TB(P) Series Specification

Non-contact Infrared Temperature Sensor

Ordering Information

TB□ - I2C - ◇◇◇

package code	
P	pcb type
(blank)	sensor only

lens code	FOV
H04	3.814°
H08	7.16°
H70	70°
S70	70°

* TB-I2C-H04: Sensor only, FOV=3.814°

* TBP-I2C-H08: pcb type, FOV=7.16°

* TBP-I2C-S70: pcb type, FOV=70°, Heatsink(x)

* H : Heatsink(O)

* S : Heatsink(X)

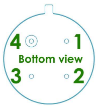
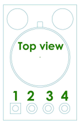
Product Specifications

laboratory temperature conditions: 25℃.

Parameter		min	Typ	Max	Unit
Supply voltage	TB Series	2.4	3.3	3.6	V
	TBP Series	3	3.3	5	V
Supply current			1.2		mA
Filter type		F5.5 (LWP cut on 5.5㎛)			
Object temperature range		-40		380	℃
Operating temperature		-40		85	℃
IR refresh rate			10		Hz
Accuracy(*)			±2		%
Resolution digital			0.01		℃
Emission coefficient		0.1	0.97	1.0	ε
Standard start-up time			200		ms
Stabilization time		1			min
SCL frequency		10		100	KHz
Weight	TB	**H04: 1.83g , H08: 1.60g, H70: 1.34g, S70: 1g**			
	TBP (without pin header)	**H04: 2.25g , H08: 2.12g, H70: 1.85g, S70: 1.5g**			
Communication interface		I2C			
Relative humidity		95% Max. non-condensing			

***: ±2% of reading or ±2℃ whichever is greater.**
Accuracy is only effective if the object is fully covered by the sensor's FOV and applicable to stable temperature conditions.

Pin Configuration

No.	TB series	TBP series
		
1	SCL	VIN(3~5V)
2	SDA	GND
3	3.3V	SCL
4	GND	SDA

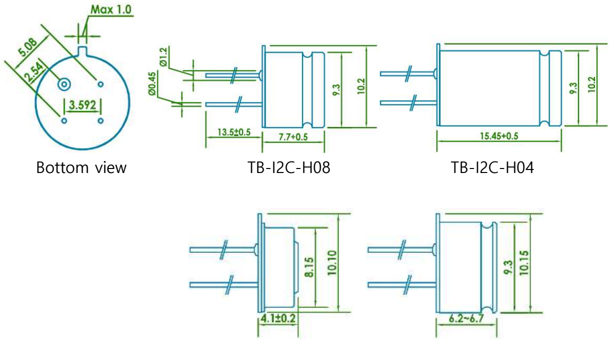
Accessories

	2.54mm pitch pin header (TBP series only)	1 pc

Dimensions

unit: mm

- TB series



Bottom view

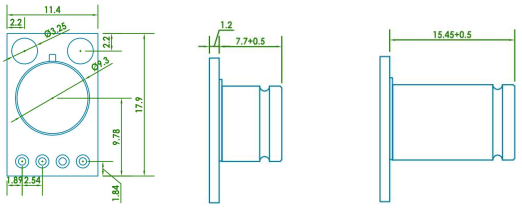
TB-I2C-H08

TB-I2C-H04

TB-I2C-S70

TB-I2C-H70

- TBP series



Top view

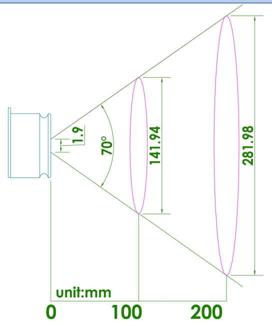
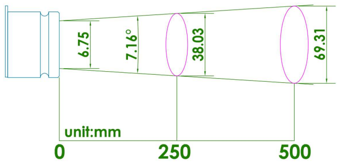
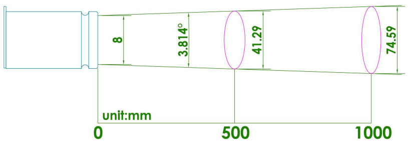
TBP-I2C-H08

TBP-I2C-H04

Calculate Field of View

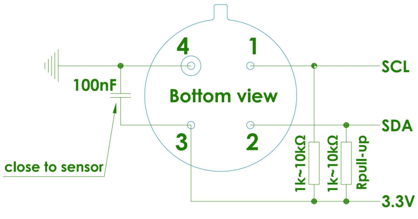
The FOV determines the size of the infrared measurement area according to the distance.

