OMRON

MEMS Differential Pressure Sensor

D6F-PH

User's Manual

MEMS Differential Pressure Sensor

A288-E1-02

Contents

1. Outline

This User's Manual is intended to demonstrate how to use and interface with Omron's D6F-PH MEMS differential pressure sensor. Please note that this document is intended to supplement the datasheet, which should be referenced when using the sensor.

2. Structure

Fig.1 shows the internal cross-section view of the D6F-PH MEMS differential pressure sensor. Air will flow from one inlet and out the other passing over the MEMS flow chip surface. The MEMS chip is able to measure the airflow as air passes over the chip.

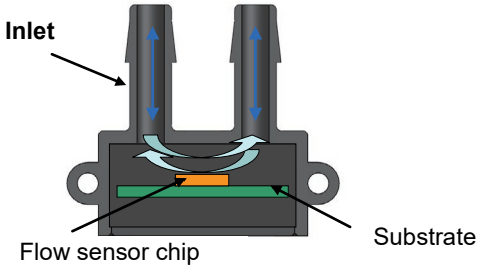


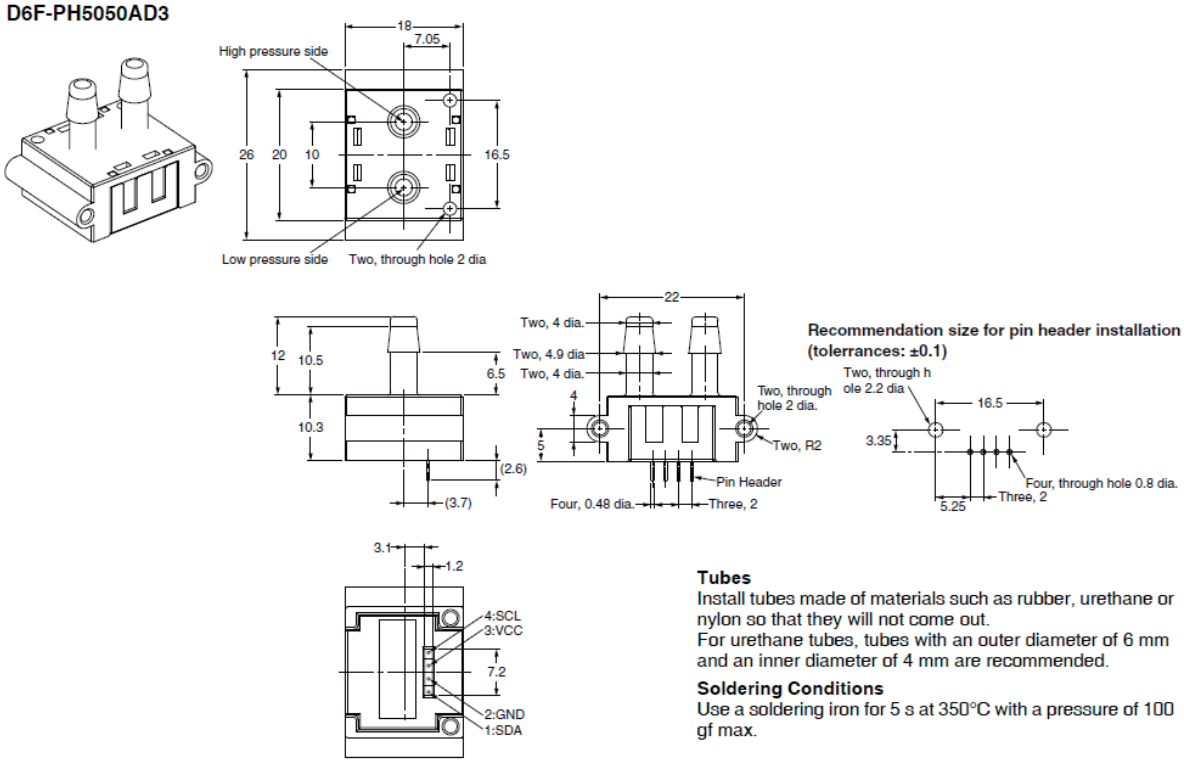
Fig.1 the internal cross-section view of D6F-PH MEMS differential pressure sensor

3. Dimensions

● Lead Terminals

CAD Data

D6F-PH0025AD1
D6F-PH0505AD3
D6F-PH5050AD3



Tubes

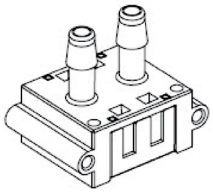
Install tubes made of materials such as rubber, urethane or nylon so that they will not come out.
For urethane tubes, tubes with an outer diameter of 6 mm and an inner diameter of 4 mm are recommended.

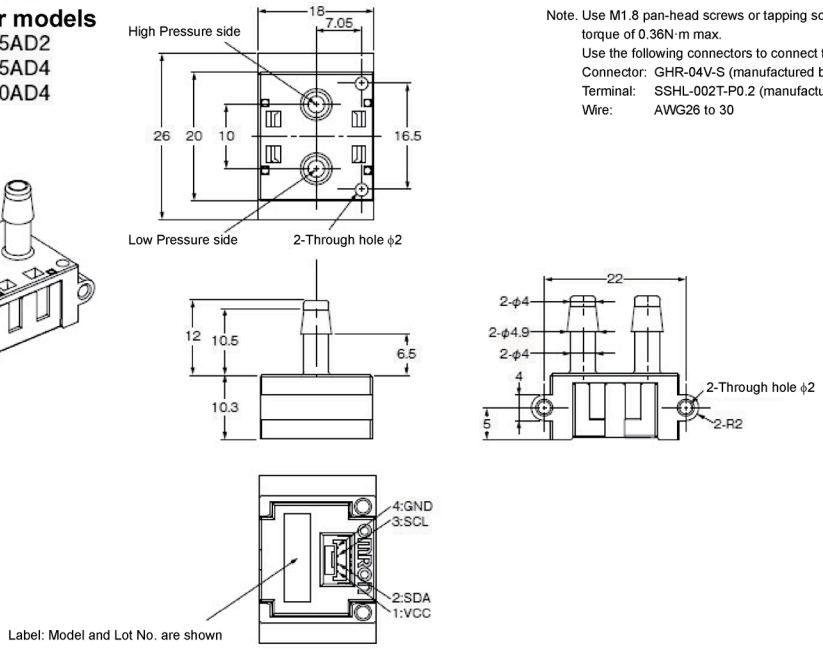
Soldering Conditions

Use a soldering iron for 5 s at 350°C with a pressure of 100 gf max.

Fig.2 Outline dimensions of D6F-PH0505AD3 / D6F-PH0025AD1 / D6F-PH5050AD3

Connector models
D6F-PH0025AD2
D6F-PH0505AD4
D6F-PH5050AD4

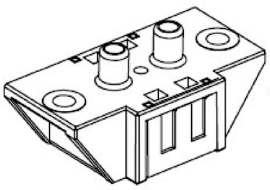




Note. Use M1.8 pan-head screws or tapping screws to mount, with a tightening torque of 0.36N·m max.
Use the following connectors to connect this product.
Connector: GHR-04V-S (manufactured by JST Connector)
Terminal: SSHL-002T-P0.2 (manufactured by JST Connector)
Wire: AWG26 to 30

Fig.3 Outline dimensions of D6F-PH0505AD4 / D6F-PH0025AD2 / D6F-PH5050AD4

D6F-PH0025AMD2
D6F-PH0505AMD4
D6F-PH5050AMD4



Note 1. Use M3 round head machine screws or tapping screws as mounting screws, with a tightening torque of 1.0N•m max.
Note 2. Outer diameter of screw head and washer must be 6mm max.
Note 3. The sealing part of the input port includes a groove of ϕ7.0 and a P4 O-ring. (JISB 2401 compliant)
Note 4. Use the following connectors to connect this product.
Connector: GHR-04V-S (manufactured by JST Connector)
Terminal: SSHL-002T-P0.2 (manufactured by JST Connector)
Wire: AWG26 to 30

Fig.4 Outline dimensions of D6F-PH0505AMD4 / D6F-PH0025AMD2 / D6F-PH5050AMD4

4. Principle of pressure detection

By using a thermal mass flow sensor, Omron's D6F-PH MEMS differential pressure sensor can detect fine changes in differential pressure.

