Meteorological Instruments Expert

HONGYUV

HY-SA3E Ultrasonic Anemometer

User Manual

12/24/2023

Table of Contents

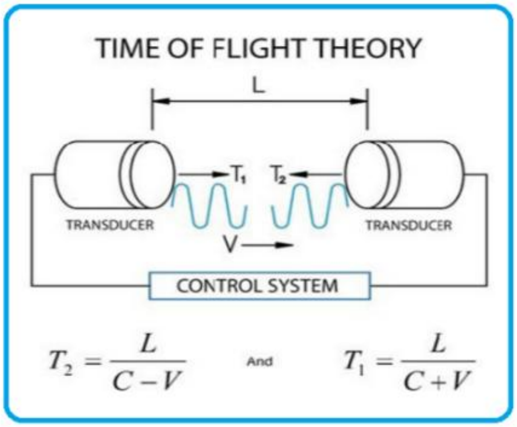
1. Introduction

This User Manual provides necessary information for user to install and communicate with HY-SA3E ultrasonic anemometer.

1.1 Overview

HY-SA3E ultrasonic anemometer is specially designed for Unmanned Aerial Vehicle UAV. It’s very robust with no moving parts, maintenance free. It can simultaneously output wind speed and direction. Each unit is calibrated in our wind tunnel testing lab prior to shipment. Typically, HY-SA3E can be powered up with 5-30 VDC, please note that some specially customized sensors may not support wide power supply range. It can be communicated by one of following communication protocols:
SDI-12, MODBUS RTU and NMEA 0183.
Three alternative serial interfaces are selectable: RS232, RS485 and SDI-12. It is typically equipped with a 4-pin M12 connector for installation.(4-20mA output will need more Pins)

Measure the transmission time of ultrasonic sensors from sensor N to sensor S, and compare with the transmission time of sensor S to sensor N. Similarly, compare the time of W to E and E to W time. (N = north, S = south, E = east, W = west) For example, if the wind blew from the north, time of ultrasonic from N to S will be shorter than from S to N, and transmission time of it from W to E and E to W is the same. Through calculating the time difference of ultrasonic transmission between two points, the wind speed and direction can be calculated. This calculation method has nothing to do with other factors such as temperature. The wind speed is represented as a scalar speed in units m/s. The wind direction is expressed in degrees (°). The wind direction reported indicates the direction that the wind comes from. North is represented as 0°, east as 90°, south as 180°, and west as 270°. About measurement sampling rate: The ultrasonic probes sample hundreds of times per second, and process those raw data as wind speed and wind direction output every second.



TECHNICAL SPECIFICATION

Parameters	Range	Accuracy	Resolution
Wind Speed	0 - 70m/s	±3%	0.1m/s
Wind Direction	0 - 360°	±3°	1°
Digital Output	RS485,RS232,SDI-12		
Analog Output	4-20mA(optional)		
Baud Rate	1200 ~ 115200 bps		
Communication Protocol	ModBus RTU or NMEA-0183 or SDI-12		
Protection Grade	IP65		
Operating Temperature	-40℃ - +70℃		
Storage Temperature	-50℃ - +80℃		
Working Humidity	0 - 100%		
Power Supply	VDC: 5-30V		
Power consumption	20mA @5V		
Dimension/Weight	Φ104×136 mm / 0.5kg		
Material	ASA engineering plastic		

Specifications may be subject to change without prior notice.

2. Getting Started

2.1 Wiring

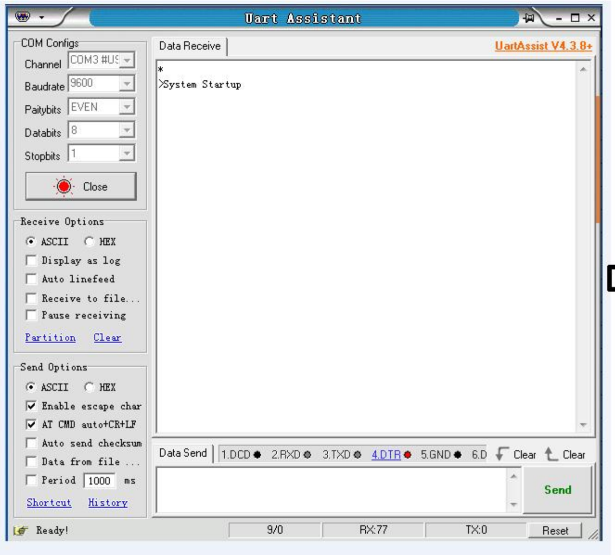
Color of cable		Red	Black	Yellow	Green
Definition	RS485	Power+	Power- & RS232 GND	DA+	DB-
	SDI-12			-	SDI-12