unit: mm

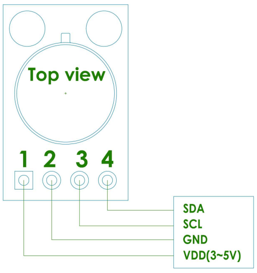
Angle	FOV
70°	 1.9 70° 141.94 281.98 unit:mm 0 100 200 diameter = 1.400415 x distance(mm) + 6.75 (mm)
7.16°	 6.75 7.16° 38.03 69.31 unit:mm 0 250 500 diameter = 0.125128 x distance(mm) + 6.75 (mm)
3.814°	 8 3.814° 41.29 74.59 unit:mm 0 500 1000 diameter = 0.066591 x distance(mm) + 8 (mm)

Typical application schematics

- TB series (External pull-up resistor required)



- TBP series(No external resistor required. 1.5kΩ pull-up resistor applied)



SMBus compatible 2-wire protocol

- The TB(P) series supports a 2 wire serial protocol. And the TB(P) only supports two protocols on standard SMBus interface.
  Supported commands are: Read Word, Write Word

Functional description

The TB(P) sensor module can only be used as a slave device. Maximum number of devices on one bus is 127, higher pull-up currents are recommended for higher number of devices, faster bus data transfer rates, and increased reactive loading of the bus. If the access to the device is a read operation it will respond with 16 data bits and 8 bit PEC only if its own slave address.

Bus Protocol

The TB(P) sensors are preset to address 0x3A by default.

1	7	1	1	8	1	1
S	Slave Address	Wr	A	Data Byte	A	P

S	Start Condition
Sr	Repeated Start Condition
Rd	Read (bit value of 1)
Wr	Write (bit value of 0)
A	Acknowledge(this bit can be 0 for ACK and 1 for Nack)
S	Stop condition
PEC	Packet Error Code(*)
□	Master-to-Slave
▨	Slave-to-Master

* The PEC calculation includes all bits except the START, REPEATED START, STOP, ACK, and NACK bits. The PEC is a CRC-8 with polynomial X8+X2+X1+1. The MSB(Most Significant Bit) of every byte is transferred first.

Packet Error Code(PEC) Calculation

A packet error checking mechanism according to SMBus specification version 1.1 is implemented. The packet error checking of SMBus devices uses an 8-bit cyclic redundancy check (CRC-8) to calculate the PEC.

The polynomial used is C(x)= 100000111 (= x8 + x2 + x + 1) .

The PEC calculation includes all bytes in the transmission, including address, command and data. It does not include ACK, NACK, START, STOP bits. Refer to the Arduino example code for detailed PEC calculation.

Read Word

1	7	1	1	8	1	1	7	1	1
S	Slave Address	Wr	A	Command Address(*)	A	Sr	Slave Address	Rd	A

	8	1	8	1	8	1	1
...	Data Byte Low	A	Data Byte High	A	PEC	A	P

*** Command Address configures**

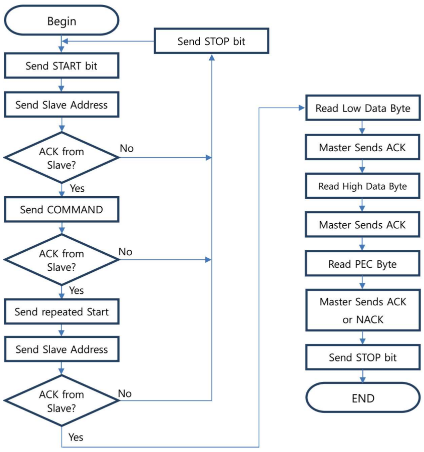
Register Name	Command Address	R/W
Ambient	0x06	R
Object	0x07	R
Emissivity correction coefficient	0x24	R/W
Slave Address	0x2E	R/W

Any device will respond to address zero

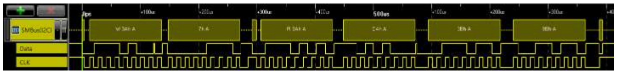
The MSB of the temperature data is an error flag (active high).

Please note after POR additional delay is needed before first reading can be started.