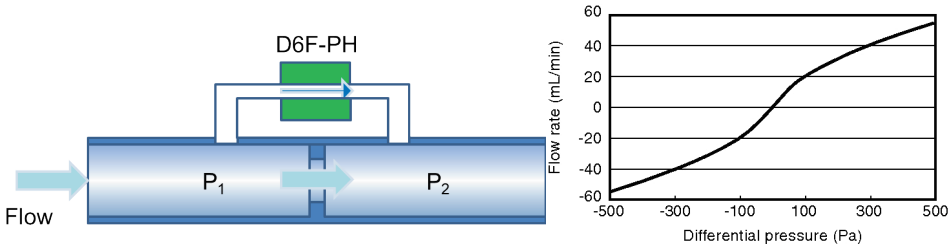
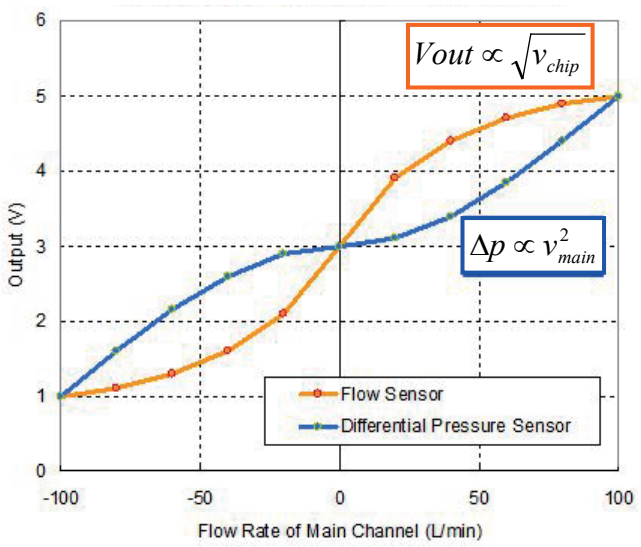


Fig.5 Principle of differential pressure sensor (a) and relationship between flow rate and differential pressure (b)

5. Features of Product

By using the thermal mass flow method (heat flow type), OMRON's D6F-PH MEMS differential pressure sensor can detect changes in the low pressure range with higher sensitivity as compared with the membrane type differential pressure sensor.



Thermal mass flow method
The output of Omron's sensor is proportional to the square root of the gas flow velocity through the sensor chip surface.

Membrane method
The output of a conventional sensor is proportional to the square of the gas flow velocity through the main channel.

Orange:
thermal mass flow method

Blue:
membrane method

Fig. 6 Comparison between membrane method and thermal mass flow method

Table 1. D6F-PH lineup

Differential pressure range	Port type	Connection	Model
0-250Pa	Bamboo joint	Leard terminal (Board Mount)	D6F-PH0025AD1
		Connector	D6F-PH0025AD2
	Manifold	Connector	D6F-PH0025AMD2
±50Pa	Bamboo joint	Leard terminal (Board Mount)	D6F-PH0505AD3
		Connector	D6F-PH0505AD4
	Manifold	Connector	D6F-PH0505AMD4
±500Pa	Bamboo joint	Leard terminal (Board Mount)	D6F-PH5050AD3
		Connector	D6F-PH5050AD4
	Manifold	Connector	D6F-PH5050AMD4

Table 2. Specifications of D6F-PH□□□□

Item	Contents				
	Min	Typ	Max	Unit	Note
Differential pressure range	-50	-	50	Pa	D6F-PH0505AD3-□ D6F-PH0505AD4 D6F-PH0505AMD4
	0	-	250	Pa	D6F-PH0025AD1-□ D6F-PH0025AD2 D6F-PH0025AMD2
	-500	-	500	Pa	D6F-PH5050AD3-□ D6F-PH5050AD4 D6F-PH5050AMD4
Resolution	-	12	-	bit	
Zero point tolerance (Note)	-0.2	-	+0.2	Pa	
Span tolerance (Note)	-3	-	+3	%R.D.	
Span shift due to temperature variation	-0.5	-	+0.5	%R.D.	With respect to a change of 10°C
Response time	-	33	50	msec	12bit Resolution
Operating temperature	-20	-	80	degC	without freezing and condensation
Storage temperature	-40	-	80	degC	without freezing and condensation
Operating humidity	35	-	85	%RH	without freezing and condensation
Storage humidity	35	-	85	%RH	without freezing and condensation
Supply voltage	2.3	3.3	3.6	VDC	
Current consumption	-	-	6	mA	Vcc=3.3V, 25degC
Frequency of SCL	-	-	400	kHz	FAST Mode

(Note) Span accuracy and zero point accuracy are the independence errors, and are not satisfied at the same time.

6. Usage

6-1. Flow path connection method

By installing an orifice in the main channel where differential pressure is measured, a small pressure change is generated front and back the orifice. D6F-PH is connected to the bypass flow path from the pressure port which is installed frond and back the orifice. By this configuration, D6F-PH is able to detect fine pressure changes.

(1) For bamboo joint (D6F-PH0505AD3 / D6F-PH0025AD1 / D6F-PH5050AD3
D6F-PH0505AD4 / D6F-PH0025AD2 / D6F-PH5050AD4)

Here, the inner diameter of the bypass tube which is connected to the D6F-PH is 4.0[mm] and its length is 800[mm] or less. Set the tube as straight as possible.

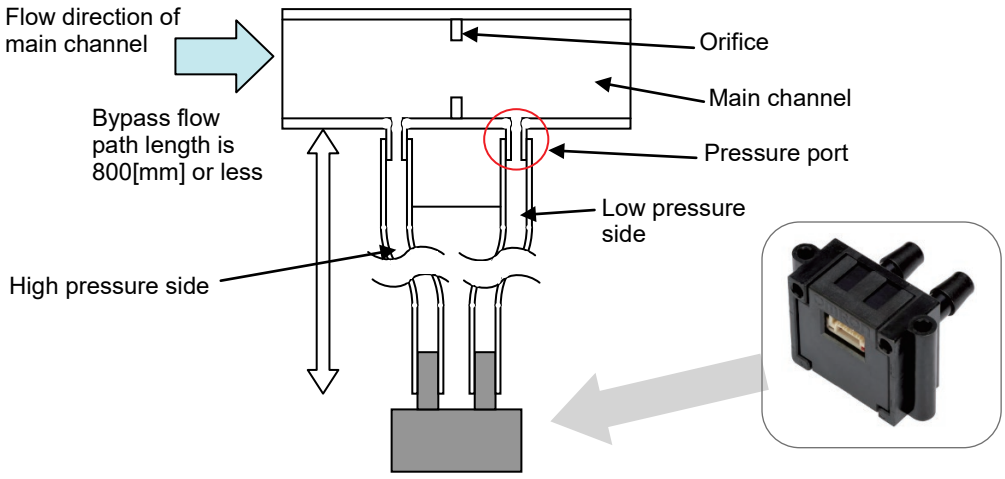


Fig.7 Connection of D6F-PH (bamboo joint) to the main flow path

(2) For manifold (D6F-PH0505AMD4 / D6F-PH0025AMD2 / D6F-PH5050AMD4)
Seal the connection to D6F-PH with an O-ring, etc.

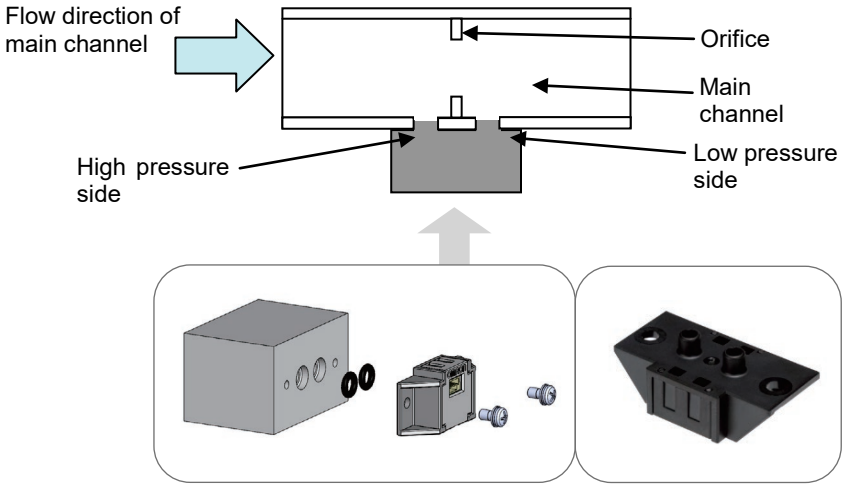


Fig.8 Connection of D6F-PH (manifold) to the main flow path

6-2. Flow Path Design

Design and evaluate the flow path connected to D6F-PH using the following procedure. The numerical values shown below are theoretical. The actual values must be evaluated in your system.

[STEP 1]
Determine the requirements below.

- Maximum flow rate in main flow path: F_{max}
- Pressure range (allowable pressure loss at the bypass): P_{max}
- Inner diameter of main flow path: D

F_{max} depends on the performance of your pump.
The pressure loss applied to the bypass at F_{max} is P_{max}.

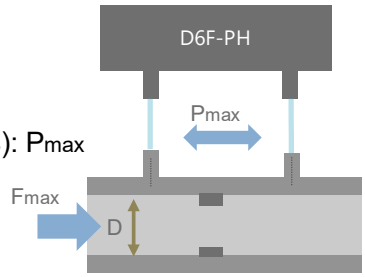


Fig.9 Bypass structure

[STEP 2]
The pressure range P_{max} determines the D6F-PH model. Select D6F-PH.

- For 0 to 250 Pa D6F-PH0025AD1 / D6F-PH0025AD2 / D6F-PH0025AMD2
- For -50 to 50 Pa D6F-PH0505AD3 / D6F-PH0505AD4 / D6F-PH0505AMD4
- For -500 to 500 Pa D6F-PH5050AD3 / D6F-PH5050AD4 / D6F-PH5050AMD4

[STEP 3]
Determine the orifice diameter d (mm) from Fmax and D using the table below.