2.2 Communication test

Note: We don't recommend user to install sensor or connect it to your data logger before testing communication by procedure below. We highly recommend you to use corresponding interface to USB converter to to test our sensor on computer before any further operation.

Our sensor will send ASCII string ">System Startup" once you power it up.

Thereafter, you should receive "CONFIGURE MODE" after sending ">*\r\n" to it. (or send in HEX: 3E2A0D0A)Until now, you can confirm that communication of sensor is working well.



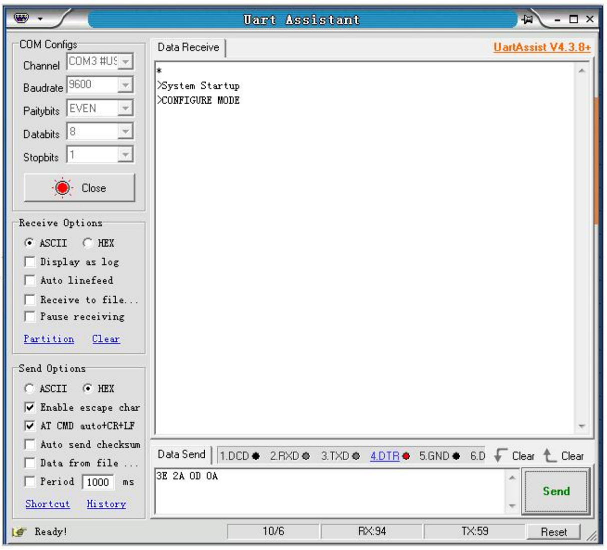
Channel: COM3 #US
Baudrate: 9600
Paitybits: EVEN
Databits: 8
Stopbits: 1

Receive Options: ◉ ASCII ○ HEX
□ Display as log
□ Auto linefeed
□ Receive to file...
□ Pause receiving

Send Options: ◉ ASCII ○ HEX
☑ Enable escape char
☑ AT CMD auto+CR+LF
□ Auto send checksum
□ Data from file ...
□ Period 1000 ms

```
*
>System Startup
```

Ready! 9/0 RX:77 TX:0



Channel: COM3 #US
Baudrate: 9600
Paitybits: EVEN
Databits: 8
Stopbits: 1

Receive Options: ◉ ASCII ○ HEX
□ Display as log
□ Auto linefeed
□ Receive to file...
□ Pause receiving

Send Options: ○ ASCII ◉ HEX
☑ Enable escape char
☑ AT CMD auto+CR+LF
□ Auto send checksum
□ Data from file ...
□ Period 1000 ms

