

# LEFOO

## LFS11 LOW AIR VELOCITY TRANSMITTER

Product Operation Manual



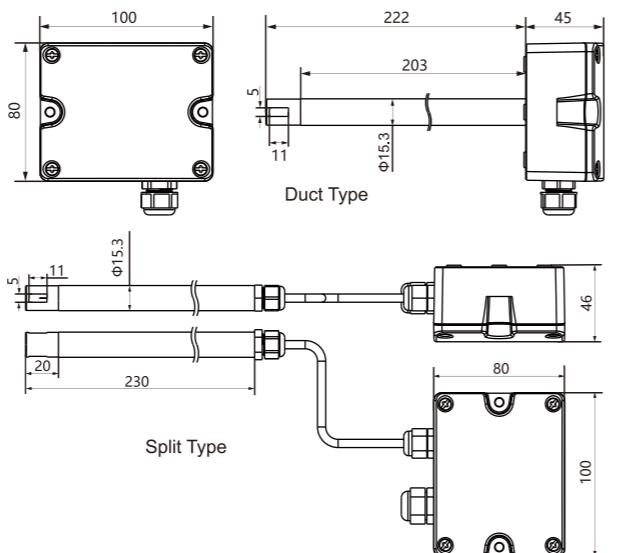
### DESCRIPTION

#### Description

LFS11 Low Air Velocity Transmitter is designed for accurate measurement of very low air velocity, and still has high measurement accuracy in application where air velocity is less than 0.15m/s. LFS11 uses a high-quality hot film sensor manufactured based on MEMS technology,

which has the characteristics of high measurement accuracy, good stability and strong environmental adaptability, which is ideal for laminar flow monitoring and special ventilation applications, mainly used in key environments such as medical institutions, laboratories, and pharmaceutical industry.

#### Dimensions : mm



01

### SPECIFICATION

#### Specification

Working voltage	12-30VDC/24VAC ( $\pm 20\%$ )
Range <sup>①</sup>	0-1m/s, 0-1.5m/s, 0-2m/s optional
Accuracy	$\pm (0.04m/s + 3\% of mv)$ (20°C, 45%RH and 1013hPa)
Resolution	0.01m/s
Output mode	RS485/Modbus, 0~10VDC/4~20mA (Voltage and current dual output) optional
Output load	$\leq 250\Omega$ (Current mode), $\geq 5K\Omega$ (Voltage type)
Working temperature	-10 ~ +60°C
Storage temperature	-20 ~ +80°C
Probe length	210mm (optional)
Display	Optional LCD display with unit display and backlight
Protection	IP65, IP20 (Probe)
Sheathing material	PC, PA6 (Probe)
Certification	ROHS, EU Electrical Safety Standards CE

① Can be selected by jumper

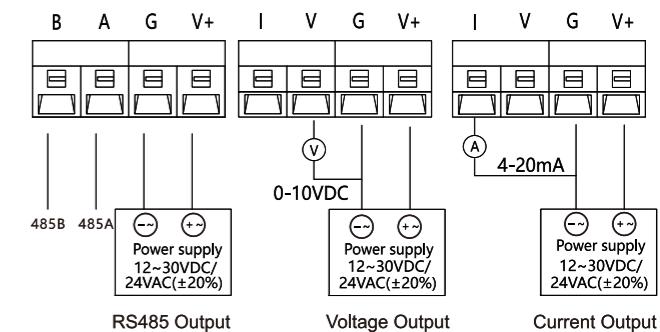
02

### Selection instructions

LFS11-	Low Air Velocity Transmitter		Model
VI	0~10VDC/4~20mA		Output
RS	RS485/Modbus		
1	Duct Type Low Air Velocity Transmitter		Installation method
2	Split Type Low Air Velocity Transmitter		
D	with display		Display
N	without display		

Selection example LFS11-RS1D:  
Duct-type Low Air Velocity Transmitter, output: RS485/Modbus, with display.

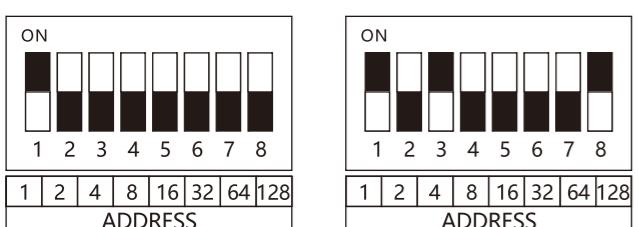
### Wiring instructions



03

### DIP switch settings (RS485 version only)

The 8-digit DIP switch sets the slave address, the address can be set to 1-255, the factory default setting is 01, the setting method is as follows: dial to ON for 1, vice versa for 0, 1~8 digits on the dial panel represent low to high.

