

I²C Data Communication Protocol for Flow Sensors Series V 1.0.3

Contents

1. Overview	2
2. Pin Definitions	2
3. Bus Properties	2
3.1 Bus Idle (A)	2
3.2 Start Bit (B)	2
3.3 Stop Bit (C)	2
3.4 Data Bits (D)	3
3.5 Acknowledge Bit	3
4. Device Address	4
5. Commands	4
5.1 Read the sensor SN (Command Code: 82h)	5
5.2 Read flow rate (Command Code: 83h)	5
5.3 Read pressure (Command Code: A3h)	6
5.4 Read flow rate and pressure (Command Code: 84h)	7
5.5 Read I ² C address (Command Code: 85h).....	8
5.6 Set I ² C address (Command Code: 05h)	8
5.7 Read filter depth (Command Code: 8Bh)	9
5.8 Set filter depth (Command Code: 0Bh).....	9
5.9 Calibrate the offset of flow rate (Command Code: 1Ch).....	9
5.10 Calibrate the offset of pressure (Command Code: 24h)	10
5.11 Read temperature (Command Code: B2h)	10
5.12 Read humidity (Command Code: B3h).....	11
References	11

1. Overview

This document specifies the commonly used I²C data communication protocols for flow sensors series. This document will also review some I²C bus basics, but is not intended to give detailed descriptions on I²C bus specification. For detailed I²C bus specification, please refer to reference ^[1]. For other communication protocols, please contact Siargo.

2. Pin Definitions

Pin definitions reference manual.

3. Bus Properties

Two primary I²C bus properties are as follows:

- The data transfer can be started only when the bus is idle.
- During data transfer, the series data line (SDA) must be kept steady whenever the series clock line (SCL) is high. When the series clock line is high, a change of the series data line from high to low or from low to high will be treated as a start condition or a stop condition.

Fig. 3-1 shows the bus timing diagram:

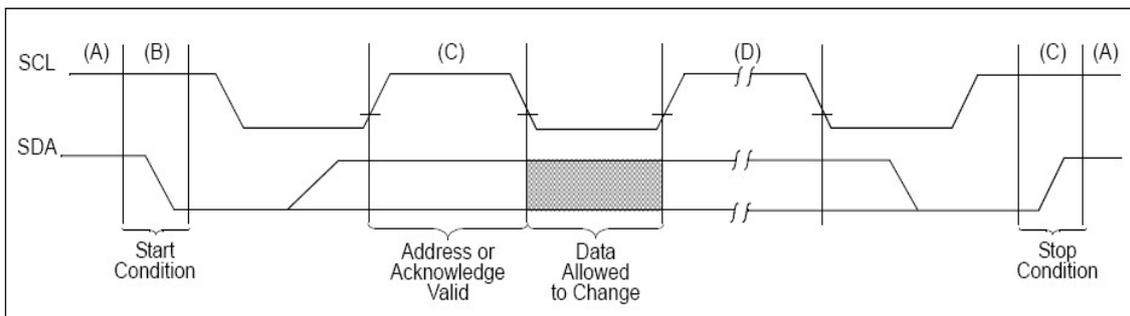


Fig. 3-1: Timing diagram of I²C bus

3.1 Bus Idle (A)

Both the series data line and the series clock line are high.

3.2 Start Condition (B)

When SCL is high, a change of SDA from high to low represents a start condition that initiates data transfer. A start condition must present before any data transfer commands can be executed.

3.3 Stop Condition (C)

When SCL is high, a change of SDA from low to high represents a stop condition that ceases data transfer. All the data transfer commands must be accomplished before a stop condition presents.

3.4 Data (D)

After the start condition, the series data line must be kept steady when the series clock line is high.

The series data line can change during the period when the series clock line is low, and each data bit must correspond to a clock pulse.

Each data transfer will begin with a start condition and cease after a stop condition. Every byte put on the series data line must be 8 bits long. The number of bytes that can be transmitted per transfer is unrestricted. Each byte has to be followed by an Acknowledge bit. The number of data bytes between a start condition and a stop condition will be decided by the bus master.