Table 3. Orifice diameter d (D6F-PH0025AD1 / D6F-PH0025AD2 / D6F-PH0025AMD2)

Flow rate:	(L/min)	10	20	30	50	100	150
Fmax	(m3/h)	0.6	1.2	1.8	3.0	6.0	9.0
D (mm)	10	3.61	5.04	6.05	7.40	8.92	9.44
	20	3.62	5.12	6.26	8.04	11.16	13.28
	30	3.62	5.12	6.27	8.09	11.39	13.86
	40	3.62	5.12	6.27	8.09	11.43	13.97
	50	3.62	5.12	6.27	8.09	11.44	14.00

Table 4. Orifice diameter d (D6F-PH0505AD3 / D6F-PH0505AD4 / D6F-PH0505AMD4)

Flow rate:	(L/min)	5	10	20	30	40	50
Fmax	(m3/h)	0.6	1.2	1.8	3.0	6.0	9.0
D (mm)	10	3.81	5.30	7.11	8.13	8.72	9.09
	20	3.83	5.41	7.62	9.27	10.61	11.73
	30	3.83	5.41	7.65	9.36	10.78	12.03
	40	3.83	5.41	7.65	9.37	10.81	12.08
	50	3.83	5.41	7.66	9.37	10.82	12.10

Table 5. Orifice diameter d (D6F-PH5050AD3 / D6F-PH5050AD4 / D6F-PH5050AMD4)

Flow rate:	(L/min)	10	20	30	50	100	150
Fmax	(m3/h)	0.6	1.2	1.8	3.0	6.0	9.0
D (mm)	10	3.04	4.27	5.18	6.48	8.25	9.01
	20	3.04	4.30	5.27	6.79	9.50	11.46
	30	3.04	4.31	5.27	6.80	9.60	11.72
	40	3.04	4.31	5.27	6.81	9.62	11.77
	50	3.04	4.31	5.27	6.81	9.62	11.78

Example) The pump is controlled so that the maximum flow rate of the main flow path is 100 l/min.
In order to have the pressure loss of 500 Pa in bypass at 100 l/min, in the case that the inner diameter D of the main flow path is 30 mm, the orifice diameter d is 9.6 mm. (See Table 5)

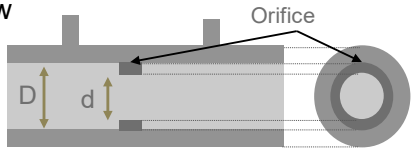


Fig.10 Orifice

[STEP 4]
Design and fabricate the flow path using the orifice diameter d determined in STEP 3.
a = 5 mm
b = 5 mm
c = approx. 1 to 2 mm
L = 10 D or higher
(The inlet/outlet length L is the distance that must be straight. If sufficient inlet/outlet length can't be designed, set L to a symmetrical value.)

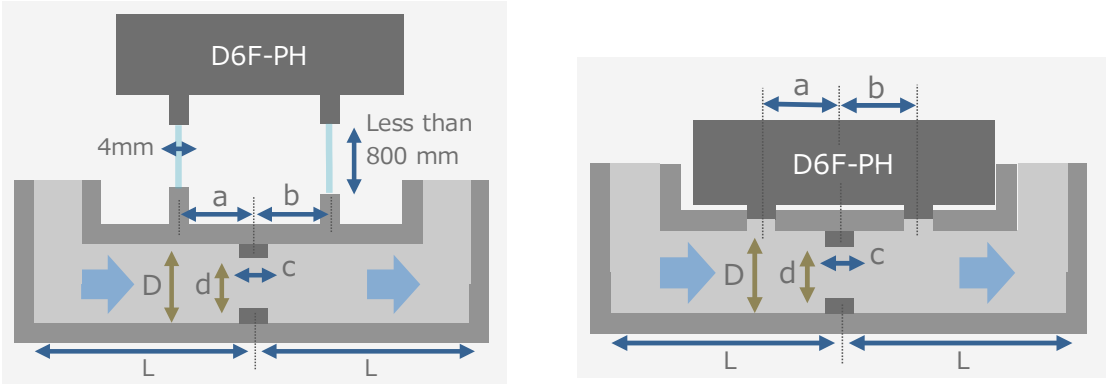


Fig. 11 Example of flow path structure (left: bamboo joint, right: manifold)

[STEP 5]
Make an actual measurement using the designed and fabricated flow path to confirm that D6F-PH outputs the correct pressure value.
If the correct value is not output, change the dimensional values of the flow path design to obtain the correct value.

(Example)
Check the output value of D6F-PH where the flow rate is F_{max}.
(Ideally, when the flow rate is F_{max}, the output pressure value of D6F-PH is around P_{max}.)
If the pressure value of D6F-PH has unacceptable variations in your system, review the ratio of orifice diameter d to main flow path diameter D.

- If the output of D6F-PH is larger than P_{max}, increase the d/D value.
- If the output of D6F-PH is smaller than P_{max}, decrease the d/D value.

Fabricate a new flow path and perform re-evaluation.

6-3. Electrical connection method of D6F-PH

For the I2C output, the D6F-PH will require a pull-up resistor to each clock line (SCL) and data line (SDA). A pull-up resistor of 2.2[kΩ] (recommended value) should be implemented between the Vcc as shown in the figure below.
In addition, please adjust the pull-up resistor's value depending on the transfer rate of SCL and the I2C wire length.

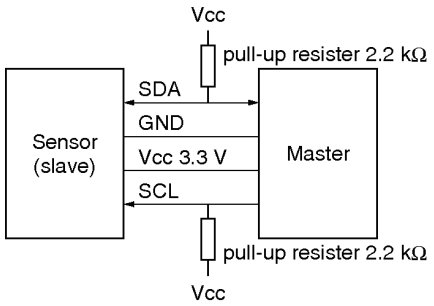


Fig.12 Electrical connection method of D6F-PH

* Precautions when connecting the flow sensor
An error may occur during communications due to the effect of the noise of the customer's environment. In such a case, check the following points, and correct the communications error.

(1) Checking the communications speed
This product supports SCL frequencies up to 400 kHz, but if communication errors tend to occur, OMRON recommends setting the SCL to a lower frequency.

(2) Checking the wiring cable
If the length of the cable used to connect the customer-controlled microcontroller and an OMRON flow sensor is long, the effect of noise may be more prominent. In such a case, OMRON recommends using a shielded cable.

(3) Checking the pull-up resistor's value
A pull-up resistor is necessary for I2C communications of this product. Although the recommended resistor value is 2.2[kΩ], please select the optimum resistor value according to the length of the cable used to connect the customer-controlled microcontroller and the OMRON flow sensor. A communications error is judged unless an ACK is returned from the sensor side. The ACK response time is equal to one clock cycle of SCL. If an ACK response is not returned after this time has elapsed, it results in a communications error. In such a case, turn off the device.

7. Communication specifications

7-1. Outline of I2C Interface

Table 6. Basic specifications of I2C communication

Models		D6F-PH0025AD1 D6F-PH0505AD3 D6F-PH5050AD3 D6F-PH0025AD2 D6F-PH0505AD4 D6F-PH5050AD4 D6F-PH0025AMD2 D6F-PH0505AMD4 D6F-PH5050AMD4	D6F-PH0025AD1-1 D6F-PH0505AD3-1 D6F-PH5050AD3-1	D6F-PH0025AD1-2 D6F-PH0505AD3-2 D6F-PH5050AD3-2	D6F-PH0025AD1-3 D6F-PH0505AD3-3 D6F-PH5050AD3-3
Communication method		I2C			
Slave Address	HEX	0x6C	0x6D	0x6E	0x6F
	BIN (7bit)	110_1100	110_1101	110_1110	110_1111
Communications frequency		Max. 400kHz			
Signals	SCL	Serial Clock			
	SDA	Data Signal			

Table 7. I2C slave address is expressed as below. (Example of 0x6C)

Bit	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
								R/W
Value	1	1	0	1	1	0	0	1/0

Write: Set LSB of slave address to “0” to form D8h (1101_1000b).
Read: Set LSB of slave address to “1” to form D9h (1101_1001b).

7-2. Operation Procedure

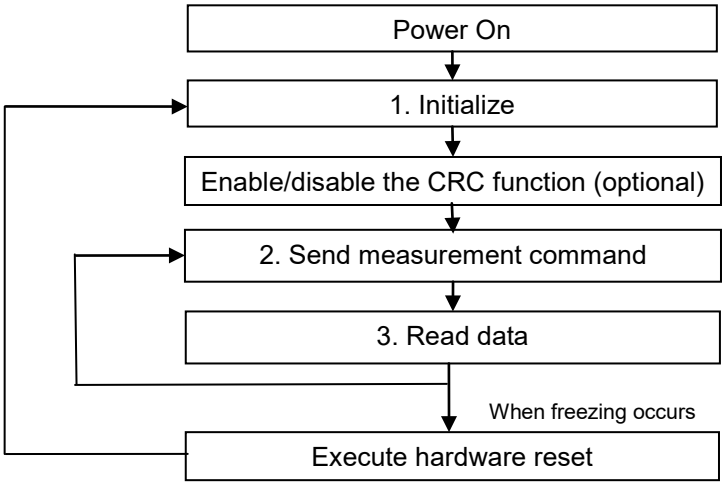


Fig.13 Flowchart of Sensor operation

・Communication time

Item	Sign	Remarks
Response time	α	α ≥ 33 ms
Sampling interval	β	β > α

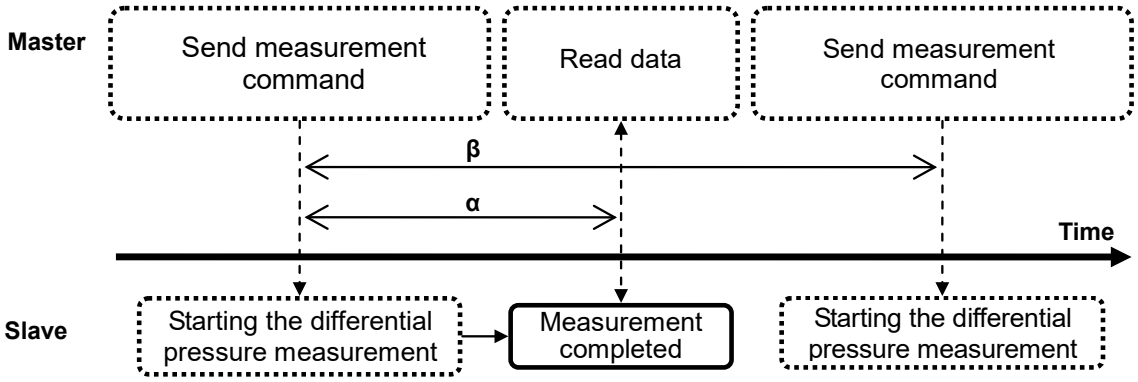


Fig.14 Sequence of D6F-PH measurement

1.Initialize

Perform this operation 200 μs after the power is turned on.

