signal an open command to the actuator is generated. If the close command is received the movement is reversed
Bit 1 : close	Close : on transition from 0 to 1 of the signal a close command to the actuator is generated. If the close command is received the movement is reversed
Bit 2 : stop	Stop : if the signal is 1 , a stop command to the actuator is generated. The open and close command will be cleared. When the signal returns to 0, the commands Open or Close can be carried out.
Bit 3 : ESD	ESD : irf the signal is 1, an ESD command to the actuator is generated. When the signal returns to 0, the ESD control is cleared
Bit 4 : enable positioner	Enable positioner : if the signal is 1 , the positioner function is enabled. The actuator will position according to content of position request register. The open, close, stop commands will be ignored. When the signal returns to 0 the positioner function is disabled
Bit 5 : reserved	
Bit 6 : reserved	
Bit 7 : reserved	
Bit 8 : reserved	
Bit 9 : reserved	
Bit 10 : reserved	

Example: force coils 0-6 of device 18 (DEC) to OFF .

A logical "1" causes the coil to be ON, vice-versa a logical "0" causes the coil to be OFF

**Query**

**Response**

Name in the message	Value (HEX)
address	12
code	0F
Coil addr. Hi	00
Coil addr. Lo	00
Quantity of coils Hi	00
Quantity of coils Lo	07
Byte count	01
Data (bit 7-0)	00
CRC	-

Name in the message	Value (HEX)
address	12
code	0F
Coil addr. Hi	00
Coil addr. Lo	00
Quantity of coils Hi	00
Quantity of coils Lo	07
CRC	-

## 8.9. PRESET MULTIPLE REGISTER

- function code 16 (DEC)
- references 4x

This function allows to update the content of the registers indicated in the query. Broadcast is supported. See function code "06".

Address	Function	Range DEC
0	actuator commands	0-4
1	position request	0=0.0% , 1000=100.0
2	deadband	0=0.0% , 255=25.5%
3	motion inhibit time	0-255 sec
4	reserved	
5	reserved	
6	reserved	
7	safety behaviour	0=off, 1=close, 2=open, 3=stay-put, 4=go to xx%
8	delay before initiating safe action	0=0 sec, 255=255 sec
9	safe position	0=0%, 100=100%
10	reserved	
11/12	next maintenance date	4 BCD
13/14	last maintenance date	4 BCD
15/16	date_sync	4 BCD
17/18	time_sync	4BCD
19/20	start-up date	4BCD
21	channel mode	CH1=0, CH2=1, AUTO=2

AUTO: the slave places the response on the same channel of the query  
Date / Time sync : to synchronise the present time and date

### Actuator commands

00	Clear commands
01	close
02	open
03	ESD
04	Positioner mode
05	* Test OP (available with Data Log Analyser SW pkg)
06	* Test CL (available with Data Log Analyser SW pkg)
07	* Reserved
08	* Clear recent data log
09	* Set torque reference
0A	* Set maintenance operation
0B	* Clear maintenance operation
* available with SW version: base >=2.4 , modbus >= 12.0	

#### Clear:

when a code 00 is received, open/close/ESD commands are cleared, and positioner function is disabled

#### Close:

when a 01 code is received, a close command to the actuator is generated.

#### Open:

when a 02 code is received, an open command to the actuator is generated.

#### ESD:

when a 03 code is received, an ESD command to the actuator is generated.

#### Enable Positioner:

it enables positioner function. The actuator will position according to content of position demand register. The open and close commands will be ignored.

#### Test OP:

Record position / torque in OP. It requires "data log analyser" SW package for PC.

#### Test CL:

Record position / torque in CL. It requires "data log analyser" SW package for PC.