3.5 Acknowledge Bit

The master is initially in the master transmit mode by sending a start bit followed by the slave address that it wishes to communicate with, which is finally followed by a single bit representing whether it wishes to write(0) to or read(1) from the slave.

If the slave exists on the bus then it will respond with acknowledge (ACK) bit (active low for acknowledged) for that address. The master must provide an extra SCL pulse for each ACK bit.

The master then continues in either transmit or receive mode (according to the read/write bit it sent), and the slave continues in its complementary mode (receive or transmit, respectively).

The address and the data bytes are sent most significant bit first. The start bit is indicated by a high-to-low transition of SDA with SCL high; the stop bit is indicated by a low-to-high transition of SDA with SCL high.

If the master wishes to write into the slave then it repeatedly sends a byte with the slave sending an ACK bit (Fig. 3-2). (In this situation, the master is in the master transmit mode and the slave is in the slave receive mode.)

If the master wishes to read from the slave then it repeatedly receives a byte from the slave, the master sending an ACK bit after every byte but the last one (Fig. 3-2). (In this situation, the master is in the master receive mode and the slave is in the slave transmit mode.)

The master then ends transmission with a stop bit, or it may send another start bit if it wishes to retain control of the bus for another transfer (a "combined message").

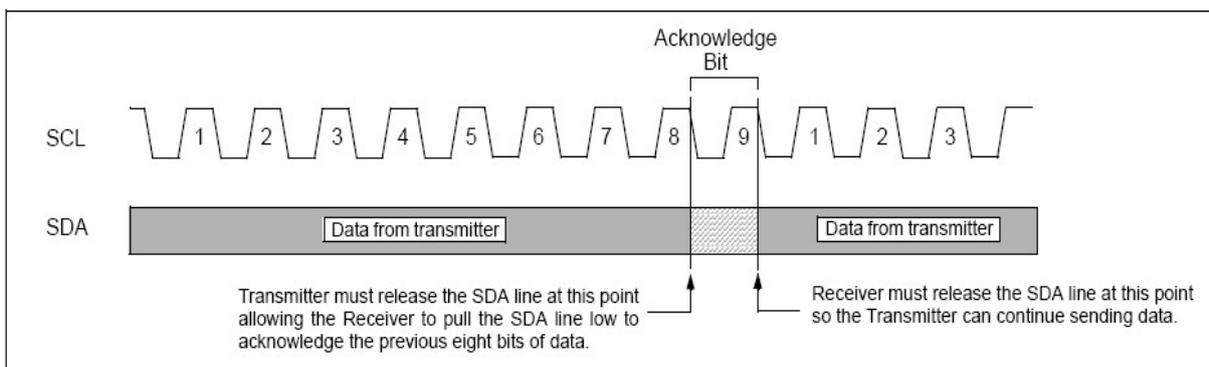


Fig. 3-2: Acknowledge Bit

4. Device Address

The I²C interface of flow sensors adopts the slave-master mode with transfer rate 10kbit/s ~ 100kbit/s. The slave address is always an even number between 02h and FEh, i.e., totally there are 127 available slave addresses. Multiple flow sensors can be configured as a multi-master multi-slave bus for multi-device communication, provided that each sensor (i.e., slave device) has a unique address. The default address is 02H.

The address 00h is reserved address for broadcasting. The user can use this reserved broadcasting address to perform a “write” operation to all the slave devices. However, it is prohibited to use 00h as any slave address. The user can use the default address of a slave or reset its address through the access provided by the broadcasting address.

5. Commands

The sensors series use a communication mode based on the command Interpretation mechanism. The data accesses are accomplished through various commands. A proper command consists of a command code and none, one or multiple command values. If a command has only a command code without any command value, it is called a “pure” command; otherwise, it is called a “compound” command. The most significant byte of the command code is utilized to indicate the command is a “read” command (80h) or a “write” command (00h). An entire command must be continuously transmitted into the sensor under inquiry within a frame (between a start bit and a stop bit); otherwise, an error may occur.