I2C command: Write 00h in "Initialization Register (0Bh)".

START	Slave Address	ACK	Access Address	ACK	Write Data	ACK	STOP
S	D8h (6Ch (7b)+ 0)	A	0Bh	A	00h	A	P

2. Send measurement command

MCU mode is executed (differential pressure measurement is performed) along with various settings. By reading this register after writing, it's possible to read the state of the MUX selected in the MCU. After running the process, MS bit will be set to "0".
Do not read or write to the device while the MCU is executing. Access after 33 msec has passed.

I2C command: Write "06h" to "Sensor Control Register (D040h)" via "Access Address Register (00h)" (MS=1 & MCU_on)

START	Slave Address	ACK	Access Address	ACK
S	D8h (6Ch (7b)+ 0)	A	00h	A

Reg Address H	ACK	Reg Address L	ACK	Serial Ctrl	ACK	Write Data	ACK	STOP
D0h	A	40h	A	18h	A	06h	A	P

3. Read data

After the initialization process, the first data is not normal data and thus must be ignored.

I2C command: Write reading request command "2Ch" (2-byte read access) of "Compensated Flow rate Register (D051h, D052h)" via "Access Address Register (00h)".

START	Slave Address	ACK	Access Address	ACK
S	D8h (6Ch (7b)+ 0)	A	00h	A

Reg Address H	ACK	Reg Address L	ACK	Serial Ctrl	ACK	STOP
D0h	A	51h	A	2Ch	A	P

I2C command: Read 2 bytes of the flow data with "Read Buffer Register (07h, 08h)".

START	Slave Address	ACK	Access Address	ACK
S	D8h (6Ch (7b)+ 0)	A	07h	A

Re-Start	Slave Address	ACK	Read Data H	ACK	Read Data L	ACK	STOP
RS	D9h (6Ch (7b)+ 1)	A	xxh	A	xxh	NA	P

The read 16-bit data is unsigned. Use this as Pv and convert it to differential pressure (Pa) using the following formula.

- ○ If differential pressure measurement range is ±50 [Pa]
  *Dp[Pa] = (Pv - 1024)/60000*RANGE – RANGE/2* (RANGE: 100)
- ○ If differential pressure measurement range is ±500 [Pa]
  *Dp[Pa] = (Pv - 1024)/60000*RANGE – RANGE/2* (RANGE: 1000)
- ○ If differential pressure measurement range is 0-250 [Pa]
  *Dp[Pa] = (Pv - 1024)/60000*RANGE* (RANGE: 250)

4. Enable CRC function (optional)

Set "1" in bit[1] of "CRC Control Register". See 7.4 for detail reading sequence of CRC.

I2C command: Write "02h"(CRC_EN = 1) to "CRC Control Register (D049h)" via "Access Address Register (00h)"

START	Slave Address	ACK	Access Address	ACK	Reg Address H	ACK	Reg Address L	ACK
S	D8h (6Ch (7b)+ 0)	A	00h	A	D0h	A	49h	A

Serial Ctrl	ACK	Write Data	ACK	STOP
18h	A	02h	A	P

5. Execute hardware reset

Set "1" in bit[7] of “Power Sequence Register". After hardware reset, bit[7] is cleared to "0" automatically.

I2C command: Write "80h"(Hard_Reset = 1) to "Power Sequence Register (0Dh)".

START	Slave Address	ACK	Access Address	ACK	Write Data	ACK	STOP
S	D8h (6Ch (7b)+ 0)	A	0Dh	A	80h	A	P

7-3. Detail Description of Registers

In D6F-PH, communication is performed via “Configuration Register".

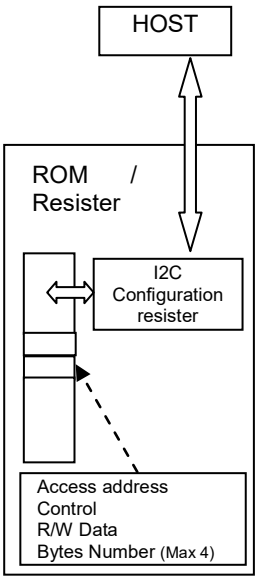


Table 8. Configuration Register Map

Configuration Address	Register name	Note
00h	Access Address 1 (Upper byte)	Upper byte of first “Access Address”
01h	Access Address 2 (Lower byte)	Lower byte of first “Access Address”
02h	Serial Control	Write / Read access control
03h	Write Buffer 0	Data to be written at "Access Address”
04h	Write Buffer 1	Data to be written at "Access Address”+ 1
05h	Write Buffer 2	Data to be written at "Access Address”+ 2
06h	Write Buffer 3	Data to be written at "Access Address”+ 3
07h	Read Buffer 0	Data read from "Access Address”
08h	Read Buffer 1	Data read from "Access Address”+ 1
09h	Read Buffer 2	Data read from "Access Address”+ 2
0Ah	Read Buffer 3	Data read from "Access Address”+ 3
0Bh	Initialization Register	Initialization after power ON
0Dh	Power Sequence	Hardware reset control

Upper byte : bit[15:8] of 16bit data, Lower byte : bit[7:0] of 16bit data

Fig.15 Configuration overview

7-3-1. Access Address Registers (00h – 01h)

"Access Address Register" is used to access "Internal Register". The address of "Access Address Register" is auto increment in multiple byte data transfer.
Therefore, set start address of "Internal register” in “Access Address Register” in multiple byte data transfer.

Table 9. Access Address Register

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
00h	A15	A14	A13	A12	A11	A10	A9	A8
01h	A7	A6	A5	A4	A3	A2	A1	A0

"Internal Register" address consists of 16 bits. In order to access "Internal Register", its address should be written in "Access Address Register" (00h and 01h).

Table 10. Internal Registers Map

Address	Register name	Descriptions
D040h	SENS_CTRL	Sensor Control Register
D046h	FLAGS	Flag Register
D049h	INT_CTRL	CRC Control Register
D051h	COMP_DATA1_H	Compensated Flow Rate Register
D052h	COMP_DATA1_L	
D061h	TMP_H	Internal Temperature Register
D062h	TMP_L	

Table 11. SENS_CTRL (D040h) : Sensor Control Register

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
D040h						MS	DV_PWR[1]	DV_PWR[0]
Write Access	None	None	None	None	None	Host & MCU	Host & MCU	Host & MCU
Default	0	0	0	0	0	0	0	0

DV_PWR[1 : 0] --- Main Device power mode setting

0 0 = Standby: All blocks are powered down.

1 0 = MCU on: Turn on MCU block. Analog part and memory part are powered on and MCU clock is running. This register should not be changed during measurement.

MS --- MCU start

0 = Stop: Sequences are stopped and each block is turned off.

1 = Start: MCU clock is started and MCU mode is executed.

Table 12. FLAGS (D046h) : Flag Register

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
D046h					OS1		HV1	SV
Write Access	None	None	None	None	Host & MCU	None	Host & MCU	Host & MCU
Default				0	0	0	0	0

SV --- Supply Voltage Flag: 0 = Within specification. 1 = Out of specification.

HV1 --- Heater Voltage Flag: 0 = Within specification. 1 = Out of specification.

OS1 --- Open Sensor Flag: 0 = Sensor is connected. 1 = Sensor isn't connected.

Table 13. INT_CTRL (D049h): CRC Control Register

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
D049h							CRC_EN	
Write Access	NONE	NONE	NONE	NONE	NONE	NONE	Host & MCU	NONE
Default	0	0	0	0	0	0	1	0

CRC function is selected (CRC supports 2-byte reading. See 7.4 for details.)