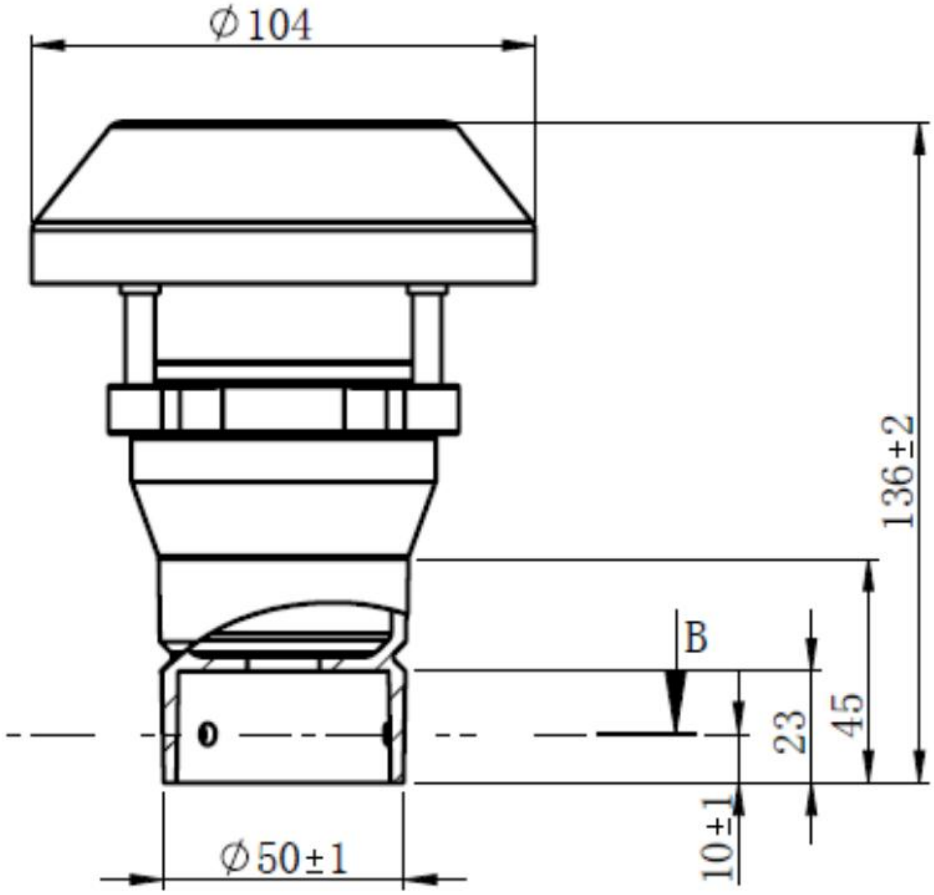
```
*
>System Startup
>CONFIGURE MODE
```

Data Send:

```
3E 2A 0D 0A
```

Ready! 10/6 RX:94 TX:59

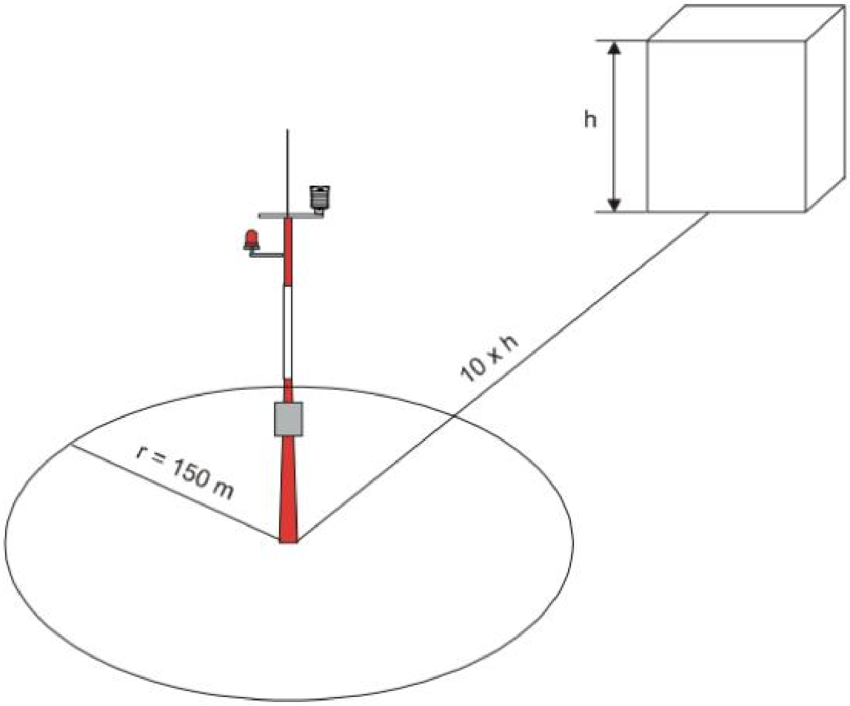
2.3 Mounting



2.4 Select Installation Location

- The mounting location must not be sheltered from the wind or in the lee of objects such as chimneys or satellite systems.
- The anemometer must be mounted in an upright position; otherwise, water can enter the anemometer and destroy it.
- If possible, install the anemometer in the center of flat roofs. Installing the anemometer at the edge of the roof may cause air turbulence that may distort the measuring results.
- Cable length come with anemometer is default in 4 meters. This cable may be cut or extended to a maximum of 1000 meter(for RS485) or 10 meters(for RS232)

