MODBUS-RTU for Humidifier

MODBUS-RTU communication protocol specifications for control of Humidifier Devices in network

Document name: **MODBUS-RTU_UMIDMS03_01-13_EN**
Software installed: **UMIDMS03.hex Rev. 12 and subsequent**

READ AND KEEP

INDEX

GENERAL DESCRIPTION — 1

DESCRIPTION OF COMMANDS — 2

DESCRIPTION OF REGISTERS AND ADDRESSES — 3

GLOSSARY — 4

1: GENERAL DESCRIPTION

1.1 MODBUS PROTOCOL

The data communication system based on Modbus protocol makes it possible to connect up to 247 instruments in a common RS 485 line with standardised communication mode and format.
Communication is implemented in half duplex by means of frames (in continuous transmission). Only the master (PC, PLC ...) can begin question/answer dialogue with the slaves (only one addressed slaves) and the interrogated slave responds. The slave answers after a minimum pause of 3.5 characters between the frame received and that which must be transmitted.
There is also the broadcast communication mode where the master sends a message to all the slaves simultaneously, which do not provide a return answer; however this latter mode cannot be used with this control.
The serial data transmission mode implemented on the control is the RTU type (Remote Terminal Unit), where data is exchanged in binary format (characters of 8 bit).

1.2 SERIAL CONFIGURATION

Serial line:	***RS485***
Baud rate:	***9600***
Data length:	***8 bit***
Parity:	***none***
Stop bit:	***1***

Serial transmission of characters in RTU format

Start	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Stop

1.3 FORMAT OF MESSAGES (FRAME)

According to the MODBUS-RTU standard, each message (Frame) consists in the following parts:

Start	Device address	Function code	Data	CRC16		Stop
4.5msec silence	Byte	Byte	n x Byte	LSByte	MSByte	4.5msec silence

- **Start / Stop:**
  The message begins with a silence of 4.5ms (time 3.5 x greater then time for transmission of a character). See chap. 1.4 for further information.

- **Device address**:
  The address of the device with which the master has established dialogue; it is a value between 1 and 247. The address 0 is reserved for the broadcast, message sent to all slave devices (not active on this control). The RS485 line makes it possible to connect together up to 32 devices (1 Master + 31 slaves), but with specific "bridges" or repeater devices it is possible to exploit the entire logic addressing field.

- **Function code:**
  The function code to be executed or which has been executed; codes 0x03 (register reading), 0x06 (single register reading) and 0x2B/0x0E (identification data reading) are active in the device.

- **Data:**
  Data which must be exchanged.

- **CRC16:**
  The format error control field according to the algorithm CRC16. CRC16 is calculated on the whole message by the transmitting master device and is attached to the message itself. At the end of reception, the slave calculates the CRC16 on the message and compares it with the value attached by the master; if the two values do not correspond, the message will be considered invalid and it will be discarded without sending the master any response.
  The following fragment of code C illustrates the calculation mode of CRC16:

```
unsigned int CRC16
void Modbus_CRC(unsigned char *Frame, unsigned char FrameLength)
{
unsigned char ByteCount;
unsigned char i;
unsigned char bit_lsb;
CRC16 = 0xFFFF;
for (ByteCount=0;ByteCount<FrameLength;ByteCount++)
  {
  CRC16^=Frame[ByteCount];
  for (i=0;i<8,i++)
    {
    bit_lsb = CRC16 & 0x0001;
    CRC16 = CRC16>>1;
    if (bit_lsb == 1)
      CRC16 ^= 0xA001;
    }
  }
}
```

1.4 SYNCHRONISATION OF MESSAGES

Message synchronisation between transmitter and receiver is achieved by interposing a pause of at least 4 ms between messages. If the receiver receives no Byte for 4 ms, it deems the previous message completed and considers the subsequent Byte received as the first of a new message.
The slave, once it has received the complete message, decodes it and, if there are no errors, sends the response message to the master. In order to send the response, the slave engages the RS485 line, waits for a 4.5 ms pause, sends the complete message, waits for a 4.5 ms pause and then frees the RS485 line. The master unit must respect these times, in order to avoid the risk of overlapping transmissions. It is especially necessary to envision an adequate reception time-out for the response before beginning a new transmission (typical time-out value: 500msec or more).