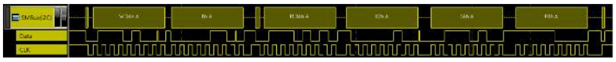
- Read operation flow diagram



Oscilloscope waveform image 1. (Read Object)



Oscilloscope waveform image 2. (Read Ambient)



- code example read Object temperature 0x07

0. power on & wait for sensor initialization(≥200ms)
1. Send START bit
2. Send 0x74 : [Slave Address(0x3A by default) + Wr(0)]
3. ACK from Slave
4. Send Command : 0x07 (Object) [Ambient: 0x06]
5. ACK from Slave
6. Send repeated Start
7. Send 0x75 : [Slave Address(0x3A) + Rd(1)]
8. ACK from slave
9. Read Data Byte Low (Send ACK)
10. Read Data Byte High (Send ACK)
11. Read PEC (Send ACK or NACK)
12. Send Stop bit
13. Wait >100ms & repeat 1...13

Object Temperature To

To is the object temperature derived from thermopile and ambient sensor outputs as absolute (Kelvin) temperature

$$T_o\left[°\mathrm{K}\right] = T_o reg \times 0.02$$

Please note that 1LSB corresponds to 0.02℃ and the MSB bit is error flag (if "1" then error). Maximum register value is 0x7FFF - maximum possible value returned by TB(P) The result is calculated by following expressions (valid for both To and Ta):

1. Convert it to decimal value i.e. 0x3B99 = 15,257d
2. Multiply by 0.02(or divide by 50) i.e. 15,257 x 0.02 = 305.14°K (result in Kelvin)
3. Convert K → ℃ i.e. 305.14 - 273.15 = 31.99℃

Ambient Temperature Ta

The Sensor die temperature is measured with a PTAT element. All the sensors conditioning and data processing is handled on-chip.

$$T_a\left[°\mathrm{K}\right] = T_a reg \times 0.02$$

Maximum register value is 0x7FFF.
Output Limit: 0x2DE4(-38.19℃) ... 0x4DC4(125.01℃)

Emissivity Correction Coefficient

The TB(P) is calibrated for an object emissivity of 0.97. It can be easily customized by the customer for any other emissivity in the range 0.1...1.0.

Factory default 0.97 = 63,568(0xF850)
(0.97 *65536) - 1 = 63568.92 round down → 63568
Max 1.00 = 65535(0xFFFF)

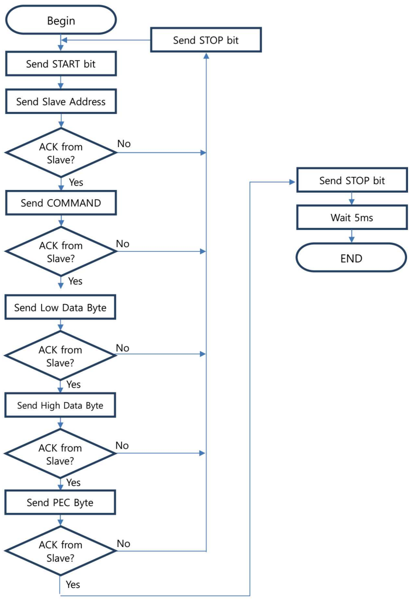
Write Word

1	7	1	1	8	1	
S	Slave Address	Wr	A	Command Address	A	...

	8	1	8	1	8	1	1
...	Data Byte Low	A	Data Byte High	A	PEC	A	P

Before changing of cell contents the respective cells need to be erased. Erasing is a writing of zeros to the respective Command Address. After writing operation the device needs 5ms to process the new values. After changing the "Command Address" cell contents, it is strongly recommended to restart the sensor by power off/on.

- Write operation flow diagram



- code example write Slave Address 0x2E

Erasing:

1. Send START bit
2. Send 0x74 : [Slave Address(0x3A by default) + Wr(0)]
3. Send Command : 0x2E
4. Send Data Byte Low : 0x00 (for erasing)
5. Send Data Byte High : 0x00 (for erasing)
6. Send PEC : 0x05 (calculated CRC value)
7. Send STOP bit
8. Wait 5ms

Writing : (Slave Address 0x3A → 0x4C)

9. Send START bit
10. Send 0x74 : [Slave Address(0x3A by default) + Wr(0)]
11. Send Command : 0x2E
12. Send Data Byte Low : 0x4C
13. Send Data Byte High : 0x00
14. Send PEC : 0xA2 (calculated CRC value)
15. Send STOP bit
16. Wait 5ms
17. Reset of the sensor (power off/ on)

- code example write Emissivity correction coefficient Address 0x24

Erasing:

1. Send START bit
2. Send 0x74 : [Slave Address(0x3A by default) + Wr(0)]
3. Send Command : 0x24
4. Send Data Byte Low : 0x00 (for erasing)
5. Send Data Byte High : 0x00 (for erasing)
6. Send PEC : 0x05 (calculated CRC value)
7. Send STOP bit
8. Wait 5ms

Writing : (coefficient → 0xF850)

9. Send START bit
10. Send 0x74 : [Slave Address(0x3A by default) + Wr(0)]
11. Send Command : 0x24
12. Send Data Byte Low : 0x50
13. Send Data Byte High : 0xF8
14. Send PEC : 0x68 (calculated CRC value)
15. Send STOP bit
16. Wait 5ms
17. Reset of the sensor (power off/ on)

Timing

The TB(P) meets all the timing specifications of the SMBus. The maximum frequency of the TB(P) SMBus is 100KHz and the minimum is 10KHz.