Following World Meteorological Organization (WMO) guidelines, a general recommendation is that there is at least 150 m open area in all directions from the mast. Any object of height (h) does not significantly disturb wind measurement at a minimum distance of 10 times the height of the object. The recommended minimum length (h) for the mast that is installed on top of a building is 1.5 times the height of the building (H). When the diagonal (W) is less than the height (h),the minimum length of the mast is 1.5 W. However, follow the application specific instructions and local regulations when placing anemometer.



3. Communicate with sensor

Noted: Our sensor has
Digital interface: RS485, SDI-12 for option.
Those are decided when you place order, and they can't be changed after production.
Following are details of optional protocols for your reference.

3.1 MODBUS RTU protocol

Unless you request for other configuration, otherwise sensor will adopts following default configuration:
Start Bit: 1 bit; Data Bits: 8 bits; Parity:NONE; Stop Bits: 1bit;
Baud Rate:9600 bps
Data structure of Modbus command and response as below:

Device Address	Function Code	Data	CRC checksum

Modbus protocol is based on time-sharing poll mechanism, it's essential to assign an unique device address to each sensor.
Function Code: We use 03 to read data from registers
CRC: Our checksum use CRC-16/MODBUS

An example to read data from three registers:

Inquiry: 01 03 00 00 00 04 44 09
Response: 01 03 08 5D FF 00 24 00 00 40 00 1F BA

	Device Address	Function Code	Data					CRC Check sum
Inquiry command			Read data start from register *			Read * registers		
	01	03	0000			0004		4409
	Description		Read data start from register 1			Read 4 registers		
Response message			Bytes Quantity	Register 1	Register 2	Register 3	Register 4	
	01	03	08	5DFF	0024	0000	4000	1FBA
	Description		8 bytes	Sensor State	Wind Dir:36° HEX to Decimal	Wind Speed:2.0m/s IEEE754 Standard CDAB float type		

MODBUS RTU register definition map for example.

Address of register	Length	Data Type	Definition	Range
0000H	16 bit	16 bit int	Reserved	Reserved
0001H	16 bit	16 bit int	Wind Direction	0 - 359 °
0002H	16 bit	32 bit float	Wind Speed	0 - +40 m/s all 32 bit float date in this protocol comply to IEEE754 Standard 32 Bit Float Little_endian byte swap
0003H	16 bit			

3.2 NMEA0183 protocol

Unless you request for other configuration, otherwise sensor will adopts following default configuration:

Start Bit:1 bit; Data Bits: 8 bits; Parity:NONE; Stop Bits:1bit;

Baud rate:9600 bps

Wind speed and direction data frame

$WIMWV,180,R,0.01,M,A*06<CR><LF>

Data segment	E.g.	Format	Name	Description
0	**$WIMWV**	String	Fixed frame head	NMEA Wind Protocol Header
1	**180**	Number	Wind direction	Wind direction reading in °
2	**R**	Character	Type of wind	R stand for Relative wind speed and direction
3	**0.01**	Number	Wind speed	Wind speed reading
4	**M**	Character	Unit of Wind speed	K stand for Km/h M stand for m/s N stand for knots

5	**A**	Character	Status of data	A=Valid
				V=Invalid

6	***06**	HEX character	Hex checksum	Checksum, the 2 digit Hex Checksum sum figure is calculated from the Exclusive OR of the bytes between (and not including) the $ and * characters.
7	**<CR><LF>**	ASCII character	Line Feed Character	

3.3 Unsolicited active output in ASCII protocol

Unless you request for other configuration, otherwise sensor will adopts following default configuration:

Start Bit: 1 bit; Data Bits: 8 bits; Parity:NONE; Stop Bits: 1bit;
Baud Rate:9600 bps
Data frame(ASCII)

E.g.