0 = CRC check calculation disable 1 = CRC check calculation enable

Table 14. Compensated Flow Rate Register (D051h, D052h), Internal Temperature Register (D061h, D062h)

Address	Registers Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Description
D051h	COMP_DATA1_H	DATA<15:8>								Differential pressure data
D052h	COMP_DATA1_L	DATA<7:0>								
D061h	TMP_H	DATA<15:8>								Internal temperature data
D062h	TMP_L	DATA<7:0>								

Differential Pressure data [D051h – D052h] (Unsigned: Uint16)

- ±50[Pa] or ±500[Pa] type (RANGE: 100 or 1000, Pv: value at D051h-D052h)

  *Dp[Pa] = (Pv - 1024)/60000*RANGE – RANGE/2*

- 0-250[Pa] type (RANGE: 250, Pv: value of D051h – D052h)

  *Dp[Pa] = (Pv - 1024)/60000*RANGE*

Temperature data [D061h – D062h] (Signed:Int16)

Tv [°C] = (Rv – 10214) / 37.39 (Rv: value of D061h – D062h)

Note: Temperature data is for reference ONLY. Its accuracy doesn't guaranteed.

7-3-2. Serial Control Register (02h)

Table 15. Serial Control Register (02h)

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
02h	D_byte_cnt [3]	D_byte_cnt [2]	D_byte_cnt [1]	D_byte_cnt [0]	Req	R_WZ	Acc_ctl2 [1]	Acc_ctl2 [0]

- Acc_ctl2 [1:0] – Access Control bits
  0 0 = 16bits address (A15-A0) access (internal ROM and registers)
  0 1 = 8bits address (A7-A0) access (MCU internal 256 byte dual port RAM.)
  1 0 = reserved
  1 1 = reserved
- R_WZ – Read or Write access select bit
  0 = Write Access
  1 = Read Access
- Req- Request bit
  0 = The previous request is done
  1 = New request. After the serial bus bridge controller finishes a request, it will clear Req to 0. For "Write" requests, the bridge controller moves the data in "Write Buffer" to the location pointed by "Access Address". For "Read" requests, the bridge controller stores the read data into "Read Buffer".
- D_byte_cnt3 [3:0]
  Transfer data byte count. It only supports 1, 2, 3, 4 data byte transfer.

7-3-3. Write Buffer Registers (03h – 06h)

Four write buffers for writing values to "Internal Register". They can be written in the following 2 ways. The following is an example of writing data [0] to the Address = A [15:0] register of "Internal Register". The 18h below means writing 18h (newly writing 1 byte) with the "Serial Control Register" (02h).

(Method 1)

Burst write the following 5 bytes.

・00h、A[15:8]、A[7:0]、18h、data[0]

* The first 00h is “Access Address Register” (00h).

(Method 2)

Write in 2-byte increments in order.

・00h、A[15:8]
・01h、A[7:0]
・03h、data [0]
・02h、18h

* When "Serial Control Register" (02h) is read, if bit [3] is "0", it means that the write access is completed.

7-3-4. Read Buffer Registers (07h – 0Ah)

Four read buffers for reading the value of the "Internal Register". They can be read in the following 2 methods. The following is an example of reading 1-byte register value of Address = A [15:0] "Internal Register". The "1Ch" below means writing "1Ch" (newly writing 1 byte) with "Serial Control Register" (02h).

(Method 1)

Burst write the following 4 bytes and issue the read request.

・00h、A[15:8]、A[7:0]、1Ch

After the read request is completed, "Read Buffer Register 07h" will be read.

* When the read request is completed, the request bit of "Serial Control Register (02h)" will be cleared to "0".

(Method 2)

Write in 2-byte increments in order.

・00h、A[15:8]

・01h、A[7:0]

・02h、1Ch

After the read request is completed, "Read Buffer Register (07h)" will be read.

7-3-5. Initialization Register (0Bh)

After turning the power on, it is necessary to write 00h to “Initialization Register (0Bh)”. (In order to load the trimming data of NVM)

7-3-6. Power Sequence Register (0Dh)

Table 16. Power Sequence Register (0Dh)

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0Dh	Hard_ Reset	ADC_ state	ADC_ state	ADC_ state	Pwr_seq_ state5	Pwr_seq_ state5	Pwr_seq_ state5	Pwr_seq_ state5

Table 17. Details of Power Sequence Register

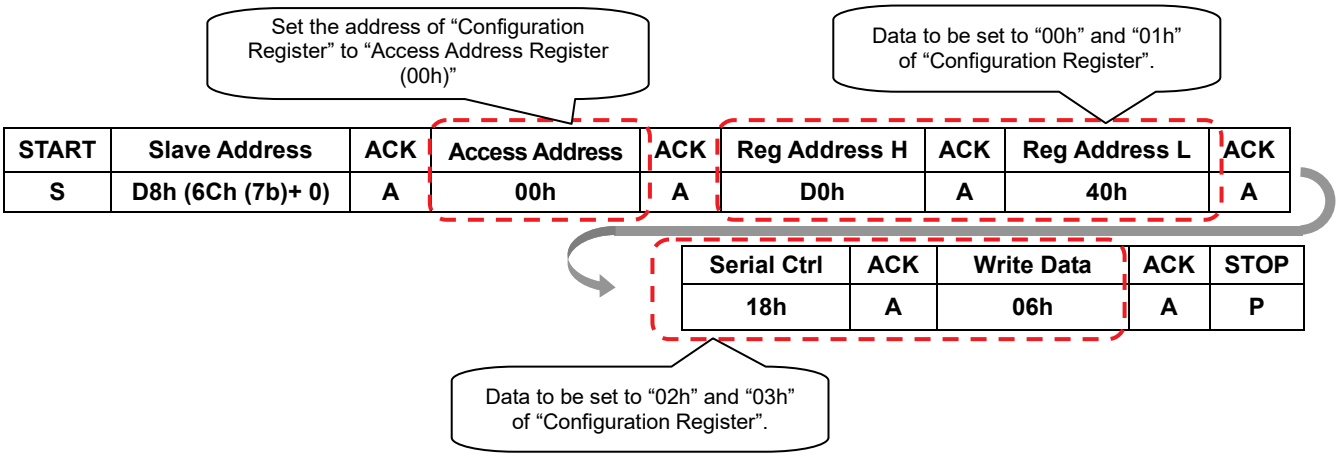
Bit	Name	R/W	Description
[3:0]	Pwr_seq_state5	R	Indicates the state of the power sequence. h0(0000b): Idle (After the initialization or when the power supply is reset) h2(0010b): Active (Writing 06h to D040h) h9 (1001b): Execute (calculating)
[6:4]	ADC_state	R	State in which ADC is controlled
[7]	Hard_Reset	R/W	1-> Hardware reset is performed (Automatically cleared after execution) 0-> Hardware reset is not performed

When a hardware reset is performed, the hardware reset bit is automatically cleared to “0” after the execution of reset, the internal register returns to the default value, and the internal trimming value is reloaded from the non-volatile memory. This hardware reset function is similar to the power reset function.

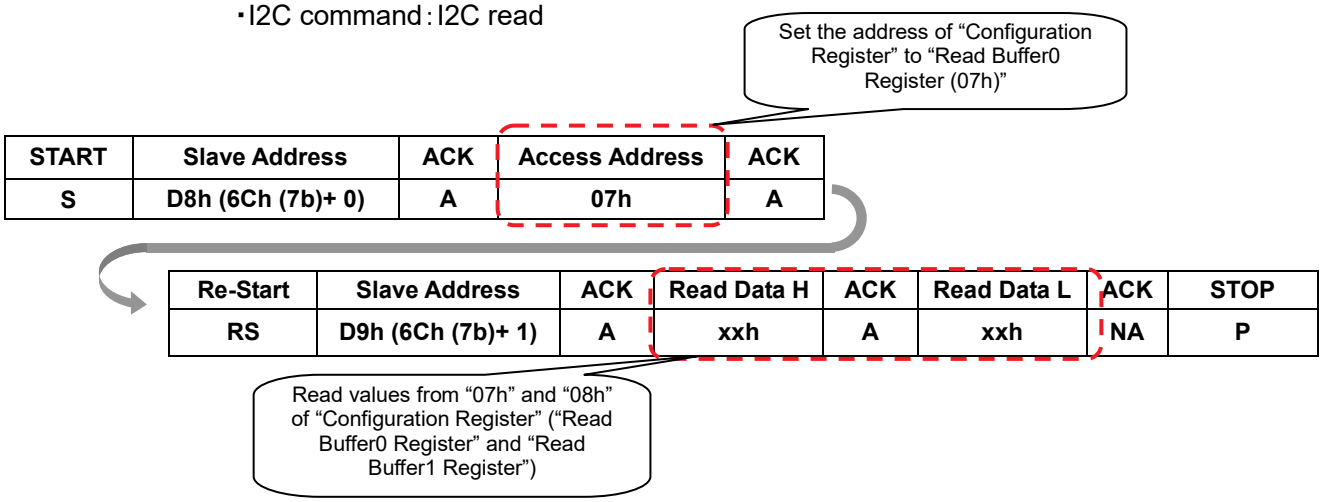
When using hardware reset, set bits “Bit0 – Bit6” to “0”.

7-3-7. Example of I2C Access Commands

・I2C command: I2C write



・I2C command：I2C read



7-4. CRC

・CRC Overview

The CRC is used as an error detection method in a data communication. Our flow sensor uses the CRC8 polynomial x^8 + x^5 + x^4 + 1. The following is an example of I2C access 2 byte read using CRC function.

