# WINDMETER®

with **VANEMETER**

## USER MANUAL

**fw ver. 81.02-00.00 / hw rev. RGB103**

### GENERAL DESCRIPTION

The WindMeter® (WM) is a high technology electronic device primary designed to accurately measure the wind speed and make it available to the user in the best suitable way for its applications.

It's mainly intended, but not limited, to be used in wind energy applications for preliminary studies, for commissioning testing and for continuous performance checking and monitoring.

It's based on a 3 cup anemometer that through an internal CPU give values of wind speed available in Serial Modbus RTU open protocol. Values are already filtered to give reliable values along the time.

It's equipped with an additional temperature sensor that through same Modbus protocol give the working temperature of cup anemometer.

The measures can be read by a powerful, versatile EIA/TIA-RS485 bus interface with the well known industry standard protocol Modbus RTU.

### FEATURES

**Inputs:**

| | |
|---|---|
| wind speed: | 0 ÷ 50 m/s (0 ÷ 180 km/h) |
| temperature range: | -30 ÷ +85 °C measurable with internal temp. sensor |

**Outputs:**

| | |
|---|---|
| serial: | RS485, standard Modbus RTU protocol |

**Measurements precision:**

| | |
|---|---|
| resolution: | 0.227 km/h |
| precision: | < ± 2% |
| starting velocity | > 0.5m/s |

| | |
|---|---|
| **Supply:** | 5 ÷ 40 Vdc / 9 ÷ 28 Vac |
| **Case:** | anodized aluminium with screws to fix it on one end of a pole |
| **Wiring:** | 3 m cable UV resistant, 4 conductors |
| **Connectors:** | M12 8 pin, IP67 code, UV resistant |
| **Dimensions:** | rotor diameter : Ø 166mm<br>body diameter: Ø 44 ext., Ø 36 int. mm<br>height: 223 mm |
| **Operating temperature:** | -30 ÷ +70 °C (transport and storage -35°C ÷ +70 °C) |

**Every WM is factory calibrated.**

### PIECE'S LIST

- WM with screws for end pole positioning
- instruction manual
- calibration report
- transportation box

FINAL TEST AND CALIBRATION REPORT:

S/N .....

OPER.....

DATE.....

**Important : Do not try to open the case of Windmeter: warranty will be void.**

## ASSEMBLY

WM is provided with screws to be applied in end pole of a 12 - 35mm diameter. For applications in other locations (i.e. in a arm) it is necessary a mounting bracket

## CONNECTIONS

### [without connector and without Vanemeter]

Correspondence colour wires - signals

| colour | Name | Description |
|---|---|---|
| black | SUPPLY 1 | power supply pin1 |
| red | SUPPLY 2 | power supply pin2 |
| green | RS485+/B | communication bus non inverting signal (**note 1**) |
| white/green | RS485-/A | communication bus inverting bus signal (**note 1**) |

Tab. 1

### [with connector M12 8Pole]

The IP67 M12 8-pin circular male connector carries supply and signals to and from the WM as in Tab. 1 and Fig. 3, that shows a front view of the male connector (or a back side view of the female connector):

| # | Name | Description | Cable colors |
|---|---|---|---|
| 1 | SUPPLY +VIN | power supply input, 5-40 Vdc / 9-28 Vac, typ. 90mA @ 12 Vdc (**note 1**) | Red |
| 2 | GND | power supply ground reference and for output signals | Black |
| 3 | | Not used | |
| 4 | Analog Input | Vanemeter signal input | white/yellow |
| 5 | RS485-/A | communication bus inverting bus signal (**note 2**) | White/Green |
| 6 | RS485+/B | communication bus non inverting signal (**note 2**) | Green |
| 7 | Digital Input | Reserved | |
| 8 | | Not used | |