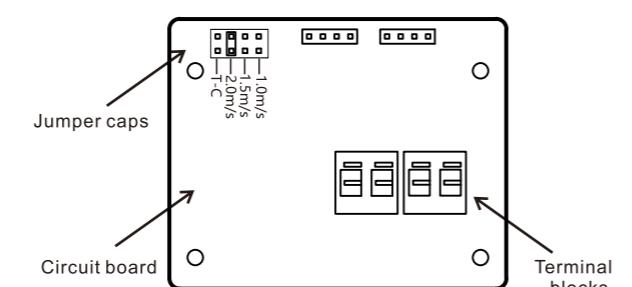


Default address 01, ----- 0x01 (hex) The address is 1+4+128=133, 0x85 (hex)  
0000 0001 (binary) 1000 0101 (binary)

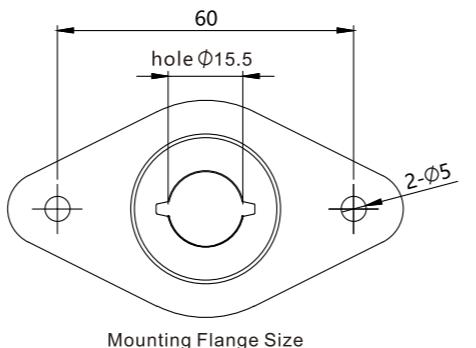
Note 1: In order to prevent product damage, it is recommended to adjust the DIP switch in the event of a power failure, and after changing the address using the DIP switch, it must be powered on again for the change to take effect.

Note 2: The address set by the DIP switch has the highest priority, when modifying the slave address online, all DIP switches must be set to 0 to modify successfully, if necessary, the DIP switch is preferred to modify the address

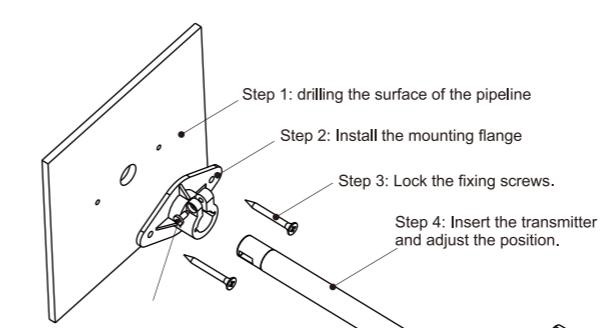
### Range selection



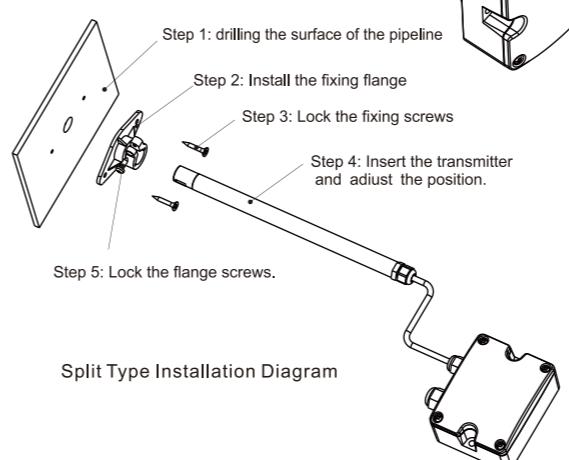
### Installation



04



Duct Type Installation Diagram



Split Type Installation Diagram

### Installation Note:

1. LFS11 recommends that flange accessories be used for installation, and the insertion depth can be adjusted. Fix the mounting flange on the air duct with two screws, and the screws on the flange can lock the inserted probe. The opening of the duct is  $\phi 15.5$ mm. After the probe is installed, the duct should be sealed to avoid air leakage.

2. When installing the air duct, pay special attention to the fact that the air inlet is consistent with the wind speed flow inside the duct, and the sensor is parallel to the wind speed flow.

3. Open the upper cover, connect the power wires and signal wires into the bottom box through the waterproof connector, complete the wiring according to the wiring diagram, and install the upper cover back as it is. Pay attention to the sealing between waterproof joint and bottom box (with sealing ring) and the sealing between upper cover and bottom box (with sealing ring), so that the overall protection level can reach IP65.