#### Clear recent data log:

it clears the registers of the "recent data log"

#### Set torque reference:

it sets the torque reference registers

#### Set maintenance operation:

it inhibits fail safe function and warning detection due to bus failure and enables commands y bus

#### Clear maintenance operation:

it enables fail safe function and warning detection due to bus failure

### Date format

dd	mm	20	yy
----	----	----	----

reg.n      reg.n+1

### Time format

00	hh	mm	ss
----	----	----	----

reg.n      reg.n+1

Example: preset registers 40000 to 04 and 40001 (set position demand) to 50.0% = 500 DEC, of device 18 (DEC) .

Query

Name in the message	Value (HEX)
address	12
code	10
Register addr. Hi	00
Register addr. Lo	00
N° of registers Hi	00
N° of registers Lo	02
Byte count	04
Data Hi	00
Data Lo	04
Data Hi	01
Data Lo	F4
CRC	-

Response

Name in the message	Value (HEX)
address	12
code	10
Register addr. Hi	00
Register addr. Lo	00
N° of registers Hi	00
N° of registers Lo	02
CRC	-

### 8.10. REPORT SLAVE ID

- **function code 17 (DEC)**

This function returns the information relevant to the type of slave. Actuator manufacturer, actuator serial number, valve tag name and SW version are reported. 81 bytes (for ICON 2000) and 82 bytes (for F01 2000) bytes corresponding to 81 or 82 ASCII characters are retransmitted. Broadcast is not supported.

Bytes 1, ... , 15 : BIFFI ICON 2000	Bytes 1, ... , 16 : BIFFI F 0 1 2000
Bytes 16, ... , 43 : actuator serial number	Bytes 17, ... , 44 : actuator serial number
Bytes 44,... , 71 : valve tag name	Bytes 45,... , 72 : valve tag name
Bytes 72, ... , 81 : SW version	Bytes 73, ... , 82 : SW version
SW : 4 char (SW version of base module) , 2 blank, then 4 char (SW version of bus interface)	

Example: request of report from device 18 (DEC) .

**Query**

Name in the message	Value (HEX)
address	12
code	11
CRC	-

**Response**

Name in the message	Value (HEX)
address	12
code	11
Byte count	52
Slave ID	00
Run indicator status	00
15 bytes (char)	Identifier
28 bytes (ASCII char)	Actuat. Serial number
28 bytes (ASCII char)	Valve tag name
10 bytes (ASCII char)	SW version
CRC	-

### 8.11. EXCEPTION RESPONSES

When a master sends a query to a slave, it expects a response. The following case may occur:

- The slave receive the query and returns the normal response
- The slave does not receive the query and no response is returned. The master wait for a certain time (timeout) and then restarts with new queries
- The slave receive the query but detects a communication error (CRC). No response is returned . The master wait for a certain time (timeout) and then restarts with new queries
- The slave receive the query without any communication error, but it cannot process the data. The slave returns an exception response with the following codes

Code	Name	Description
01	Illegal function	The function code is not allowable for the slave
02	Illegal data address	The data address is not allowable for the slave
03	Illegal data value	A value contained in the data field is not allowable for the slave
04	Slave device failure	An error occurred while the slave was performing the action requested by the query
05	Acknowledge	The slave has received the query but it takes along time to be done. This response allows the master to increase the timeout time
06	Slave device busy	The slave is busy. The master should retransmit the message

## 9. Configuration via local operator interface of ICON 2000

The Modbus RTU interface "MOD\_RTU\_MOD" is an additional module that allows to connect the ICON 2000 to a Modbus RTU fieldbus. The module can be used with either the **base version** or the **optional modules AOC, APTM/APTM1, PSM/PSM1** of ICON 2000. Here below are described the facilities available by the **view and setup menu** of ICON 2000.