1.5 ERROR MESSAGES (EXCEPTIONS)

If the device is not capable of performing the operation requested by the command received, it responds with an error message according to the following format:

Device address	Function code	Exception code	CRC16	
Byte	Byte	Byte	LSByte	MSByte

- **Device address**:
  The address of the slave device which responds
- **Function code**:
  Functioning code with MSb =1 (to indicate the exception); example 0x83 (for the reading 0X03) or 0x86 (for the writing 0x06)
- **Exception code**:
  The following exception codes are managed by the device:

Exception code	Description	Cause for generation of exception
0x01	Function not implemented	And unavailable function code has been requested, different than 0x03, 0x06 and 0x2B/0x0E.
0x02	Invalid address	This is generated in different situations: - a register which is not implemented (or an inexistent area) has been requested - the reading of a number of registers which goes beyond the implemented area has been requested (starting from the requested address) - you have attempted to write in a read-only area
0x03	Invalid value for the datum	This is generated in different situations: - the reading of more than 10 registers has been requested - the DeviceIdCode of the message 0x2B/0x0E is not correct - you have attempted to write a parameter with a value out of range

- **CRC16:**
  The format error control field according to the algorithm CRC16.

N.B.: If the device identifies a format error in the message received or in CRC16, the message is discarded (it is not considered valid) and no response is generated.

2: DESCRIPTION OF COMMANDS

In order to standardise the interpretation mode, all the registers are managed in Word format (16 bit), even if they contain an 8 bit parameter.

2.1 REGISTER READING (0X03)

Format of command sent by Master:

Device address	Function code	Register address		Number of registers		CRC16	
Byte	Byte	MSByte	LSByte	MSByte	LSByte	LSByte	MSByte

- **Device address**:
  The address of the slave device to be interrogated
- **Function code**:
  Function code to be executed, in this case register reading (0x03)
- **Register address**:
  starting register address for reading expressed in two Bytes; (MSByte) and (LSByte).
- **Number of registers**:
  indicates the number of Words requested starting from the initial address. If more than 1 register is requested, in the response message all of the registers requested with consecutive addresses will be supplied starting from the address carried in the "register address" field.
  The number of registers to be read is expressed in two Bytes, in particular for this control (MSByte) it must always be 0x00 and (LSByte) with a 1-10 range.
- **CRC16:**
  The format error control field according to the algorithm CRC16.

Format of slave's response message:

Device address	Function code	N. of Bytes of datum	Datum 1		Datum 2		Datum n		CRC16	
Byte	Byte	Byte	MSByte	LSByte	MSByte	LSByte	MSByte	LSByte	LSByte	MSByte

- **Device address**:
  The address of the slave device which responds
- **Function code**:
  Function code being answered, in this case register reading (0x03)
- **Number of Bytes of datum**:
  contains the total number of Bytes of the data.
  Consider that the number of Bytes of the datum is twice the number of registers (since it deals with words). For example, if 2 registers are requested in the demand message, in the response message the number of Bytes of the datum must be set at 4.
- **Datum n** :
  contains the sequence of data each expressed in two Bytes; (MSByte) and (LSByte).
- **CRC16:**
  The format error control field according to the algorithm CRC16.

2.2 *WRITING OF SINGLE REGISTER (0X06)*

Format of command sent by Master:

Device address	*Function code*	*Register address*		*Datum*		*CRC16*	
Byte	Byte	MSByte	LSByte	MSByte	LSByte	LSByte	MSByte

- **Device address**:
  The address of the slave device to be interrogated
- **Function code**:
  Function code to be executed, in this case single register writing (0x06)
- **Register address**:
  register address to be written expressed in two Bytes; (MSByte) and (LSByte).
- **Datum**:
  value to be assigned to register expressed in two Bytes; (MSByte) and (LSByte).
- **CRC16:**
  The format error control field according to the algorithm CRC16.

Format of slave's response message:

Device address	*Function code*	*Register address*		*Datum*		*CRC16*	
Byte	Byte	MSByte	LSByte	MSByte	LSByte	LSByte	MSByte

The response message is a simple echo of the request message to confirm that the variable has been modified.

