

# LF1100 Series Liquid Flow Sensor I<sup>2</sup>C Interface Protocol

## V 1.0.1

### Content

<b>1. Overview</b> .....	2
<b>2. User Interface</b> .....	2
<b>3. Bus Properties</b> .....	2
3.1 Bus Idle (A) .....	3
3.2 Start Bit (B) .....	3
3.3 Stop Bit (C) .....	3
3.4 Data Bits (D) .....	3
3.5 Acknowledge Bit .....	3
<b>4. Device Address</b> .....	4
<b>5. Commands</b> .....	4
5.1 Retrieve Serial Number (Command Code=82h) .....	5
5.2 Read Instant Flow Rate (Command Code=83h) .....	6
5.3 Read I <sup>2</sup> C Address (Command Code=85h) .....	7
5.4 Set I <sup>2</sup> C Address (Command Code=05h) .....	7
5.5 Read the Sensor's Maximum Flow Rate (Command Code=87h) .....	7
5.6 Retrieve Filter Depth (Command Code=8Bh) .....	8
5.7 Set Filter Depth (Command Code=0Bh) .....	9
Reference .....	9

## 1. Overview

This document specifies the commonly used I<sup>2</sup>C data communication protocols for liquid flow sensor LF1100. This document will also review some I<sup>2</sup>C bus basics, but is not intended to give detailed descriptions on I<sup>2</sup>C bus specification. For detailed I<sup>2</sup>C bus specification, please refer to reference <sup>[1]</sup>.

## 2. User Interface

### 2.1 Pin Definition

The pin layout and definition are shown in Figure 21. and Table 2.1 respectively.

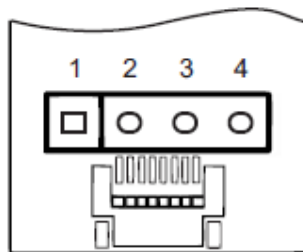


Figure 2.1: Pin layout.

Pin Num.	Definition
1	Vcc (input DC voltage)
2	SCL (I <sup>2</sup> C clock)
3	SDA (I <sup>2</sup> C data)
4	GND (ground)

Table 2.1: Pin definition.

### 2.2 Pin Description

#### Serial Data (SDA) Pin

Serial Data Line (SDA) is bidirectional data line.

#### Serial Clock Line (SCL)

Serial Clock Line (SCL) is used to synchronize all data transfers over the I2C bus.

Both SDA and SCL require pull-up resistors connected with VCC and they are designed on board already.

#### Power supply (VCC)

VCC range:  $5.0V \leq VCC \leq 5.5V$

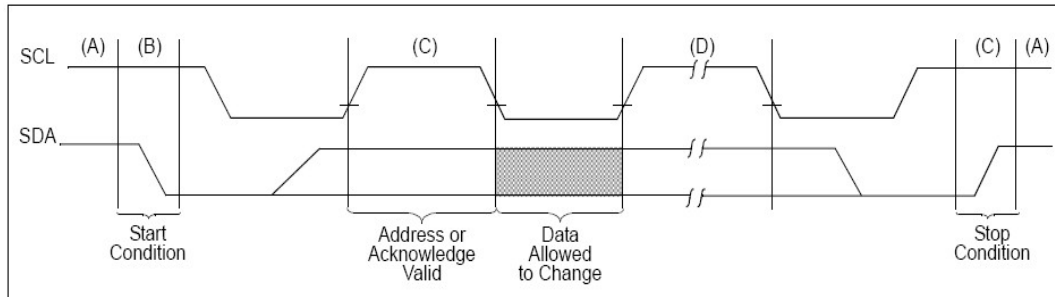
## 3. Bus Properties

I<sup>2</sup>C bus protocols are defined as the following:

- 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.

Following figure shows the bus timing diagram:



### 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

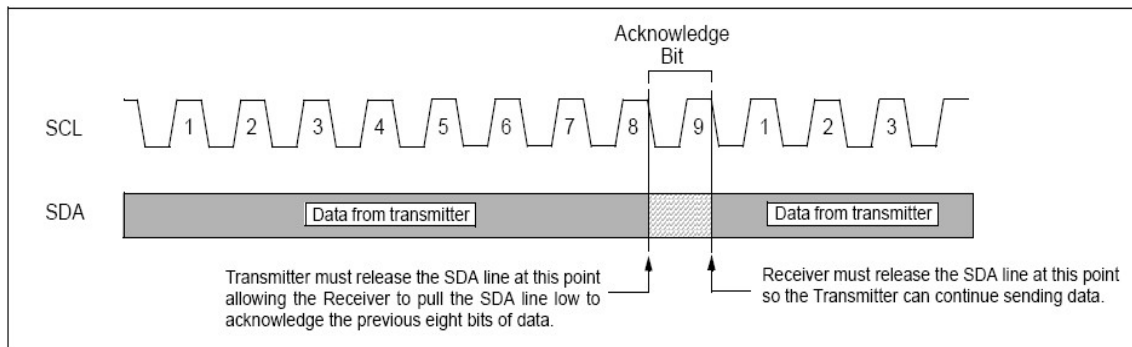
If the slave exists on the bus then it will respond with acknowledge (ACK) bit (active low for acknowledged) for that address. The master must provide an extra SCL pulse for each ACK bit.

The 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 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 (See figure below). (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").



#### 4 Device Address

The I<sup>2</sup>C interface of the LF1100 sensor adopts the slave-master mode with transfer rate 10kbit/s ~ 20kbit/s, making it easy to establish connection and simplify the commands. The slave address is always an even number between 02h and FEh, i.e., totally there are 127 available slave addresses. Multiple sensors can be configured as a multi-master multi-slave bus for multi-device communication, provided that each board (i.e., slave board) 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 boards. 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 sensor uses 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:

**Table 5.1 LF1100 Sensor I<sup>2</sup>C Command List**

Number	Command	Description	Read <sup>1</sup>	Write <sup>2</sup>	Write Verification <sup>3</sup>
1	02H	Set/Retrieve Sensor Serial Number	⊙	●	※
2	03H	Read Instant Flow Rate	⊙		
3	05H	Set/Retrieve Sensor's I <sup>2</sup> C Address	⊙	⊙	
4	07H	Set/Retrieve Maximum Flow Value	⊙	●	※
5	0BH	Set/Retrieve Filter Depth	⊙	⊙	

Note:

- 1) “⊙” stands for the command open to the public
- 2) “●” stands for the command only used by the manufacturer
- 3) “※” stands for the command requires verification of keyword before the write operation, similar to input a protection code.
- 4) If a command is not included in above table, it may cause unknown error.

## 5.1 Retrieve Series Number (Command Code=82h=02h+ 80h)

### 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;
- .....
- 8) The slave returns the 11th byte of the series number, and the master acknowledges;
- 9) The slave returns the 12th byte of the series number, and the master doesn't acknowledge;

10) The master sends a stop bit.

### 5.1.2 Explanation

This command is used for enquiring the series number of the current 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 Instant Flow Rate (Command Code: 83h=03h+80h)

### 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 1st byte of the instant flow rate, and the master acknowledges;
- 7) The slave returns the 2nd byte of the instant flow rate, and the master acknowledges;
- 8) The slave returns the 3rd byte of the instant flow rate, and the master acknowledges;
- 9) The slave returns the 4th byte of the instant flow rate, 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 rate. It is of a 32-bit floating-point number. The flow rate is calculated as follows.

$$\begin{aligned} \text{The flow rate} = & \{ \text{the 1st byte} \} * 16777216 + \\ & \{ \text{the 2nd byte} \} * 65536 + \\ & \{ \text{the 3rd byte} \} * 256 + \\ & \{ \text{the 4th byte} \} / 1000 \end{aligned}$$

The unit of the flow is usually specified in the documents shipped along with the flow sensor..

### 5.3 Read I<sup>2</sup>C Address (Command Code= 85h=05h+80h)

#### 5.3.1 Procedure

This command is a “pure” command (which has no command values). The procedure for reading I<sup>2</sup>C 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 I<sup>2</sup>C address, and the master doesn’t acknowledge;
- 7) The master sends a stop bit.

#### 5.3.2 Explanation

This command is used for enquiring the I<sup>2</sup>C address. For detailed discussions on I<sup>2</sup>C address, please refer to Section 4.

### 5.4 Set I<sup>2</sup>C Address (Command Code= 05h=05h+00h)

#### 5.4.1 Procedure

This command includes command values. The command values have to be an even number between 02h~FEh, which represent the sensor’s I<sup>2</sup>C address. 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 I<sup>2</sup>C address, and the slave acknowledges;
- 5) The master sends a stop bit.

#### 5.4.2 Explanation

This command is used for setting the I<sup>2</sup>C address. For detailed discussions on I<sup>2</sup>C address, please refer to Section 4.

### 5.5 Read the Sensor’s Maximum Flow Rate (Command Cod= 87h=07h+80h)

#### 5.5.1 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 “87h”, 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 positive direction flow rate, and the master acknowledges;
- 7) The slave returns the 2nd byte of the positive direction flow rate, and the master acknowledges;
- 8) The slave returns the 3rd byte of the positive direction flow rate, and the master acknowledges;
- 9) The slave returns the 4th byte of the positive direction flow rate, and the master doesn't acknowledge;
- 10) The master sends a stop bit.

### **5.5.2 Explanation**

This command is used for enquiring maximum flow rate of the sensor. It returns 4-byte hexadecimal number, which represents a 32 bit floating-point number as the maximum flow rate. The default maximum flow rate is 1000 mL/hr.

### **5.6 Retrieve Filter Depth (Command Code: 8Bh=0Bh+80h)**

#### **5.6.1 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 “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 does not acknowledge;



- 7) The master sends a stop bit.

### **5.6.2 Explanation**

This command is used for retrieving the sensor's filter depth. The value of the filter depth is between 00h~FFh. When filter depth is equal or less than 2, no filtering is processed. The larger the filter depth value is, the more stable the measurement (reading of flow rate from I<sup>2</sup>C) is, but the response time is slower. The filter depth will not affect the response time of the analog output.

## **5.7 Set Filter Depth (Command Code: 0Bh=0Bh+00h)**

### **5.7.1 Procedure**

This command includes command values. The command values have to be between 00h~FFh. 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 "filter depth", and the slave acknowledges;
- 5) The master sends a stop bit.

### **5.7.2 Explanation**

This command is used for setting the sensor's filter depth.

### **Reference:**

1. THE I2C-BUS SPECIFICATION VERSION 2.1, Philips Semiconductors, JANUARY 2000.