The following commands are supported:

- (1) Read the sensor SN (Command Code: 82h)
- (2) Read flow rate (Command Code: 83h)
- (3) Read pressure (Command Code: A3h)
- (4) Read flow rate and pressure (Command Code: 84h)
- (5) Read I²C address (Command Code: 85h)
- (6) Set I²C address (Command Code: 05h)
- (7) Read filter depth (Command Code: 8Bh)
- (8) Set filter depth (Command Code: 0Bh)
- (9) Calibrate the offset of flow rate (auto zero, Command Code: 1Ch)
- (10) Calibrate the offset of pressure (auto zero, Command Code: 24h)
- (11) Read temperature (Command Code: B2h)
- (12) Read humidity (Command Code: B3h)

5.1 Read the sensor SN (Command Code: 82h)

5.1.1 Procedure

This command is a “pure” command (which has no command values). The procedure is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “82h”, and the slave acknowledges;
- 4) The master sends a re-start bit;
- 5) The master sends “slave address + read bit”, and the slave acknowledges;
- 6) The slave returns the 1st byte of the series number, and the master acknowledges;
- 7) The slave returns the 2nd byte of the series number, and the master acknowledges;
-
- 16) The slave returns the 11th byte of the series number, and the master acknowledges;
- 17) The slave returns the 12th byte of the series number, and the master doesn’t acknowledge;
- 18) The master sends a stop bit.

5.1.2 Explanation

This command is used for enquiring the series number of the flow sensor. The series number is unique for each flow sensor, and consists of 12 alphabets and/or numeric numbers. The series number can serve as the manufacturer ID number for checking its factory default parameter settings.

5.2 Read flow rate (Command Code: 83h)

5.2.1 Procedure

This command is a “pure” command (which has no command values). The procedure is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “83h”, and the slave acknowledges;
- 4) The master sends a re-start bit;
- 5) The master sends “slave address + read bit”, and the slave acknowledges;
- 6) The slave returns the most significant 8 bits, and the master acknowledges;
- 7) The slave returns the next most significant 8 bits, and the master acknowledges;
- 8) The slave returns the next least significant 8 bits, and the master acknowledges;
- 9) The slave returns the least significant 8 bits, and the master doesn’t acknowledge;
- 10) The master sends a stop bit.

5.2.2 Explanation

This command is used for enquiring current instant flow index. The index consists of four 8-bit values, which are combined to give a 32-bit value as follows:

$$\begin{aligned} \text{Flow rate index} = & \quad \{\text{the most significant 8 bits}\} \times 16777216 + \\ & \quad \{\text{the next most significant 8 bits}\} \times 65536 + \\ & \quad \{\text{the next least significant 8 bits}\} \times 256 + \\ & \quad \{\text{the least significant 8 bits}\} \end{aligned}$$

The actual flow value can be calculated as the following.

$$\text{The actual flow value} = \text{the flow index} / 1000$$

5.3 Read pressure (Command Code: A3h)

5.3.1 Procedure

This command is a “pure” command (which has no command values). The procedure is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “A3h”, and the slave acknowledges;
- 4) The master sends a re-start bit;
- 5) The master sends “slave address + read bit”, and the slave acknowledges;
- 6) The slave returns the most significant 8 bits, and the master acknowledges;
- 7) The slave returns the next most significant 8 bits, and the master acknowledges;
- 8) The slave returns the next least significant 8 bits, and the master acknowledges;
- 9) The slave returns the least significant 8 bits, and the master doesn’t acknowledge;
- 10) The master sends a stop bit.