:01,56,0.3,...,**<CR><LF>

Data Segment	Char acter	Description	Number of Characters	Remark
Frame header	:	Frame header	1	0x3A
Address	01	Device Address	2	0x01~0xFF
Separator	,		1	
Data 1	56	Wind direction	Unfixed length	0~359 °
Separator	,		1	
Data 2	0.03	Wind speed	Unfixed length	0~40 m/s
Separator	,		1	
Data 3	... other data			
Separator	,		1	
Checksum	**	LRC Checksum	2	0x00~0xFF
Frame Ending	<CR>	Frame Ending 1	1	0x0D
	<LF>	Frame Ending 2	1	0x0A

3.4 SDI-12 protocol

Unless you request for other configuration, otherwise sensor will adopts default configuration:

Start Bit: 1 bit; Data Bits: 7 bits; Parity:EVEN; Stop Bits: 1bit;

Baud Rate:1200 bps

SDI-12 Commands without CRC

?! Returns Unit Address (default is 0).

a Current unit address letter (factory default is 0, range is 0 to 9, a to z, A to Z).

b New address letter, range is 0 to 9, a to z, A to Z.

aAb! Change unit address from a to b see above.

aM! Address, Relative Wind Direction, Speed, True Wind Direction, Speed,State.

aD0! Request a line of the above data

Command	Description	Response	Example
?!	Inquiry unit address	a<CR><LF>	0<CR><LF>
aAb!	Change the unit address from a to b	b<CR><LF>	1<CR><LF>
aI!	Unit Identification	a...<CR><LF>	014HONGYUV 1000002.3000<CR><LF>
aM!	Request to prepare data of Relative Wind Direction, Speed, True Wind Direction, Speed	atttn<CR><LF> a:address ttt:data preparation time in seconds. n:qty of data	00015<CR><LF> Address 0 Data will be ready within 1 second 5 data will be provided
aD0!	Fetch data of above command	<address> <relative wind direction> <relative wind speed> <true wind direction> <true wind speed> <status> <CR><LF>	0+056+00.03+000+00.00+1100 <CR><LF> 0:address 056:relative wind direction 56° 00.03:relative wind speed 0.03m/s

			000: true wind direction 56° 00.00:true wind speed 0.00m/s 1100:state of each data, 1 stand for valid, 0 stand for invalid

SDI-12 Commands with CRC

aMC! Address, Relative Wind Direction, Speed, True Wind Direction, Speed,State and CRC.

aD0! Request a line of the above data

Command	**Description**	**Response**	**Example**
aMC!	Request to prepare data of Relative Wind Direction, Speed, True Wind Direction, Speed	atttn<CR><LF> a:address ttt:data preparation time in seconds. n:qty of data	00015<CR><LF> Address 0 Data will be ready within 1 second 5 data will be provided
aD0!	Fetch data of above command	<address> <relative wind direction> <relative wind speed> <true wind direction> <true wind speed> <status><CRC> <CR><LF>	0+056+00.03+000+00.00+1100CJP <CR><LF> 0:address 056:relative wind direction 56° 00.03:relative wind speed 0.03m/s 000: true wind direction 56° 00.00:true wind speed 0.00m/s 1100:state of each data, 1 stand for valid, 0 stand for invalid CJP:CRC Checksum

3.5 Serial port configuration commands

You can download our software for Windows OS to configure serial port by link:
http://www.hongyuv.com/download/Serial_Configuration_Tool.zip

Following commands can be used to change communication address, baud rate, parity, data bit, stop bit;

Commands		Content	Response
Instruction 1	ASCII	>*\r\n	>CONFIGURE MODE\r\n
	HEX	3E 2A 0D 0A	0A 3E 43 4F 4E 46 49 47 55 52 45 20 4D 4F 44 45 0D 0A
Remark	Enter Setting Mode		
Instruction 2	ASCII	>CUS 9600 8-N-1\r\n	>CMD IS SET
Remark	Configure serial port configuration as Baud Rate 9600 bps; Data bits:8 bits; Parity:NONE; Stop bits:1 bit.		
Instruction 3	ASCII	>ID 2\r\n	>CMD IS SET
Remark	Configure address of device as 2. Inquiry address command is HEX: 3E 49 44 0D 0A		
Instruction 4	ASCII	>RESET\r\n	System start ok!\r\n
Remark	Reboot device		
Instruction 5	ASCII	>!\r\n	>NORMAL MODE\r\n
Remark	Exit setting mode to normal mode.		
Notice: 1. Characters “\r\n” is CRLF Carriage-Return Line-Feed, corresponding to HEX (0x0D,0x0A)			