The specific timings in TB(P)'s SMBus are:

Timeout L is the maximum allowed time for SCL to be low during communication. After this time the TB(P) will reset its communication block and will be ready for new communication – not more than 27ms;

Timeout H is the maximum allowed time for SCL to be high during communication. After this time TB(P) will reset its communication block assuming that the bus is idle (according to the SMBus specification) – not more than 45μs.

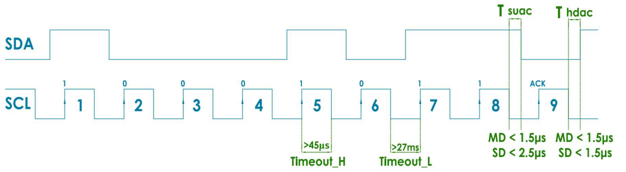
Tsuac(SD) is the time after the eighth falling edge of SCL that TB(P) will force SDA low to acknowledge the last received byte – not more than 2.5μs.

Thdac(SD) is the time after the ninth falling edge of SCL that TB(P) will release the SDA (so the MD can continue with the communication) – not more than 1.5μs.

Tsuac(MD) is the time after the eighth falling edge of SCL that TB(P) will release SDA (so that the MD can acknowledge the last received byte) – not more than 1.5μs.

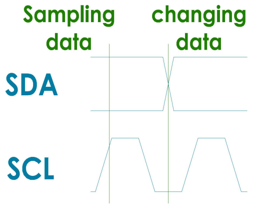
Thdac(MD) is the time after the ninth falling edge of SCL that TB(P) will take control of the SDA (so it can continue with the next byte to transmit) – not more than 1.5μs.

The indexes MD and SD for the latest timings are used – MD when the master device is making acknowledge; SD when the slave device is making acknowledge.



SMBus timing

Bit transfer



Bit transfer on SMBus

The data on SDA must be changed when SCL is low (min 300ns after the falling edge of SCL). The data is fetched by both MD and SDs on the rising edge of the SCL.

Products handling precaution

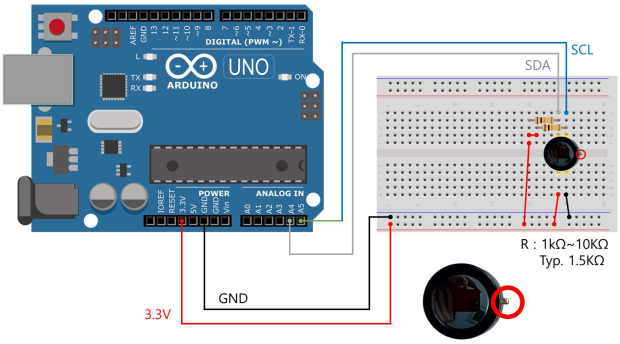
※ When it comes to dust removal by air, the best method is to use a blower, and to avoid using compressed air.
※ Do not press the lens with your hands or any other object.
※ Do not scratch the lens surface with sharp objects.
※ Voluntary disassembly and modification of the product is prohibited.
※ Avoid direct sunlight, chemical substance, heat or fire.
※ Water resistance is not guaranteed.

Tutorial 1(Arduino UNO - TB(P)-I2C-H08)

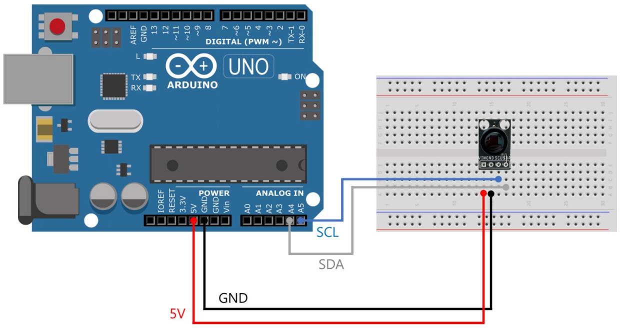
- **Requirements**

  Hardware: Arduino Uno, TB(P) series Sensor
  Software: Arduino IDE.