### 9.1. BUS CONTROL

- **DIN 1, ..., DIN 6**: by this routine it is possible to choose the condition associated to bits 8, ..., 13 of function code 02 (read input status). Here below there is the list of the available conditions:

STATUS	ALARM
<ul style="list-style-type: none"> <li>• open limit</li> <li>• closed limit</li> <li>• position &gt;= xx %</li> <li>• position &lt;= xx %</li> <li>• closing</li> <li>• opening</li> <li>• motor running</li> <li>• blinker</li> </ul>	<ul style="list-style-type: none"> <li>• mid-travel position</li> <li>• local selected</li> <li>• remote selected</li> <li>• local stop active</li> <li>• ESD signal on</li> <li>• manual operation</li> </ul>
<ul style="list-style-type: none"> <li>• motor over-temperature</li> <li>• over-torque</li> <li>• over-torque in OP</li> <li>• over-torque in CL</li> <li>• valve jammed</li> <li>• warnings</li> </ul>	<ul style="list-style-type: none"> <li>• valve jammed in OP</li> <li>• valve jammed in CL</li> <li>• low alkaline battery (if present)</li> <li>• mid travel alarm in CL/OP</li> </ul>

The following setting is supplied as standard:

- DIN 1: mid-travel position
  - DIN 2: local stop active
  - DIN 3: motor over-temperature (motor thermostat alarm)
  - DIN 4: over-torque (hi\_hi torque alarm)
  - DIN 5: valve jammed alarm
  - DIN 6: mid-travel alarm in OP/CL
- 
- **Baud rate**: by this function it is possible to set the transmission speed. All devices on the same data line must have the same transmission speed.
  - **Parity**: this routine allows to choose the parity in the MODBUS RTU message (ODD, EVEN, NO PARITY)
  - **Node**: by this function it is possible to enter the node address. Each device must have its address. Each address must be associated to one only device. The available address range is from **1 to 247**.
  - **Termin 1** status: by this routine the internal termination of channel 1 can be connected to the bus line (ON / OFF). Set "**termin 1 = ON**" only if the actuator is at the begin or at the end of the bus line and if channel 1 is selected.
  - **Termin 2** status: by this routine the internal termination of channel 2 can be connected to the bus line (ON / OFF). Set "**termin 2 = OFF**" if **single bus line** is used.
  - **Mode** : by this routine the channel 1 or channel 2 are enabled to work. The option is used only when **redundant cable** is requested. Set "**mode = CH1**" if **single bus line** is used.

#### Configuration procedure:

- *Move the local selector to OFF and then press simultaneously OPEN and STOP. Select the language and then enter the password according to the instructions "entering the set-up mode". When the message of display is "SET-UP MODE OK?" press YES. Press YES to select actuator*

set-up menu, press **NO** to scroll the list of available routines and then press **YES** to select **BUS CONTROL**.

- Press **YES** if the condition linked to **DIN 1** is correct, or press **NO** to change, then press **YES**.
- Repeat the previous step for **DIN 2, DIN 3, ..., DIN 6**
- Press **YES** if the configured value of the **BAUD RATE** is correct, or press **NO** to change, then press **YES**.
- Press **YES** if the configured parity (**PARITY**) is correct (**ODD, EVEN, NONE**), or press **NO** to change, then press **YES**
- Press **YES** if the configured value of the node address (**NODE**) is correct (from 1 to 247), or press **NO** to change, then press **YES**.
- Press **YES** if the configured status of termination 1 (**TERMIN 1**) is correct (**ON / OFF**), or press **NO** to change, then press **YES**.
- Press **YES** if the configured status of termination 2 (**TERMIN 2**) is correct (**ON / OFF**), or press **NO** to change, then press **YES**.
- Press **YES** if the configured **MODE** is correct (**CH1, CH2, AUTO**), or press **NO** to change, then press **YES**.

**View procedure:**

- Move the local selector to **OFF** and then press simultaneously **OPEN** and **STOP**. Select the language and then enter the password according to the instructions "entering the view mode". When the message of display is "**VIEW MODE OK?**" press **YES**. Press **YES** to select actuator set-up menu, press **NO** to scroll the list of available routines and then press **YES** to select **BUS CONTROL**.
- Press **YES** to scroll the list of **BUS CONTROL** parameters.