Procedures:

No.	Function	Instructions
1	Set Communication Address	1→3→5→4
2	Set Serial Port Parameters	1→2→5→4

3.6 Explanation of decoding CDAB float type data

All 32 bit float date in this protocol comply to IEEE754 Standard
32 Bit Float Little_endian byte swap
For example:
Send command to fetch data in register 02H and 03H:
Send: 01 03 00 02 00 02 65 CB
Received: 01 03 04 51 EC 40 18 1A F0
Now we get raw data 51 EC 40 18

1.We swap higher and lower byte, then we get 40 18 51 EC
2.We convert it into binary:
HEX:401851EC
=>Binary:01000000000110000101000111101100 (32 bits)

1st bit is sign bit, 0 stand for positive and 1 stand for negative.
Exponent= (convert 2nd~9th bits to decimal - 127)=128-1=1

Mantissa=(convert 10th~32nd bits to decimal)/(2^23)
=1593836/8388608=0.19000005722045898484375

Final result=sign*(1+Mantissa)*2^Exponent
=2.38000011444091796875

Now we succeed to decode CDAB float data “401851EC” as 2.38, which means we had 2.38 m/s wind speed.

You can also use C language's sub-function to transform 4 bytes(HEX) as float data(C language)

```
union
{
float  TestData_Float;
unsigned char TestArray[4];
}TData;
```

Analysis example:

D3	D2	D1	D0
Higher byte of register 2	Lower byte of register 2	Higher byte of register 1	Lower byte of register 1
40	AC	19	DF
Higher byte	Middle byte 1	Middle byte 2	Lower byte

After transformed to float data, value: 5.378

Subfunction:

```
float Tempfloat;
TData.TestArray [3]= 0x40; //input higher byte
TData.TestArray [2]= 0xac; //
TData.TestArray [1]= 0x19; //
TData.TestArray [0]= 0xdf; //input lower byte
Tempfloat = TData.TestData_Float;  //return result 5.378
```

3.7 Explanation of CRC calculation

Our checksum use CRC-16/MODBUS

The CRC we are using is 16 bits, lower byte comes first.

The cyclic redundancy check (CRC) field is two bytes which contain 16 bits binary value. The value of the CRC appended to the message is calculated by the transmitting device. When receiving the message, the receiving device recalculates the CRC value and compares the calculated result with the actual received CRC value. If the two values are not equal, it is an error.

During the generation of CRC, each 8-bits characters are XOR with the value in the register. The result then shifts 1 bit in the LSB direction, while the MSB position is charged to zero. Then extract and check LSB: if LSB is 1, the value in the register is XOR with a fixed preset value; if LSB is 0, no XOR operation is performed.

This process will be repeated until 8 shifts have been performed. After the last (8th) shift and related operations, the next 8-bit byte is XOR with the current value of the register, and then repeat 8 times as described above. The final value in the register obtained after all sub sections of the message are calculated is CRC.

Procedure of calculating a CRC:

1. Load a 16 bit register with hexadecimal FFFF (all 1). Call it CRC register
2. XOR the first byte of 8 bits in message with the low er byte of the 16 bit CRC register, and place the result in the CRC register
3. Move the CRC register to the right by 1 bit (in the direction of LSB), fill the MSB with zero, extract and detect LSB
4. If LSB is 0: repeat step 3 (do another shift)
(if LSB is 1): conduct XOR operation with CRC register.
5. Repeat steps 3 and 4 until 8 shifts have been completed. When this is done, the full 8-bit byte operation will be completed.
6. Repeat steps 2 to 5 for the next byte in the message, and continue the operation until all messages are processed.
7. The final content in CRC register is CRC value
8. When placing CRC value in message, as described below, higher and lower byte must be exchanged.