5.3.2 Explanation

This command is used for enquiring current instant pressure index. The index consists of four 8-bit values, which are combined to give a 32-bit value as follows:

$$\begin{aligned} \text{Pressure index} = & \quad \{\text{the most significant 8 bits}\} \times 16777216 + \\ & \quad \{\text{the next most significant 8 bits}\} \times 65536 + \\ & \quad \{\text{the next least significant 8 bits}\} \times 256 + \\ & \quad \{\text{the least significant 8 bits}\} \end{aligned}$$

The actual pressure value can be calculated as the following.

$$\text{The actual pressure value} = \text{the pressure index} / 1000$$

5.4 Read flow rate and pressure (Command Code: 84h)

5.4.1 Procedure

This command is a “pure” command (which has no command values). The procedure is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “84h”, and the slave acknowledges;
- 4) The master sends a re-start bit;
- 5) The master sends “slave address + read bit”, and the slave acknowledges;
- 6) The slave returns the most significant 8 bits of the flow rate index, and the master acknowledges;
- 7) The slave returns the next most significant 8 bits of the flow rate index, and the master acknowledges;
- 8) The slave returns the next least significant 8 bits of the flow rate index, and the master acknowledges;
- 9) The slave returns the least significant 8 bits of the flow rate index, and the master doesn’t acknowledge;
- 10) The slave returns the most significant 8 bits of the pressure index, and the master acknowledges;
- 11) The slave returns the next most significant 8 bits of the pressure index, and the master acknowledges;
- 12) The slave returns the next least significant 8 bits of the pressure index, and the master acknowledges;
- 13) The slave returns the least significant 8 bits of the pressure index, and the master doesn’t acknowledge;
- 14) The master sends a stop bit.

5.4.2 Explanation

This command is used for enquiring both flow rate and pressure. The index consists of eight 8-bit values, the first 4-bit values are flow rate, and last 4-bit values are pressure. See detailed as below:

$$\begin{aligned} \text{Flow rate index} = & \{\text{the most significant 8 bits}\} \times 16777216 + \\ & \{\text{the next most significant 8 bits}\} \times 65536 + \\ & \{\text{the next least significant 8 bits}\} \times 256 + \\ & \{\text{the least significant 8 bits}\} \end{aligned}$$

The actual flow value can be calculated as the following.

$$\text{The actual flow value} = \text{the flow index} / 1000.$$

The flow unit is SLPM.

$$\begin{aligned} \text{Pressure index} = & \{\text{the most significant 8 bits}\} \times 16777216 + \\ & \{\text{the next most significant 8 bits}\} \times 65536 + \\ & \{\text{the next least significant 8 bits}\} \times 256 + \\ & \{\text{the least significant 8 bits}\} \end{aligned}$$

The actual pressure value can be calculated as below.

The actual pressure value = the pressure index / 1000.

The pressure unit is cmH₂O.

5.5 Read I²C Address (Command Code: 85h)

5.5.1 Procedure

This command is a “pure” command (which has no command values). The procedure for reading I²C address of the flow sensor is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “85h”, and the slave acknowledges;
- 4) The master sends a re-start bit;
- 5) The master sends “slave address + read bit”, and the slave acknowledges;
- 6) The slave returns the I²C address, and the master doesn’t acknowledge;
- 7) The master sends a stop bit.

5.5.2 Explanation

This command is used for enquiring the I²C address. For detailed discussions on I²C address, please refer to Section 4.

5.6 Set I²C Address (Command Code: 05h)

5.6.1 Procedure

This command is a “compound” command with one command value, which should be an even number between 02h and FEh, representing I²C address of the flow sensor. The procedure is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “05h”, and the slave acknowledges;
- 4) The master sends the command value (i.e., I²C address 02), and the slave acknowledges;
- 5) The master sends a stop bit.

5.6.2 Explanation

This command is used for setting the I²C address. For detailed discussions on I²C address, please refer to Section 4.

5.7 Read filter depth (Command Code: 8Bh)

5.7.1 Procedure

This command is a “pure” command (which has no command values). The procedure for reading I²C address of the flow sensor is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “8Bh”, and the slave acknowledges;
- 4) The master sends a re-start bit;
- 5) The master sends “slave address + read bit”, and the slave acknowledges;
- 6) The slave returns the filter depth, and the master doesn’t acknowledge;
- 7) The master sends a stop bit.

5.7.2 Explanation

This command is used for enquiring the filter depth.

5.8 Set filter depth (Command Code: 0Bh)

5.8.1 Procedure

This command is a “compound” command with one command value, which should be an even number between 00h and FEh, representing filter depth of the flow sensor. The procedure is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “0Bh”, and the slave acknowledges;
- 4) The master sends the command value (i.e., filter depth 254), and the slave acknowledges;
- 5) The master sends a stop bit.

5.8.2 Explanation

This command is used for setting the filter depth, value between 0 and 254.

5.9 Calibrate the offset of flow rate (Auto Zero, Command Code: 1Ch)

5.9.1 Procedure

This command is a “compound” command. Its command value can be any number since it acts as dummy. The procedure is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “1Ch”, and the slave acknowledges;
- 4) The master sends the command value (e.g., any number), and the slave acknowledges;

5) The master sends a stop bit.

5.9.2 Explanation

This command is used for requesting the flow sensor to calibrate the offset of flow rate automatically and then store it internally. Please ensure that there is NO flow in the flow channel. This offset calibration will reduce the measurement error, and the modification of the offset voltage index will do no harm on the measurement accuracy.

5.10 Calibrate the offset of pressure (Auto Zero, Command Code: 24h)

5.10.1 Procedure

This command is a “compound” command. Its command value can be any number since it acts as dummy. The procedure is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “24h”, and the slave acknowledges;
- 4) The master sends the command value (e.g., any number), and the slave acknowledges;
- 5) The master sends a stop bit.

5.10.2 Explanation

This command is used for requesting the flow sensor to calibrate the offset of pressure automatically and then store it internally. Please ensure that there is NO flow in the flow channel. This offset calibration will reduce the measurement error, and the modification of the offset voltage index will do no harm on the measurement accuracy.

5.11 Read temperature(Command Code: B2h)

5.11.1 Procedure

This command is a “pure” command (which has no command values). The procedure is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “B2h”, and the slave acknowledges;
- 4) The master sends a re-start bit;
- 5) The master sends “slave address + read bit”, and the slave acknowledges;
- 6) The slave returns the most significant 8 bits, and the master acknowledges;
- 7) The slave returns the least significant 8 bits, and the master doesn’t acknowledge;
- 8) The master sends a stop bit.

5.11.2 Explanation

This command is used for enquiring temperature. The temperature consists of two 8-bit values, which are combined to give a 16-bit value as follows:

$$\text{Temperature index} = \{\text{the most significant 8 bits}\} \times 256 + \{\text{the least significant 8 bits}\}$$

The actual temperature value can be calculated as below.

$$\text{The actual temperature value} = \text{the temperature index} / 100.$$

The temperature unit is degC.

5.12 Read humidity (Command Code: B3h)

5.12.1 Procedure

This command is a “pure” command (which has no command values). The procedure is as follows:

- 1) The master sends a start bit;
- 2) The master sends “slave address + write bit”, and the slave acknowledges;
- 3) The master sends the command code “B3h”, and the slave acknowledges;
- 4) The master sends a re-start bit;
- 5) The master sends “slave address + read bit”, and the slave acknowledges;
- 6) The slave returns the most significant 8 bits of the Offset Index, and the master acknowledges;
- 7) The slave returns the least significant 8 bits of the Offset index, and the master doesn’t acknowledge;
- 8) The master sends a stop bit.

5.12.2 Explanation

This command is used for enquiring humidity. The humidity consists of two 8-bit values, which are combined to give a 16-bit value as follows:

$$\text{Humidity index} = \{\text{the most significant 8 bits}\} \times 256 + \{\text{the least significant 8 bits}\}$$

The actual humidity value can be calculated as below.

$$\text{The actual humidity value} = \text{the humidity index} / 100.$$

The humidity unit is RH%.

References:

- [1] I²C-Bus Specification and User Manual, Philips Semiconductors (now NXP Semiconductors). Rev. 03, 19 June 2007.

This document contains confidential information that is proprietary to Siargo Ltd, its subsidiaries and affiliates. No part of this document may be disclosed or conveyed to, used by, or copied for, any third party without prior written permission of Siargo Ltd.