2.3 DEVICE IDENTIFICATION DATA READING (0X2B / 0X0E)

Format of command sent by Master:

Device address	Function code	Type MEI	Read Device Id Code	Object Id	CRC16	
Byte	Byte	Byte	Byte	Byte	LSByte	MSByte

- **Device address**:
  The address of the slave device to be interrogated
- **Function code**:
  Function code to be executed, in this case identification data reading (0x2B)
- **MEI type**:
  Type of Modbus Encapsulated Interface: must be 0x0E.
- **Read Device Id Code**:
  Indicates the type of data access: must be 0x01.
- **Object Id**:
  Indicates the starting object for data reading (range: 0x00 – 0x02).
- **CRC16:**
  The format error control field according to the algorithm CRC16.

Format of slave's response message:

Device address	Function code	Type MEI	Read Device Id Code	Conformity level	More Follows	Next Object Id	Number Of Object	Object Id (n)	Object Length (n)	Object Value (n)	CRC16	
Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	Byte	ASCII String	LSByte	MSByte

- **Device address**:
  The address of the slave device which responds
- **Function code**:
  Function code to be executed, in this case identification data reading (0x2B)
- **MEI type**:
  type of Modbus Encapsulated Interface: must be 0x0E.
- **Read Device Id Code**:
  indicates the type of data access: must be 0x01.
- **Conformity level**:
  indicates the conformity level of the slave: it is always 0x01.
- **More Follows**:
  indicates the additional number of transactions requested: it is always 0x00.
- **Next Object Id**:
  indicates the object to be requested in an eventual subsequent transaction: it is always 0x00.

- **Number Of Object**:
  number of objects which follow (1, 2 or 3).
- **List of**:
  - **Object Id**:
    current object number.
  - **Object Length**:
    length of following string.
  - **Object Value**:
    ASCII string containing identification information.
- **CRC16:**

The format error control field according to the algorithm CRC16.

Example of reading of all identification information of controls with UMIDMS03 rev.13 software and (address 1)

Request message: (01 2B 0E 01 00 70 77)

- **Device address**: 0x01
- **Function code**: 0x2B
- **MEI type**: 0x0E
- **Read DeviceIdCode**: 0x01
- **ObjectId**: 0x00
- **CRC16:** to be calculated on previous values

Response message: (01 2B 0E 01 01 00 00 03 00 04 50 45 47 4F 01 08 554D 49 44 4D 53 30 33 02 03 30 31 33 73 1C)

- **Device address**: 0x01
- **Function code**: 0x2B
- **MEI type**: 0x0E
- **Read DeviceIdCode**: 0x01
- **Conformity level**: 0x01
- **More Follows**: 0x00
- **Next ObjectId**: 0x00
- **Number Of Object**: 0x03
- **ObjectId**: 0x00
- **Object Length**: 0x04
- **Object Value**: 'PEGO' (Vendor Name field in ASCII)
- **ObjectId**: 0x01
- **Object Length**: 0x08
- **Object Value**: 'UMIDMS03' (Product Code field in ASCII)
- **ObjectId**: 0x02
- **Object Length**: 0x03
- **Object Value**: '013' (Revision field in ASCII)
- **CRC16:** to be calculated on previous values

REGISTERS AND ADDRESSES DESCRIPTION

Each register has a dimension of 16 bits. Blocks of variables have been formed (each with different address MSByte) depending on the type of variables. The following paragraphs describe in detail all the blocks available and, for each block, the variables implemented.
At the beginning of each table, on the first line is indicated whether the data corresponding to it can only be read (READ-ONLY) or read and written (READ/WRITE).

DESCRIPTION OF TABLE COLUMNS:

- **Register**:
  Indicates the register address to be used in the structure of the Modbus command to read or write the data in the instrument. It is expressed in two Bytes; (MSByte) and (LSByte).
- **Description**:
  Description of register and any corresponding programming variable of the instrument.
- **Meaning and range of Bytes:**
  Dimension (MSByte and LSByte), allowed range and notes relative to register.
- **U.M.:**
  Unit of measurement of datum contained in register.
- **Conv. :**
  The values in the registers which represent the variables with a sign require a conversion and are marked by an **X** sign in the following column.
  Conversion procedure:
  - if the value contained in the register is in between 0 and 32767, it represents a positive number or zero (the result is the value itself)
  - if the value contained in the register is in between 32768 and 65535, it represents a negative number (the result is the value of the register - 65536)
- **Molt:**
  Indicates the multiplication factor that must be applied to the datum of the register and which, combined with the U.m and Conv column, allows the value it contains to be interpreted exactly.
  Examples:
  A datum **(0x0012)** = 18 with Molt =**0.1** / U.m= **°C** / Conv=**C** corresponds to a temperature of (18x0,1)= **1.8 °C**
  A datum **(0xFFF0)** = 65520 with Molt =**0.1** / U.m= **°C** / Conv=**C** corresponds to a temperature [(65520 – 65536) x0.1] = **-1.6 °C**
  A datum **(0x0078)** = 120 with Molt =**1** / U.m= **min** / Conv=**C** corresponds to the time of (120x1)= **120 minutes**
  A datum **(0x0014)** = 20 with Molt =**0.1** / U.m= **°C** / Conv=**C** corresponds to a temperature of (20x0.1)= **2.0 °C**

3.1 *ANALOGICAL INPUTS*

READ-ONLY						
Register	***Description***	***Meaning and range of Bytes***		***U.M.***	***Conv***	***Molt***
256	Room humidity	MSByte LSByte	Resolution 1% range: 0 .. 99% Values > 99% indicate probe faulty	%		1
257	External humidification consent 0-10V	MSByte LSByte	Resolution 1% range: 0 .. 100%	%		1
258	Ambient temperature	MSByte LSByte	Resolution 0.1°C range: -45°C . +99°C Values > +99°C indicate probe faulty	°C	X	0,1
259	Channel humidity	MSByte LSByte	Resolution 1% range: 0 .. 99% Values > 99% indicate probe faulty	%		1
260	Current read by T.A.	MSByte LSByte	Resolution 0.1 A range: 0,0 .. 64.0A	A		0,1
261	Percentage of actual steam production (calculated)	MSByte LSByte	Resolution 1% range: 0 .. 100%	%		1

3.2 PARAMETERS

READ / WRITE						
Register	**Description**	**Meaning and range of Bytes**		**U.M.**	**Conv**	**Molt**
768	temperature setpoint	MSByte LSByte	0.1 ? steps range: 0..HSE	°C		0,1
769	Humidity setpoint	MSByte LSByte	1% steps range: 0..HSE	%		1
770	**Pr** Steam output percentage	MSByte LSByte	1% steps range: 20..100 %	%		1
771	**bP** proportional band	MSByte LSByte	1% steps range: 1..20 %	%		1
772	**StC** Channel humidity setpoint	MSByte LSByte	1 % steps range: 25..99 %	%		1
773	**r0** Channel humidity differential	MSByte LSByte	1 % steps range: 1..(StC - 20) %	%		1
774	**r1** temperature differential	MSByte LSByte	0.1 ? steps range: 0,2..10,0 °C	°C		0,1
775	**S0** deconcentration discharge duration	MSByte LSByte	0.1 second steps range: 0,1..12,7 seconds	sec		0,1
776	**S2** Deconcentration discharge interval	MSByte LSByte	1 minute steps range: 1..250 minutes	min		1
777	**S3** Electrode activ. delay after discharge	MSByte LSByte	1 second steps range: 1..12 seconds	sec		1
778	**S4** Water discharge due to inactivity	MSByte LSByte	1 hour steps range: 0..24 hours (0 = disabled)	hours		1
779	**S5** Min. current differential for loads	MSByte LSByte	0.1 ampere steps range: 0,2..10,0 amperes	A		0,1
780	**S6** Percentage of overcurrent due to discharge	MSByte LSByte	1 % steps range: 1..50 %	%		1
781	**S7** Duration of discharge due to overcurrent	MSByte LSByte	0.1 second steps range: 0,1..50.0 seconds	sec		0,1