**9.2. POSITIONER FUNCTION**

The function is available only on the **modulating actuators**. The value 0 of position request, received from bus, corresponds to close request and the value 1000 corresponds to open request. The **ICON 2000** compares the **present position %** of the actuator with the **position request % received from the bus**, and if the difference is greater than the **dead band**, the actuator is driven to reach the new requested position.

The following options can be configured via either bus or local operator interface:

- dead band: configurable from 0.0% to 25.5% of the maximum position error (difference among position request % and present position %). The configured value should be great enough to avoid "hunting" effect of the actuator.
- Motion inhibit time: it allows to adjust the length of the delay time between two cycles of the motor. It can be configured from 0 to 255 sec and allows to set the maximum number of start / hour of electrical motor.

**Configuration procedure:**

- Move the local selector to **OFF** and then press simultaneously **OPEN** and **STOP**. Select the language and then enter the password according to the instructions "entering the set-up mode". When the message of display is "**SET-UP MODE OK?**" press **YES**. Press **YES** to select actuator set-up menu, press **NO** to scroll the list of available routines and then press **YES** to select **POSITIONER**.
- Press **YES** if the configured value of the **Dead Band** is correct (from 0 to 25.5% of position error), or press **NO** to change, then press **YES**.
- Press **YES** if the configured value of the **Motion Inhibit Time** is correct (from 0 to 255 sec), or press **NO** to change, then press **YES**.

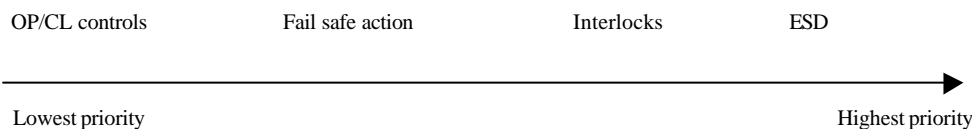
**View procedure:**

- Move the local selector to OFF and then press simultaneously OPEN and STOP. Select the language and then enter the password according to the instructions "entering the view mode". When the message of display is "VIEW MODE OK?" press YES. Press YES to select actuator set-up menu, press NO to scroll the list of available routines and then press YES to select **the routine (POSITIONER)**.
- Press YES to scroll the list of parameters.

**9.3. FAIL SAFE FUNCTION**

**This function is available only if requested on order.** It allows to configure the action of the actuator in case of loss of the bus signal. The action takes place only if the local selector is in REMOTE and if bus is operating. When the bus signal restores, also the actuator restores at its normal functioning. The fail safe function can be configured via either bus or local operator interface.

The hard-wired controls ESD and INTERLOCKS override the Fail Safe action according to the following diagram (the hard-wired controls INTERLOCKS are available only if optional modules APTM/APTM1 or PSM/PSM1 are present).



The following options can be configured:

- Fail safe action: open, close, stay-put, go to position %, no action (OFF)
- Length of the delay time before than the fail safe action takes place (length = 10sec + configured value)

**Configuration procedure:**

- Move the local selector to OFF and then press simultaneously OPEN and STOP. Select the language and then enter the password according to the instructions "entering the set-up mode". When the message of display is "SET-UP MODE OK?" press YES. Press YES to select actuator set-up menu, press NO to scroll the list of available routines and then press YES to select **FAIL SAFE**.
- Press YES if the configured ACTION is correct (open, close, stay-put, go to position xx% , off), or press NO to change, then press YES.
- Press YES if the configured value of the DELAY is correct (from 0 to 255 sec), or press NO to change, then press YES.

**View procedure:**

- Move the local selector to OFF and then press simultaneously OPEN and STOP. Select the language and then enter the password according to the instructions "entering the view mode". When the message of display is "VIEW MODE OK?" press YES. Press YES to select actuator set-up menu, press NO to scroll the list of available routines and then press YES to select **the routine(FAIL SAFE)**.
- Press YES to scroll the list of parameters.