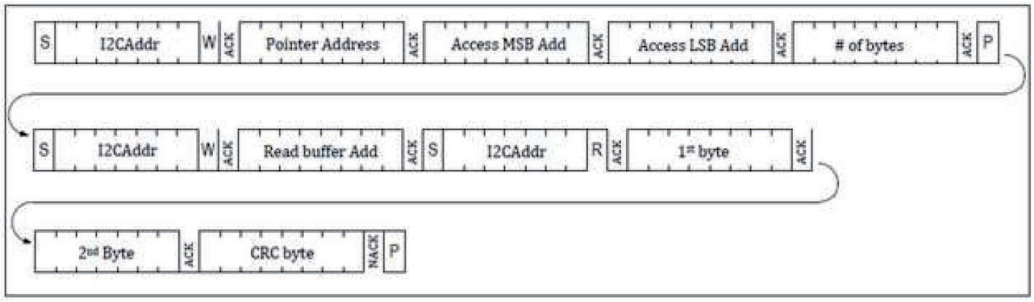


Fig.16 Example of 2byte read with CRC

・Bit unit CRC-8 calculation method

1. The data bit sequence will be aligned in a line.
2. The polynomial bit string will be aligned under the line of the data bit sequence.
3. If the data bit above the leftmost the polynomial bit sequence is 0, the polynomial bit sequence is shifted one bit to the right. If the data bit above the leftmost polynomial bit sequence is 1, the data bit and the polynomial bit are calculated by XOR. Then the polynomial bit sequence are shifted one bit to the right.
4. Steps 1-3 are repeated until the polynomial bit sequence reaches the right end of the data bit sequence.

The following example shows how to calculate the CRC byte based on XOR calculation.

	hex	bin
1st Byte of data	04h	00000100
2nd Byte of data	02h	00000010
Polynomial (x^8+ x^5+ x^4+ 1)	131h	100110001
CRC-byte checksum	225h	11100001

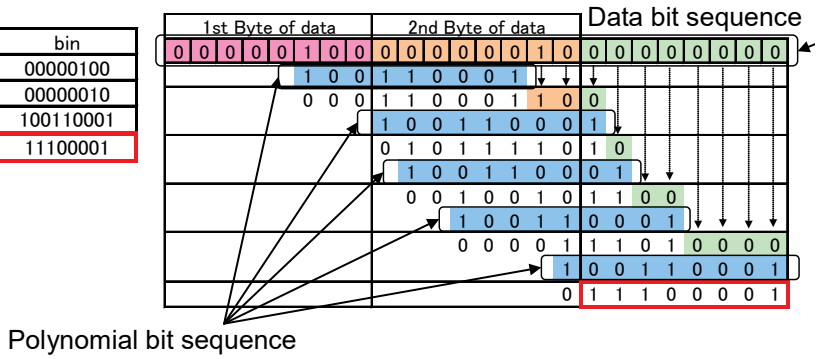


Fig.17 XOR operation example of CRC-8

8. Development tools

Two types of sample code are available as software development support tools.

1. Sample code for Raspberry Pi
2. Sample code for Arduino

The sample codes for Raspberry Pi and Arduino can be used together with the OMRON evaluation board. The OMRON evaluation board supports the following 3 types of platforms. Evaluation can be performed easily by connecting differential pressure sensor D6F-PH, evaluation board, and harness to the platforms.
The evaluation board is compatible with all D6F-PHs, but please note that the harness for connecting the evaluation board to the D6F-PH differs depending on the model of the D6F-PH.

Evaluation Board URL: (https://components.omron.com/sensor/evaluation-board/2jcie)

Table 18. Evaluation Board Map

Platform	Evaluation Board	Sample Code	Harness for Connection (Between evaluation board and D6F-PH)	Harness compatible differential pressure sensor model
Raspberry Pi *1	2JCIE-EV01-RP1 	https://github.com/omron-devhub/d6f-2jciev01-raspberrypi	2JCIE-HARNESS-02 *4	D6F-PH0025AD1
				D6F-PH0505AD3
				D6F-PH5050AD3
Arduino *2	2JCIE-EV01-AR1 	https://github.com/omron-devhub/d6f-2jcieev01-arduino	2JCIE-HARNESS-03 *5	D6F-PH0025AD2
				D6F-PH0505AD4
				D6F-PH5050AD4
ESP32 Feather *3	2JCIE-EV01-FT1 	https://github.com/omron-devhub/d6f-2jcieev01-arduino		D6F-PH0025AMD2
				D6F-PH0505AMD4
				D6F-PH5050AMD4

*1. Raspberry Pi is a registered trademark of the Raspberry Pi Foundation.
*2. Arduino is a registered trademark of Arduino LLC and Arduino SRL.
*3. Feather is a registered trademark of Adafruit Industries LLC.
*4. Model 2JCIE-HARNESS-02 has a connector on one side and a lead wire on the other. The lead wire needs to be connected to D6F-PH before use.
*5. Model 2JCIE-HARNESS-03 is a double-sided connector. Both the D6F-PH and evaluation board can be easily linked with a connector.

The sample code can be used without the evaluation board. However, the customer needs to provide wiring to the sensor.
The sample code is for evaluation purposes only, and OMRON does not guarantee its function.

8-1. Sample Code for Raspberry Pi

Sample code for Raspberry Pi can be found in the URL below.
Github URL: https://github.com/omron-devhub/d6f-2jcieev01-raspberrypi

--For differential pressure measurement range 0-250 Pa
https://github.com/omron-devhub/d6f-2jcieev01-raspberrypi/blob/master/d6f-ph0025.c
--For differential pressure measurement range ±50 [Pa]
https://github.com/omron-devhub/d6f-2jcieev01-raspberrypi/blob/master/d6f-ph0505.c
--For differential pressure measurement range ±500 [Pa]
https://github.com/omron-devhub/d6f-2jcieev01-raspberrypi/blob/master/d6f-ph5050.c

The structure of the sample code is as follows.

```
/** <!-- main - Differential pressure sensor {{{1 -->
 * 1. read and convert sensor.
 * 2. output results, format is: [Pa]
 */
int main() {
    // 1. Initialize sensor (0Bh, 00h)
    i2c_write_reg16(D6F_ADDR, 0x0B00, NULL, 0);
    delay(900);
    while(1){
        // 2. Trigger getting data (00h, D0h, 40h, 18h, 06h)
        uint8_t send0[] = {0x40, 0x18, 0x06};
        i2c_write_reg16(D6F_ADDR, 0x00D0, send0, 3);

        delay(50);  // wait 50ms

        // 3. Read data (00h, D0h, 51h, 2Ch) (07h)
        uint8_t send1[] = {0x51, 0x2C};
        i2c_write_reg16(D6F_ADDR, 0x00D0, send1, 2);
        uint8_t rbuf[2];
        uint32_t ret = i2c_read_reg8(D6F_ADDR, 0x07, rbuf, 2);

        uint16_t rd_flow = conv8us_u16_be(rbuf);
        float flow_rate;

        // calculation for +/-50[Pa] range
        flow_rate = ((float)rd_flow - 1024.0) * 100.0 / 60000.0 - 50.0;

        printf("%6.2f", flow_rate);
        printf(" [Pa]\n");
    }
}
```

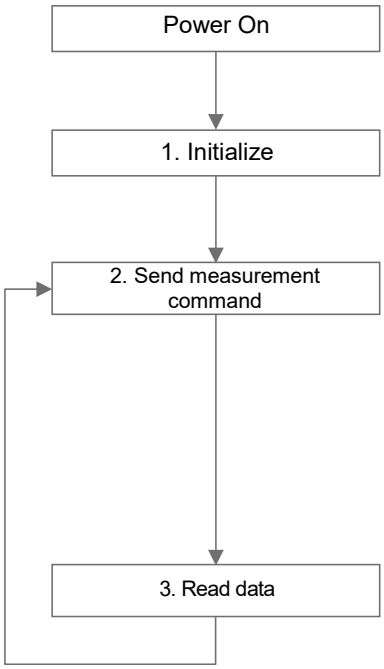


Fig.18 Sample code structure for Raspberry Pi

The operation procedure of the sample code for Raspberry Pi is as follows.

(1)Connect D6F-PH, harness and OMRON evaluation board (2JCIE-EV01-RP1) to the Raspberry Pi

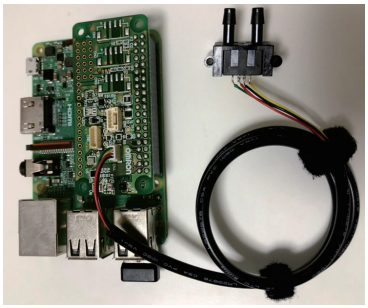


Fig.19 Set-up

(2) Enable the I2C

Launch Raspberry Pi, open "Preferences" > "Raspberry Pi Configuration" from the Start menu, "Enable" the I2C, then restart.

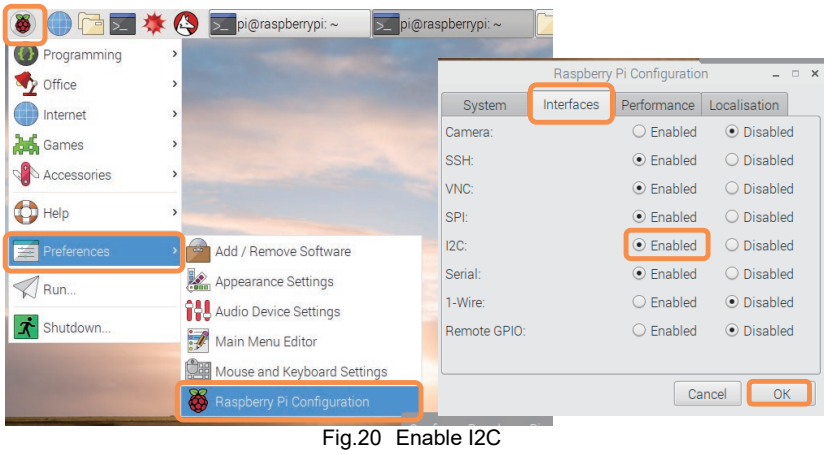


Fig.20 Enable I2C

(3) Download sample code

Open Github from the URL below and download Zip file.