3.8 Explanation of LRC calculation

Using a C language function code to generate LRC values

The function code uses 2 independent variables：

```
unsigned char *auchMsg;  // To generate the LRC value, point the pointer to the buffer containing the binary data
unsigned short usDataLen; //Number of bytes in the buffer。
//This function returns LRC as a type“unsigned char” 。
// RC check code generation
static unsigned char LRCCheck(auchMsg, usDataLen)
unsigned char *auchMsg; /* calculating by information byte LRC*/
unsigned short usDataLen; /*calculating by information byteLRC*/
{
unsigned char uchLRC = 0 ; /*Initializing LRC characters */
while (usDataLen--) /*through the data buffer*/
uchLRC += *auchMsg++; /*Add buffer byte Buffer byte no carry*/
return ((unsigned char)(–((char)uchLRC))) ; /*reture to Binary complement*/
}
```

4. Troubleshooting & Support

Note: Each sensor is well tested in our factory to guarantee their accuracy and with no communication problem. Therefore, you can trust it will work well as long as power supply is stable and wiring is correct.

We would recommend you to use computer with R232/RS485 or SDI-12 to USB converter to test our sensor directly, so that we can exclude other potential failure factors.

4.1 No start up string or garbled code received upon power on

All HongYuv sensors will send ">System Startup\r\n" upon powered on unless specially requested to come without this function.

You can check power consumption of sensor, sensor itself will consume at least 10mA@12VDC.

If problem still exist, please change communication interface converter and try again.

4.2 Garbled code received upon power on

Following possibilities that may lead to this phenomenon:

1. Incorrect data cable wiring.
2. Choosing incorrect baud rate or parity.

4.3 Can't enter configuration mode

All HongYuv sensors will respond ">CONFIGURE MODE\r\n" after receiving command ">*\r\n".

Following possibilities that may lead to this failure:

1. Parity is incorrect, please try to change parity and try again.
2. You didn't input "\r\n" character correctly, "\r\n" is CRLF Carriage-Return Line-Feed, corresponding to HEX (0x0D,0x0A). Some UART software use Shift+Enter to enter those characters.
3. You misuse RS232 as RS485 or vice versa.

4.4 Can’t fetch data by MODBUS RTU request command

Following possibilities that may lead to this phenomenon:
1. Current protocol is not MODBUS RTU.
2. Access registers outside of scope, please reduce quantity of registers in inquiry command.
3. Device address isn’t correct.
4. CRC checksum is incorrect.

4.5 Wind speed and direction is invalid

Please check ultrasonic probes and sampling path.
Wind speed and direction will be unavailable once any of them is blocked.