### 9.4. VIEWING TRANSMISSION INFO

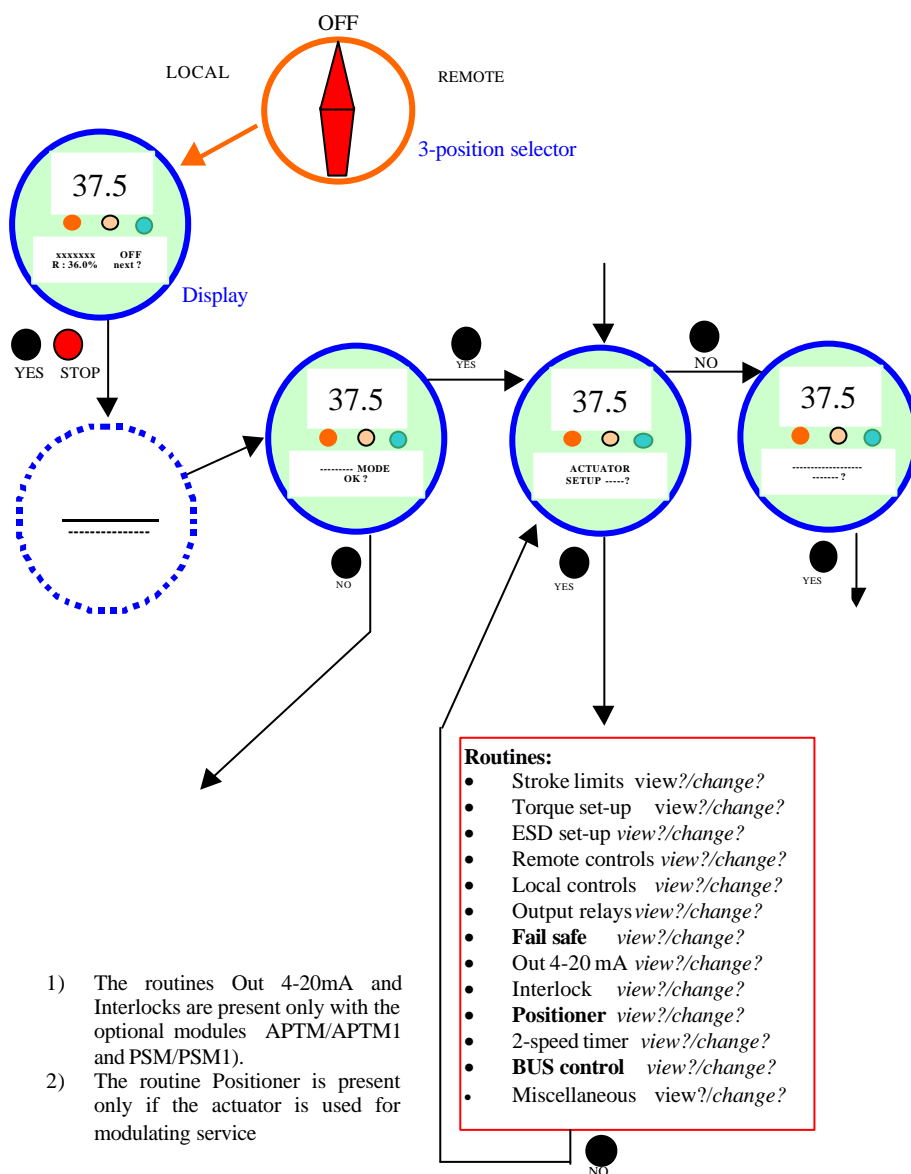
The following procedure allows to see the most significant info relevant to the bus data transmission:

- Move the local selector to OFF or REMOTE and then press YES until the display shows NODE REPORT. Press NO to exit or press YES to scroll the list of transmission info

### 9.5. BUS SIGNAL FAILURE INDICATION

In case of loss of bus signal a warning is generated. It is signalled by the flashing of the relevant ALARM/WARNING LED and by indication on the local 2 lines /16 char. display .

The figure below shows the list of routines available in the ICON 2000 **view or setup** menu.



**Note:** On order, the ICON 2000 can be configured to inhibit “bus signal failure” indication.