Tab. 2

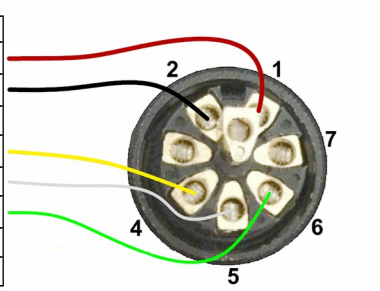


Fig. 2: Female connector back view with connection scheme

WM +VN

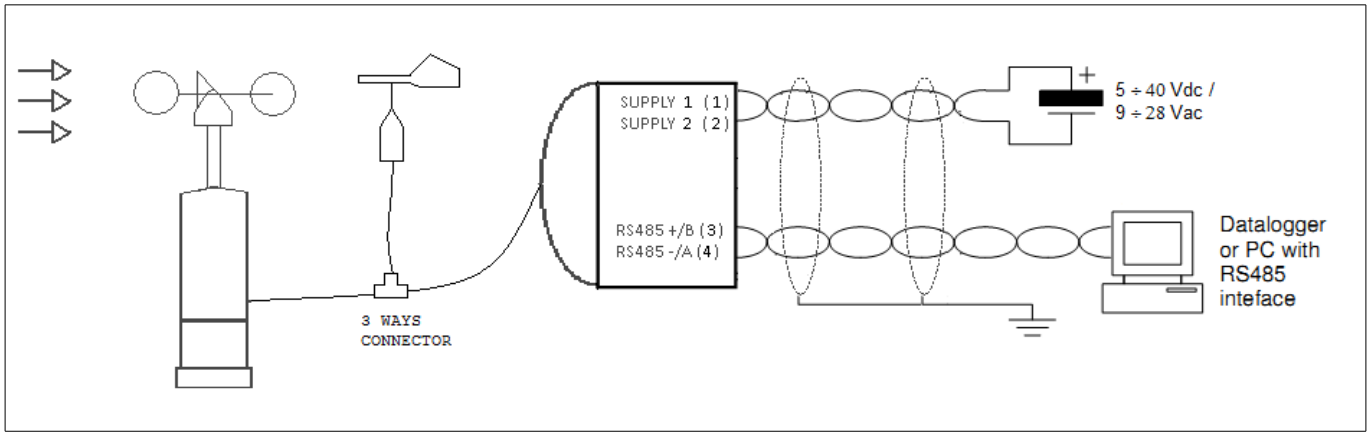


Fig 3: typical connections/usage

We strongly suggest to use a shielded connection cable with twisted pairs, AWG22 / 0.32mm$^2$

Notes:

1) balanced differential bus RS485 needs to be terminated, at the extremities of the bus, by a 100-120 Ω resistor (1/4 W) between RS485+/RS485- lines in order to avoid signal's reflections. In the case that SM is the device at one extremity, place the resistor into the supplied female connector.
   Even if RS485 have –7/+12Vdc common mode rejection range, normally sufficient to compensate ground potential difference between connected devices, it is strongly recommended to always carry a ground reference among the bus signals and to connect it to the SM's Signal GND.

2) the digital input need to be activated by shorting to GROUND (either supply or signal, latter preferably). Do not attempt to supply voltage to this input.

## MODBUS PROTOCOL

Modbus is a Master-Slave protocol, widely used as an industry standard. Simple, efficient and reliable, can be easily used to access and collect data or exchange information between digital systems over a serial line local bus (and with its TCP/IP extension through a LAN or World Wide Web).
Please refer to specific detailed documentation and implementations freely available at www.modbus.org

WM is a Modbus RTU slave that implements the following standard access functions:

| Function code | Description |
|---|---|
| **0x03** | READ HOLDING REGISTERS |
| **0x04** | READ INPUT REGISTERS |
| **0x06** | WRITE SINGLE REGISTER |
| **0x10** | WRITE MULTIPLE REGISTERS |

Tab. 3

Please note that in the current implementation of WM function codes 0x03 and 0x04 are equivalent and address the same data area.

Data is accessible through Modbus's functions by 16 bits units called "registers". In the current implementation of WM these registers are available:

| Register # | Description | Access | NV save |
|---|---|---|---|
| **0x0101** | **Current Speed** (5s.) [speed unit], 10.6 format (x64) | R | |
| **0x0102** | **Current Direction** (5s.) [ ° ], Integer format ( x 1) | R | |
| **0x0103** | **Status**, bit coded<br>Bit 0: Factory calibration/configuration; 1 = OK; 0 = need recalibration<br>Bit 1: Not volatile parameters; 1 = OK; 0 = default loaded, need to be changed/saved<br>Bit 2: Digital input monitor; 1 = not active (open); 0 = active (shorted to GND)<br>Bit 3: DIP switch 1; 1 = electrical contact open; 0 = electrical contact shorted<br>Bit 4: DIP switch 2; 1 = electrical contact open; 0 = electrical contact shorted<br>Bit 5: Watchdog; 1 = reset by watchdog timeout occurred; 0 = normal operation<br>all undefined bits read as 0 | R | |
| **0x0104** | **Last min. average speed** [speed unit], 10.6 format (x64) | R | |
| **0x0105** | **Last min. max speed** [speed unit], 10.6 format (x64) | R | |
| **0x0106** | **Last 2 min. average speed** [speed unit],10.6 format (x64) | R | |
| **0x0107** | **Last 2 min. max. speed** [speed unit],10.6 format (x64) | R | |
| **0x0108** | **Last 5 min. average speed** [speed unit],10.6 format (x64) | R | |
| **0x0109** | **Last 5 min. max. speed** [speed unit],10.6 format (x64) | R | |
| **0x010A** | **Last 10 min. average speed** [speed unit],10.6 format (x64) | R | |
| **0x010B** | **Last 10 min. max. speed** [speed unit],10.6 format (x64) | R | |
| **0x010C** | **Highest speed** [speed unit], | R/W* | |
| **0x010D** | **Current direction, unit vector** (5s) [ ° ], Integer format ( x 1) | R | |
| **0X010E** | **Last min. direction unit vector** [ ° ], Integer format ( x 1) | R | |
| **0X010F** | **Last 2 min. direction unit vector** [ ° ], Integer format ( x 1) | R | |
| **0X0110** | **Last 5 min. direction unit vector** [ ° ], Integer format ( x 1) | R | |
| **0X0111** | **Last 10 min. direction unit vector** [ ° ], Integer format ( x 1) | R | |
| **0X0112** | **Current speed resultant vector** (5s) [speed unit],10.6 format (x64) | R | |
| **0X0113** | **Current direction resultant vector** (5s) [ ° ], Integer format ( x 1) | R | |
| **0X0114** | **Last min. speed resultant vector** [speed unit],10.6 format (x64) | R | |
| **0X0115** | **Last min. direction resultant vector** [ ° ], Integer format ( x 1) | R | |