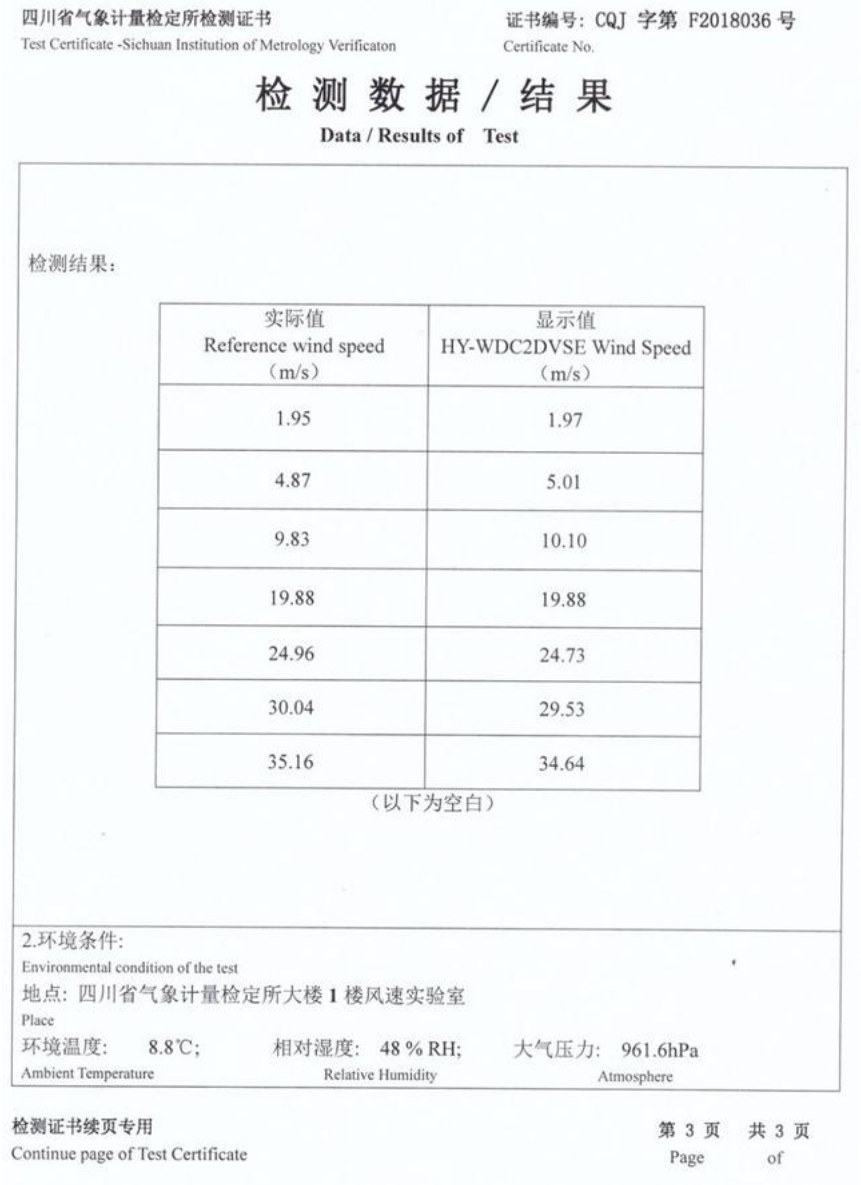
5. Calibration

Each unit is calibrated in our wind-tunnel testing lab prior to shipping.

The anemometer calibration is based on fundamental physical principles and does not change with use. Re-calibration should therefore not be necessary.

However, you may test it within wind tunnel to inspect its accuracy from time to time.

A test report from 3rd party Institution of Meteorological department below for your reference.



四川省气象计量检定所检测证书
Test Certificate -Sichuan Institution of Metrology Verificaton

证书编号：CQJ 字第 F2018036 号
Certificate No.

检测数据／结果
Data / Results of Test

检测结果：

实际值 Reference wind speed （m/s）	显示值 HY-WDC2DVSE Wind Speed （m/s）
1.95	1.97
4.87	5.01
9.83	10.10
19.88	19.88
24.96	24.73
30.04	29.53
35.16	34.64

（以下为空白）

2.环境条件：
Environmental condition of the test
地点：四川省气象计量检定所大楼 1 楼风速实验室
Place
环境温度： 8.8℃； 相对湿度： 48 % RH； 大气压力： 961.6hPa
Ambient Temperature Relative Humidity Atmosphere

检测证书续页专用
Continue page of Test Certificate

第 3 页 共 3 页
Page of

6. Warranty

The Warranty Period of products purchased from HongYuv is 24 months since the date of delivery.

Following conditions are not covered by warranty

- Damage of a product resulting from negligence
- Damage of a product resulting from unauthorized modification of the product
- Damage caused by natural disaster

6.1 How do customers get repair/service under warranty?

99.99% communication issue is caused by incorrect wiring or configuration.

You may contact stefan@hongyuv.com to get further technical guidance by email or remote support by Anydesk www.anydesk.com

You may contact your vendor or HongYuv for malfunctions caused by hardware.

Following common questions will help us understand situation well.

1. When did malfunction happen? Did you change anything before malfunction?
2. What is power consumption of sensor?
3. Is there any startup string upon powered up?
4. Has it ever supplied with over-voltage?

We will reply you within 24 hours after receiving your email.

6.2 How do you charge for repairing?

For quality issue caused by our fault within one/six month since shipment, HongYuv will cover for double-way/single-way shipping charge

For other situation, it’s negotiable.

For malfunction caused by client’s negligent operation:

HongYuv will only charge for repair fee based on our raw cost, and client shall pay shipping charge.