- **Connection Diagram**
  - ***TB-I2C-H08***



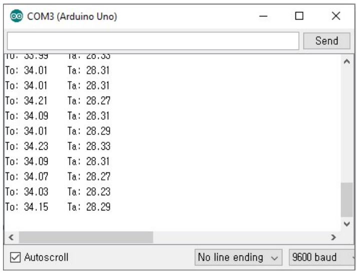
  - ***TBP-I2C-H08***



- **Sample code**

  https://www.diwellshop.com/web/en/TB/TB_example.zip

- **Expected Results.**



```
To: 33.99    Ta: 28.33
To: 34.01    Ta: 28.31
To: 34.01    Ta: 28.31
To: 34.21    Ta: 28.27
To: 34.09    Ta: 28.31
To: 34.01    Ta: 28.29
To: 34.23    Ta: 28.33
To: 34.09    Ta: 28.31
To: 34.07    Ta: 28.27
To: 34.03    Ta: 28.23
To: 34.15    Ta: 28.29
```

☑ Autoscroll

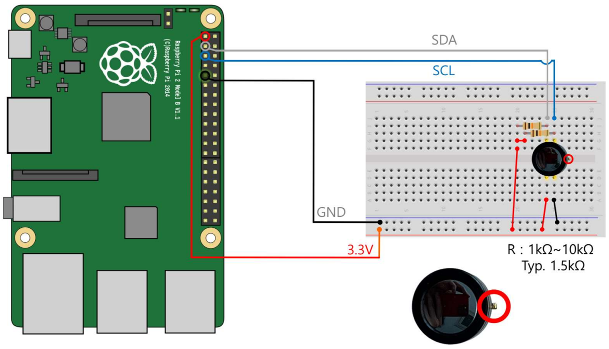
Tutorial 2(Raspberry Pi 2 - TB(P)-I2C-H08)

- **Requirements**

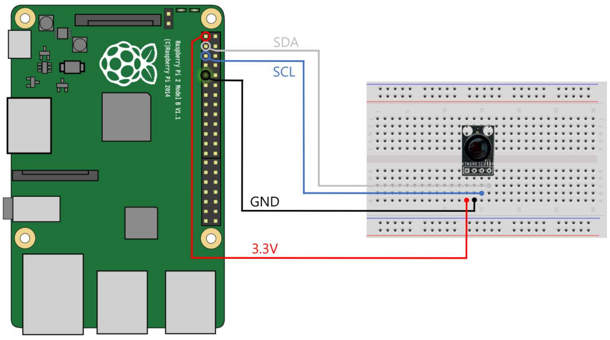
  Hardware: Raspberry Pi 2 , TB(P) series Sensor

  Software: wiringPi library

- **Connection Diagram**
  - *TB-I2C-H08*



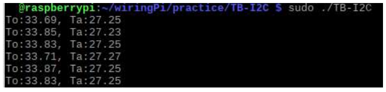
  - *TBP-I2C-H08*



- **Sample code**

  https://www.diwellshop.com/web/en/TB/TB_example.zip

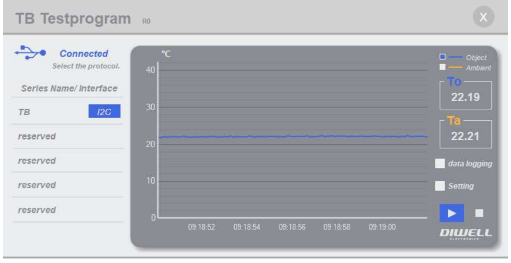
- **Expected Results.**



```
  @raspberrypi:~/wiringPi/practice/TB-I2C $ sudo ./TB-I2C
To:33.69, Ta:27.25
To:33.85, Ta:27.23
To:33.83, Ta:27.25
To:33.71, Ta:27.27
To:33.87, Ta:27.25
To:33.83, Ta:27.25
```

PC Software

The program requires Windows 10 to run.
It is not guaranteed to be used on other OS.
For more information, refer to the Test Board manual.
https://www.diwellshop.com/web/en/TB/TB-I2C_Testboard_en.pdf



Additional information

Manufacturer: DIWELL Electronics Co., Ltd. (South Korea)
Technical support: mailto:expoeb2@diwell.com, mailto:dsjeong@diwell.com

Revision history

Version	Date(Y,M,D)	Description
1.0.0	2022. 5. 9.	First version is released
1.0.1	2022. 5. 10.	Additional sensor types added (S70, H70)