4. Do not touch or rub the sensor probe, and do not use any mechanical tools to clean it.

05

06

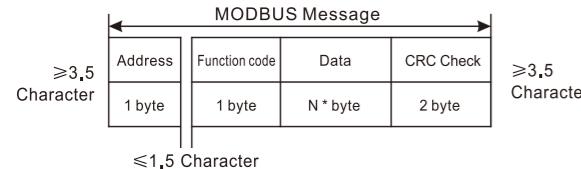
07

## APPENDIX COMMUNICATION PROTOCOL

The protocol runs on the RS485 hardware platform, and can realize remote one-to-many control and signal acquisition through the 485 bus. This communication protocol is implemented in accordance with the ModBus RTU standard protocol.

### 1.Character Format

Start: 1Bit Data: 8Bit Parity: None  
Stop: 1Bit Baud Rate: 9600bps, 19200bps  
In RTU mode, the interval between two characters must be less than 1.5 character times, otherwise the message frame is considered incomplete, and the receiving station discards the message frame. The interval between two message frames is at least 3.5 character times.



### 2.Communication protocol

#### 2.1 Slave ID address

The slave ID address is the identification number of each slave. The default value of this machine is 0x01, which can be modified by modifying the register value. The modification range is 0x01~0xFF.

#### 2.2.Read a single register (function code 0x03)

The master can read the register data of the slave through this function, and One or more registers can be read simultaneously.  
Sequence format:

Master sends read single register sequence					
Slave ID address	function code = 0x03	Register start address	Number of registers need be read	CRC low	CRC high
8Bit	8Bit	16Bit	16Bit	8Bit	8Bit
Slave Normal Acknowledgment Sequence					
Slave ID address	function code = 0x03	Number of data bytes n	Data	CRC low	CRC high
8Bit	8Bit	8Bit	N * 8Bit	8Bit	8Bit
Slave Error Acknowledgment Sequence					
Slave ID address	Error code = 0x83	Abnormal code = 0x02 or 0x03	CRC low	CRC high	
8Bit	8Bit	8Bit	8Bit	8Bit	8Bit

### Communication code example

The host sends the sequence: 01 03 00 01 00 02 95 CB  
Slave ID Function code Register start address Number of registers need be read CRC Check

Slave Normal Acknowledgment Sequence: 01 03 04 B8 52 3F FE D8 FA  
Slave ID Function code Data length Date CRC Check

Slave error response sequence: 01 83 02 C0 F1  
Slave ID Function code Exception code CRC Check

#### 2.3 Write a single register (function code 0x06)

The host can write the register data of the slave through this function, and can only operate on a single register.  
Sequence format:

Master sends write single register sequence					
Slave ID address	function code = 0x06	Register address	The value write in register	CRC low	CRC high
8Bit	8Bit	16Bit	16Bit	8Bit	8Bit
Slave Normal Acknowledgment Sequence					
Slave ID address	function code = 0x06	Register address	The value write in register	CRC low	CRC high
8Bit	8Bit	16Bit	16Bit	8Bit	8Bit
Slave Error Acknowledgment Sequence					
Slave ID address	Error code = 0x86	Abnormal code = 0x02 or 0x03	CRC low	CRC high	
8Bit	8Bit	8Bit	8Bit	8Bit	8Bit

### Communication code example

The host sends the sequence: 01 06 00 03 00 01 B8 0A  
Slave ID Function code Register start address Number of registers need be read CRC Check

Slave normal response sequence: 01 06 00 03 00 01 B8 0A  
Slave ID Function code Register start address Number of registers need be read CRC Check

Slave error response sequence: 01 86 02 C3 A1  
Slave ID Function code Exception code CRC Check

### 3.Register address reference table

Register address	Register definition	Read and write	Specific function description
0x0001	Wind speed data	Read only	The wind speed data is four-byte single-precision floating-point data. For example, the current wind speed is 1.99 (decimal), 3FFEB852(hexadecimal) and its transmission sequence is: B8 52 3F FE
0x0004	Baud rate setting	readable and writable	1=9600bps, 2=19200bps default:1
0x0005	Slave Id <sup>①</sup>	readable and writable	0x01~0xFF can be set, default:0x01

Note<sup>①</sup>: The address set by the DIP switch has the highest priority, when modifying the slave address online, the DIP switch must be set to 0 to be successfully modified, if not necessary, Use the dip switch to modify the address first.

### 4. Exception code analysis

exception code	Wrong reason	Solution
0x02	Read register start address error	Compare the register address reference table to check whether the start address of the read register is readable
0x03	Wrong value written to register	Compare the register address reference table to check whether the value written to the register is in the list