Register	Description	Meaning and range of Bytes		U.M.	Conv	Molt
782	**S8** Min. filling current differential	MSByte LSByte	0.1 ampere steps range: 0,0..5,0 amperes (0 = loading by steps)	A		0,1
783	**S9** Operation setting	MSByte LSByte	steps of 1 range: 0..8	num		1
784	**S10** Discharge test	MSByte LSByte	range: 0..1 0 = disabled 1 = enabled	num		1
785	**CA1** Room humidity probe calibration	MSByte LSByte	steps of 1 %, with sign range: -20..+20 %	%	X	1
786	**CA2** Channel humidity probe calibration	MSByte LSByte	steps of 1 %, with sign range: -20..+20 %	%	X	1
787	**CA3** Room temperature probe calibration	MSByte LSByte	steps of 0.1 °, with sign range: -10.0..+10.0 °C	°C	X	0,1
788	**t1** Humidifier restart delay	MSByte LSByte	1 second steps range: 0..240 seconds	sec		1
789	**t2** Time ON for essence	MSByte LSByte	1 second steps range: 1..30 seconds	sec		1
790	**t3** Time OFF for essence	MSByte LSByte	1 minute steps range: 0..99 minutes	min		1
791	**t4** Manual activation time (in stand-by) of extraction fans	MSByte LSByte	1 hour steps range: 0..24 hours (0 = disabled)	hours		1
792	**t5** Activation time E9 serious Alarm	MSByte LSByte	1 minute steps range: 0..99 minutes (0 = disabled)	min		1
793	**In1** Digital input In1	MSByte LSByte	steps of 1, with sign range: -10..+10	num	X	1
794	**In2** Digital input In2	MSByte LSByte	steps of 1, with sign range: -10..+10	num	X	1
795	**In3** Digital input In3	MSByte LSByte	steps of 1, with sign range: -10..+10	num	X	1
796	**do4** Digital output do4	MSByte LSByte	steps of 1, with sign range: -3..+4	num	X	1
797	**do5** Digital output do5	MSByte LSByte	steps of 1, with sign range: -3..+4	num	X	1
798	**HSE** Maximum setpoint limit	MSByte LSByte	1 % steps range: 0..99 %	%		1

3.2a READ-ONLY PARAMETERS

READ						
Register	**Description**	**Meaning and range of Bytes**		**U.M.**	**Conv**	**Molt**
512	**S1** Humidifier operation hour meter	MSByte	1 hour steps range: 0..65535	hours		1
		LSByte				

3.2b CONFIGURATION PARAMETERS

READ / WRITE						
Register	**Description**	**Meaning and range of Bytes**		**U.M.**	**Conv**	**Molt**
1024	**I1** Current at 100% of production	MSByte	0.1 ampere steps range: 2,0..60,0 amperes	A		0,1
		LSByte				

INPUTS / OUTPUTS / ALARMS STATUS

READ-ONLY							
Register	**Description**	**Meaning of Bytes**			**U.M.**	**Conv**	**Molt**
1280	output status	MSByte	bit 7 (MSb)	Not used	num		1
			bit 6				
			bit 5				
			bit 4				
			bit 3				
			bit 2				
			bit 1				
			bit 0 (LSb)				
		LSByte	bit 7 (MSb)	Not used			
			bit 6	Not used			
			bit 5	Not used			
			bit 4	RL5 (do5) configurable			
			bit 3	RL4 (do4) alarm/configurab.			
			bit 2	RL3 water discharge pump			
			bit 1	RL2 EV water load			
			bit 0 (LSb)	RL1 electrodes			

READ-ONLY							
Register	**Description**	**Meaning of Bytes**			**U.M.**	**Conv**	**Molt**
1281	input status	MSByte	bit 7 (MSb)	Not used	num		1
			bit 6				
			bit 5				
			bit 4				
			bit 3				
			bit 2				
			bit 1				
			bit 0 (LSb)				
		LSByte	bit 7 (MSb)	Not used			
			bit 6	Reduction steam pr. from In1			
			bit 5	Discharge from digital input			
			bit 4	Discharge from keyboard input			
			bit 3	Reduction steam pr. from In3			
			bit 2	Max. level sensor			
			bit 1	Reduction steam pr. from In2			
			bit 0 (LSb)	Steam production consent			

READ-ONLY							
Register	**Description**	**Meaning of Bytes**			**U.M.**	**Conv**	**Molt**
1282	alarm status	MSByte	bit 7 (MSb)	Not used	num		1
			bit 6	Not used			
			bit 5	Not used			
			bit 4	Not used			
			bit 3	Not used			
			bit 2	Not used			
			bit 1	Serious alarm from digital input (E9)			
			bit 0 (LSb)	Alarm from digital input (E8)			
		LSByte	bit 7 (MSb)	EEPROM error (EE)			
			bit 6	Level alarm			
			bit 5	Water discharge alarm (E6)			
			bit 4	Water discharge pre-alarm (E5)			
			bit 3	Water load alarm (E3)			
			bit 2	Channel probe fault (E2)			
			bit 1	Current reading fault (E1)			
			bit 0 (LSb)	Room probe fault (E0)			