| | | | |
|---|---|---|---|
| **0X0116** | **Last 2 min.speed resultant vector** [speed unit],10.6 format (x64) | R | |
| **0X0117** | **Last 2 min. Direction resultant vector** [ ° ], Integer format ( x 1) | R | |
| **0X0118** | **Last 5 min. speed resultant vector** [speed unit],10.6 format (x64) | R | |
| **0X0119** | **Last 5 min. direction resultant vector** [ ° ], Integer format ( x 1) | R | |
| **0X011A** | **Last 10 min. speed resultant vector** [speed unit],10.6 format (x64) | R | |
| **0X011B** | **Last 10 min.direction resultant vector** [ ° ], Integer format ( x 1) | R | |
| **0x8001** | **Serial number**, least significant word | R | |
| **0x8002** | **Serial number**, most significant word | R | |
| **0x8003** | **Firmware main version**, hexadecimal | R | |
| **0x8004** | **Firmware minor version**, hexadecimal | R | |
| **0x8005** | **Node address**, range 1 ÷ 247, decimal, **default 1** | R/W | Y |
| **0x8006** | **Bitrate**, coded, range 0 ÷ 4, decimal, **default 1**<br>0 – 9600 bps<br>1 – 19200 bps<br>2 – 38400 bps<br>3 – 57600 bps<br>4 – 115200 bps | R/W | Y |
| **0x8007** | **Serial configuration**, coded, range 0 ÷ 3, decimal, **default 0**<br>0 – 8N1 (8 bit / no parity / 1 stop bit)<br>1 – 8E1 (8 bit / even parity / 1 stop bit)<br>2 – 8O1 (8 bit / odd parity / 1 stop bit)<br>3 – 8N2 (8 bit / no parity / 2 stop bit) | R/W | Y |
| **0x8008** | **Serial reply delay** [ms], range 0 ÷ 100, decimal, **default 1** | R/W | Y |
| **0x8009** | **Speed unit**, coded, range 0 ÷ 4, decimal, **default 0**<br>0 – m/s<br>1 – km/h<br>2 – ft/s<br>3 – knot<br>4 – mph | R/W | Y |
| **0x8101** | **Not volatile params save command**, write 1 to execute (then wait 1 s before to send next message) | W | |
| **0x8102** | **Software reset command**, write 1 to execute (then wait 6 s before to send next message) | W | |
| **0x8201** | **Analog output level** [], range 0 ÷ 65535, decimal, fixed point 0.16 format (16 bits fractional) | W | |

Tab. 4

Please note that, conventionally, Modbus register's numbering starts from 1 but register's addressing start from 0 so, to obtain the register's address you had simply to subtract 1 from its number. That's meaningful depending on, as a master, you are using an high level Modbus utility/program (that normally refers to the registers' number) or a low level driver (that normally directly works with addresses).
(*) Write access is allowed only for reset, i.e. zero value.

## CALIBRATION

Windmeter can be calibrated in MEASNET laboratory. However, if calibration is made by our factory, we have developed wind conditions and procedure able to insert corrective parameters into firmware, by direct comparision of a MEASNET calibrated Windmeter.

## OPTIONALS

Available upon request:

- Wind Vane device. Can be wired to a VANE with output in 0 - 5V input range or 4 – 20mA current loop

## CONTACTS

Software utilities (for MS Windows systems) and other solar products can be requested to the following address:

**Soluzione Solare**
Tel. +39.0444.530234 - Fax +39.0444.1830563 Vicenza – Italy E-mail: support@soluzionesolare.it