GitHub URL: https://github.com/omron-devhub/d6f-2jcieev01-raspberrypi

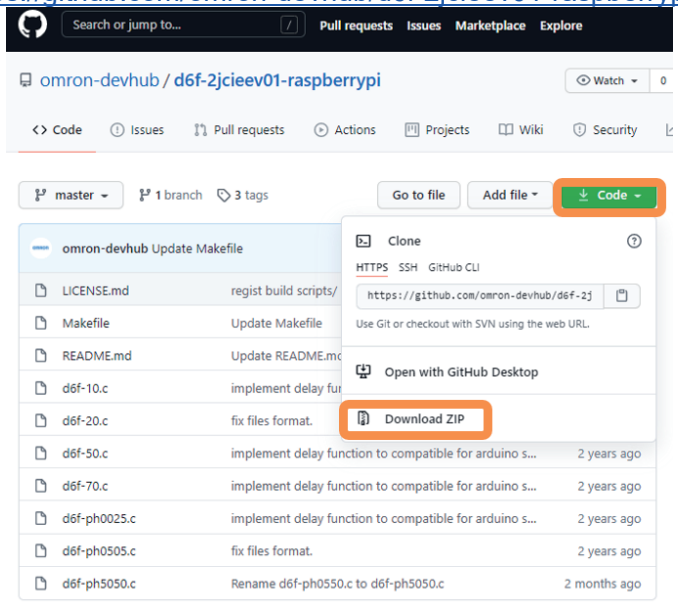


Fig.21 Download sample code

Alternatively, open the Terminal and execute the following command:

```
~$ git clone https://github.com/omron-devhub/d6f-2jcieev01-raspberrypi
```

Alternatively, after downloading the Zip file on another PC with the Internet connection, move the zip file to the Raspberry Pi using a USB memory, etc.

(4) Make file

Open the Terminal and execute the following command.

```
pi@raspberrypi:~ $ cd Downloads/
pi@raspberrypi:~/Downloads $ unzip d6f-2jcieev01-raspberrypi-master.zip
pi@raspberrypi:~/Downloads $ cd d6f-2jcieev01-raspberrypi-master/
pi@raspberrypi:~/Downloads/d6f-2jcieev01-raspberrypi-master $ make all
```

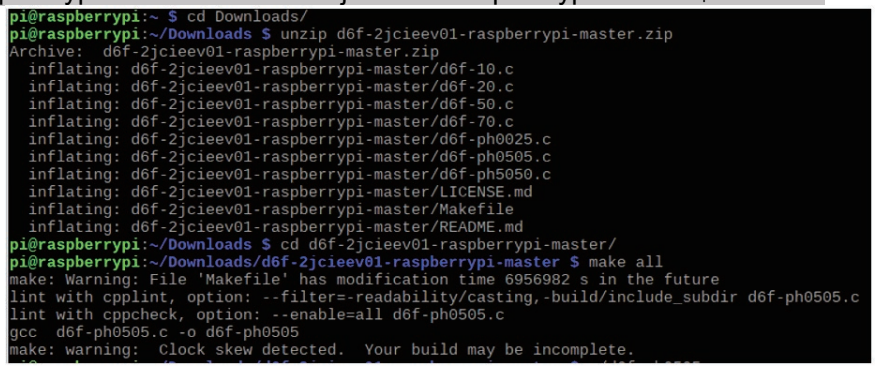


Fig.22 Make file

(5) Run the file

Execute the following command to obtain the data.

--If differential pressure measurement range is 0-250 Pa

```
pi@raspberrypi:~/Downloads/d6f-2jcieev01-raspberrypi-master $ ./d6f-ph0025
```

--If differential pressure measurement range is ±50 Pa

```
pi@raspberrypi:~/Downloads/d6f-2jcieev01-raspberrypi-master $ ./d6f-ph0505
```

--If differential pressure measurement range is ±500 Pa

```
pi@raspberrypi:~/Downloads/d6f-2jcieev01-raspberrypi-master $ ./d6f-ph5050
```

(To stop obtaining data, press the "Ctrl" and "C" keys at the same time.)

Fig.23 Run the file

8-2. Sample code for Arduino

Sample code for Arduino can be found in the URL below.
Github URL: https://github.com/omron-devhub/d6f-2jcieev01-arduino
--For differential pressure measurement range 0-250 Pa
https://github.com/omron-devhub/d6f-2jcieev01-arduino/tree/master/examples/d6f-ph0025
--For differential pressure measurement range ±50 [Pa]
https://github.com/omron-devhub/d6f-2jcieev01-arduino/tree/master/examples/d6f-ph0505
--For differential pressure measurement range ±500 [Pa]
https://github.com/omron-devhub/d6f-2jcieev01-arduino/tree/master/examples/d6f-ph5050

The structure of the sample code is as follows.

```
void setup() {
    Serial.begin(115200);
    Serial.println("peripherals: I2C");
    Wire.begin();  // i2c master

    Serial.println("sensor: Differential pressure Sensor");
    delay(32);

    // 1. Initialize sensor (0Bh, 00h)
    i2c_write_reg16(D6F_ADDR, 0x0B00, NULL, 0);
}

/** <!-- loop - Differential pressure sensor {{{1 -->
 * 1. read and convert sensor.
 * 2. output results, format is: [Pa]
 */
void loop() {
    delay(900);

    // 2. Trigger getting data (00h, D0h, 40h, 18h, 06h)
    uint8_t send0[] = {0x40, 0x18, 0x06};
    i2c_write_reg16(D6F_ADDR, 0x00D0, send0, 3);

    delay(50);  // wait 50ms

    // 3. Read data (00h, D0h, 51h, 2Ch) (07h)
    uint8_t send1[] = {0x51, 0x2C};
    i2c_write_reg16(D6F_ADDR, 0x00D0, send1, 2);
    uint8_t rbuf[2];
    if (i2c_read_reg8(D6F_ADDR, 0x07, rbuf, 2)) {  // read from [07h]
        return;
    }
    uint16_t rd_flow = conv8us_u16_be(rbuf);
    float flow_rate;

    // calculation for +/-50[Pa] range
    flow_rate = ((float)rd_flow - 1024.0) * 100.0 / 60000.0 - 50.0;

    Serial.print(flow_rate, 2);  // print converted flow rate
    Serial.println(" [Pa]");
}
```

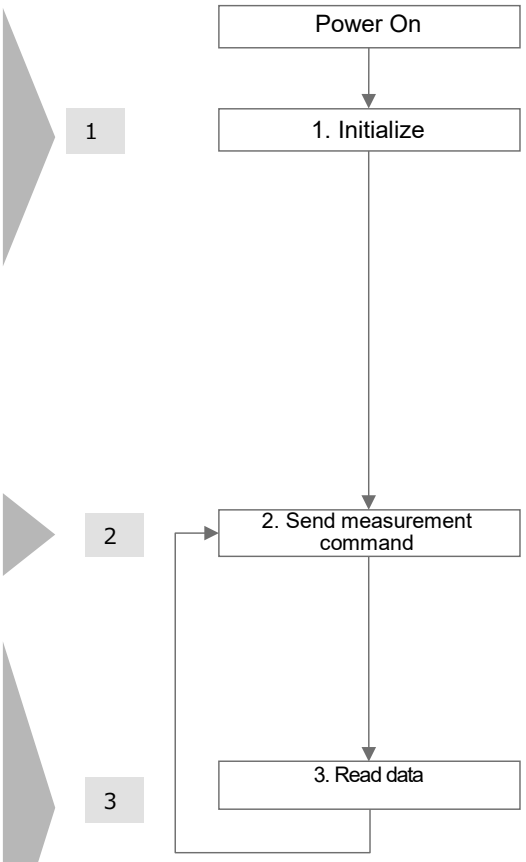


Fig.24 Sample code structure for Arduino

The operation procedure of the sample code for Arduino is as follows.

(1) Connect D6F-PH, harness and OMRON evaluation board (2JCIE-EV01-AR1) to Arduino

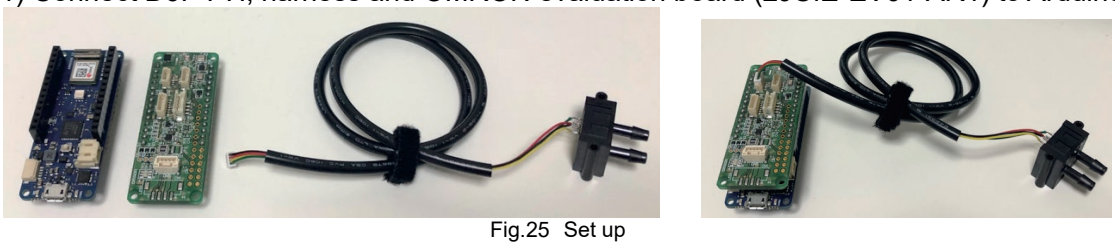


Fig.25 Set up

(2) Download Arduino IDE

Download Arduino IDE from the URL below.

https://www.arduino.cc/en/Main/Software

(3) Recognize Arduino on Arduino IDE

Open Arduino IDE and connect Arduino to the PC via USB.

If the following message appears, install the package for Arduino MKR.

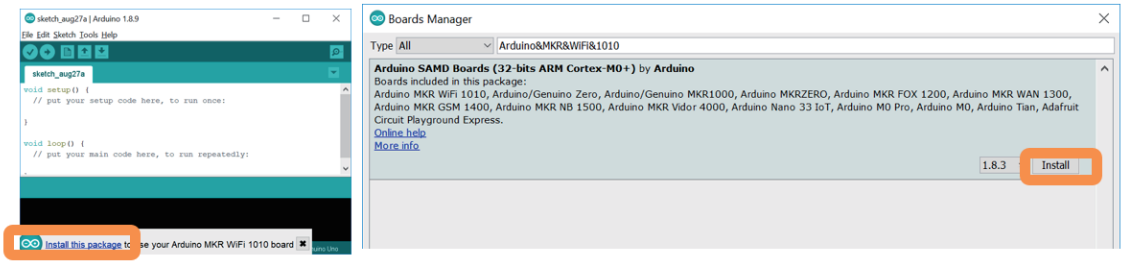


Fig.26 Arduino IDE

Install the Driver.

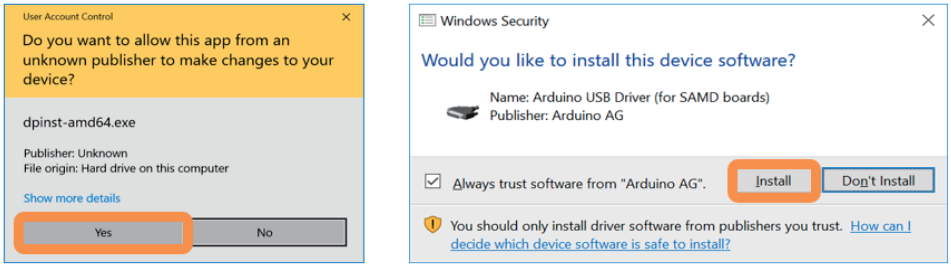


Fig.27 Driver

Search “Device Manager” in the Windows Start menu.

Check Arduino's COM port number recognized by the PC.

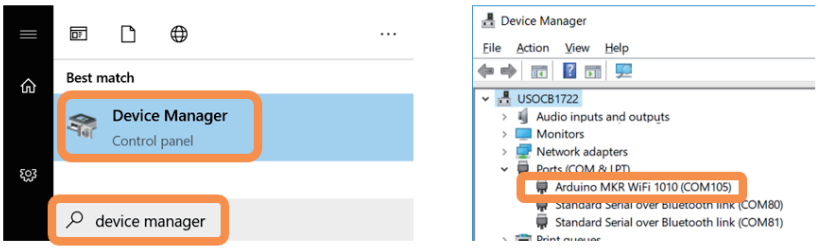


Fig.28 Device manager

“Tools” -> “Board Arduino” -> “Arduino MKR WiFi1010”.

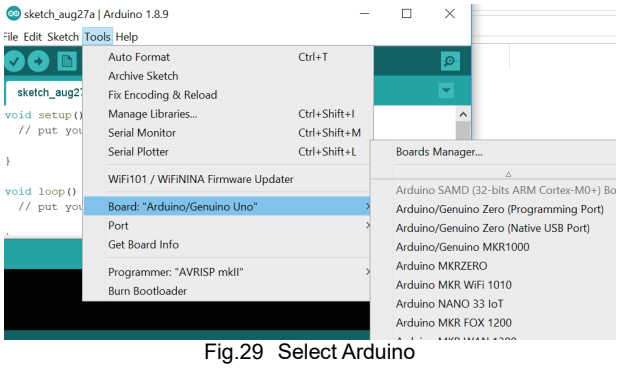


Fig.29 Select Arduino

“Tools” -> “PORT” -> Select COM port of Arduino.

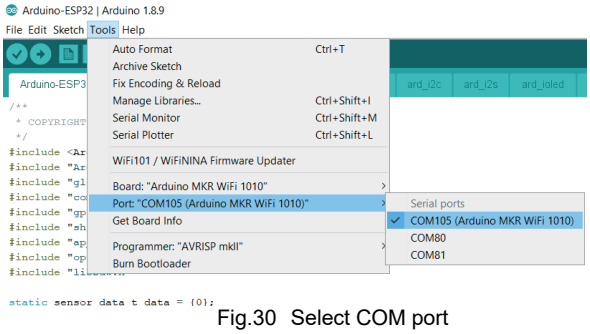


Fig.30 Select COM port

(4) Download sample code

Download the zip file by connecting to the GitHub's URL below.

GitHub URL: https://github.com/omron-devhub/d6f-2jcieev01-arduino

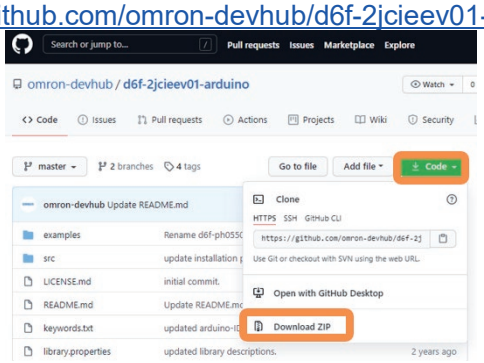


Fig.31 Download sample code

(5) Upload the sample code to Arduino.

“Sketch” -> “Include Library” -> “Add .ZIP Library”

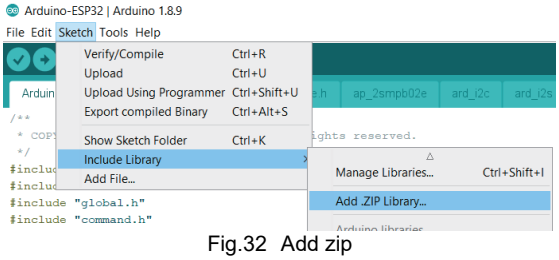


Fig.32 Add zip

Select the “Downloads” folder. Select "d6f-2jcieev01-arduino-master.zip".

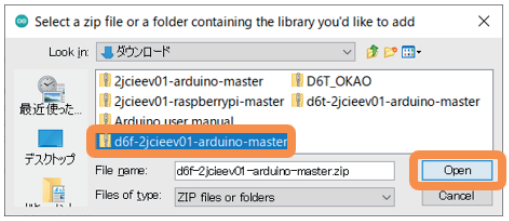


Fig.33 Select zip file

Select the file according to the differential pressure measurement range of the sensor.
“File” -> “Examples” -> “D6F-2JCIE-EV01”
-> “d6f-ph0025” or “d6f-ph0505” or “d6f-ph5050”

Fig.34 Select file

Click “Verify”. Check for any errors.

Fig.35 Verify

Click “Upload”. Check to see if "CPU reset" is shown.

Fig.36 Upload

(6) Obtaining data

“Tools” -> “Serial Monitor”

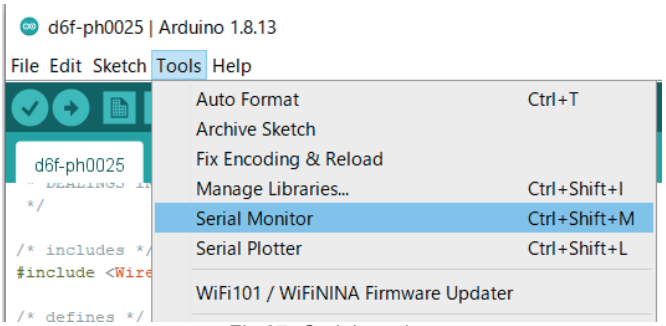


Fig.37 Serial monitor

The data is shown.

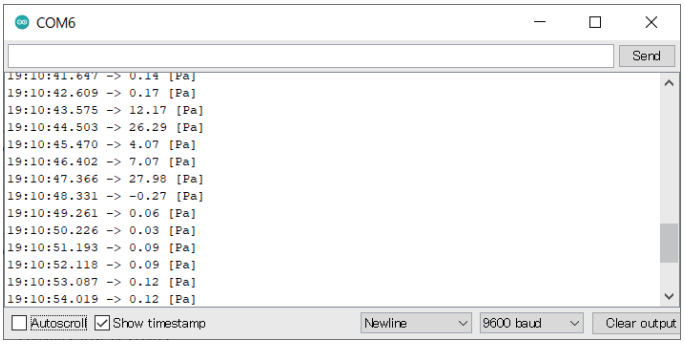


Fig.38 data

Please check each region's Terms & Conditions by region website.

OMRON Corporation

Electronic and Mechanical Components Company

Regional Contact

Americas
https://www.components.omron.com/

Asia-Pacific
https://ecb.omron.com.sg/

Korea
https://www.omron-ecb.co.kr/

Europe
http://components.omron.eu/

China
https://www.ecb.omron.com.cn/

Japan
https://www.omron.co.jp/ecb/

Cat. No. A288-E1-02
0421(0918)