DEVICE STATUS

READ / WRITE							
Register	**Description**	**Meaning of Bytes**			**U.M**	**Conv**	**Molt**
1536	device status	MSByte	bit 7 (MSb)	not used	num		1
			bit 6	not used			
			bit 5	not used			
			bit 4	timed extraction fan forcing enabling			
			bit 3	essence status modification forcing enabling			
			bit 2	hour meter reset status modification enabling			
			bit 1	discharge forcing modification enabling by Modbus			
			bit 0 (LSb)	stand-by status modification enabling			
		LSByte	bit 7 (MSb)	not used			
			bit 6	not used			
			bit 5	not used			
			bit 4	timed extraction fan forcing 1 = enables timed functioning of extraction fans (if conditions exist) 0 = no function			
			bit 3	essence enabling status change forcing. 1 = sets essence enabling at ON 0 = sets essence enabling at OFF			
			bit 2	hour meter reset forcing by Modbus 1 = resets hour meter by Modbus 0 = does not reset hour meter by Modbus			
			bit 1	discharge forcing status by Modbus 1 = discharge force by Modbus 0 = discharge OFF by Modbus			
			bit 0 (LSb)	stand-by status 1 = stand-by 0 = ON			
1537	Steam production forcing by Modbus (only if S9=8)	MSByte	1% steps range: 0 .. 100%		%		1
		LSByte					

To request the modification of one of the status bits of the device, the master must send the requested value for the bit in LSByte and the corresponding bit set at 1 in MSByte. Example: in able to force the stand-by status, the master must send MSByte = 00000001 and LSByte = 00000001. As soon as the command is sent the hour meter is immediately reset at zero and the relative bit will be read at 0 (to check whether the command was successful, just make sure that the hour meter S1 has been reset at zero).

N.B. – The discharge forcing and steam production commands envision a 1 minute time-out; if during this

period the command is not sent back, upon expiring of this time the steam production will be set at zero and the discharge pump will be disabled. This places the humidifier in safe conditions in case of accidental disconnections of the network.

4: GLOSSARY

- **Binary Number:**
  This is used in computing for the internal representation of numbers, thanks to the simplicity of physically implementing an element with 2 states (0,1) instead of a higher number, but also for the correspondence with logical true and false values.

- **Decimal number:**
  In the decimal system, all whole numbers are represented by using the 10 digits which indicate the first ten natural numbers, including zero. The value of each of these digits depends on its position inside the number and increases by powers of 10 proceeding from the right to the left.

- **Hexadecimal number:**
  This belongs to a base 16 positional numerical system; namely which uses 16 symbols instead of the 10 in the traditional decimal numerical system. For the hexadecimal, normally symbols from 0 to 9 and than letters from A to F are used, for a total of 16 symbols. By agreement, a number expressed in hexadecimal is preceded by 0x (example 0x03) or by H (example H03).

- **bit:**
  A bit is a binary digit, namely one of the two symbols of the binary numerical system, traditionally called zero (0) and one (1). It represents the unit of definition of a logical status. It is also defined as the elementary computing unit treated by a processor.

- **Byte:**
  It is the amount of bits necessary to define an alphanumerical character; in particular one Byte is made up of the sequence of 8 bits (ex. 10010110).

- **Word:**
  Unit of measurement which fixes information length at 16bits which is also equivalent to 2 Bytes (ex. 10010110 01101011).

- **LSb:**
  least significant bit of a binary number (first bit to the right of the number indicated)

- **MSb:**
  most significant bit of a binary number (first bit to the left of the number indicated)

- **LSByte:**
  least significant Byte of a Word (Byte to the right of the indicated Word)

- **MSByte:**
  most significant Byte of a Word (Byte to the left of the indicated Word)

PEGO s.r.l.
Via Piacentina, 6/b
45030 OCCHIOBELLO –ROVIGO-
Tel : 0425 762906
Fax: 0425 762905
www.pego.it
e-mail: info@pego.it

Distributor: