



- 비접촉 온도 측정
- 소형 모듈
- 30°C ~ 41°C 구간 정밀 측정(typ  $\leq \pm 0.3^\circ\text{C}$ )
- Digital Resolution : 0.01°C
- 데이터 통신 : UART 3.3V TTL
- 라즈베리파이 코드 및 회로도 제공

## ▶ 제품 설명

- 본 제품은 RAW Data 출력 모드, 보상 모드 2가지 측정모드를 지원합니다.
- RAW Data 출력 모드는 대상 표면의 온도 20~45°C 구간의 실제 온도(Raw Data)를 출력합니다.
- 보상 모드는 대상의 온도에 따라 적절한 온도 보상 수치를 적용합니다.  
자세한 사항은 당사로 문의 바랍니다(070-8235-0820).
- 보상 모드 사용시 반드시 적정 거리(12cm)를 지켜야 합니다.
- 정확도 및 측정 구간 등의 정보는 본 문서 2page의 특성표를 참고하십시오.
- 출입관리를 요하는 응용분야에 적용이 가능합니다.
- 비접촉 측정방식으로 각종 바이러스 감염 위험을 줄일 수 있습니다.

### ※ 주의 사항 및 요구 사항(필독) :

- 본 제품은 산업용으로 의료기기인 체온계가 아닙니다.
- 반드시 상온이 유지되는 실내(직사광선이 없는)에서 사용해야 합니다.
- 제품(pcb 포함)을 손으로 잡고 측정하면 안됩니다(중요) PCB를 고정하여 사용하십시오.
- 측정 방향에 열원(에어컨, 히터, 밥솥, 전자레인지 등)이 있으면 안됩니다.

## ▶ Ordering Guide

CTMS -△

측정 각도		구매 링크
3	2.886°	<a href="http://www.diwellshop.com/product/detail.html?product_no=944">http://www.diwellshop.com/product/detail.html?product_no=944</a>

## ▶ Electrical Requirements

Parameter	Symbol	Conditions	min	Typ	Max	Unit
공급전압	Vcc	Measured versus GND	4.75	5	5.5	V
소비 전류(5V)		no output load		18		mA
<b>UART TX High Voltage</b>				3.3	3.5	V
<b>UART RX High Voltage</b>			3	3.3	3.5	V

## ▶ General Specifications

- if not otherwise noted, 25°C ambient temperature, 5V supply voltage were applied

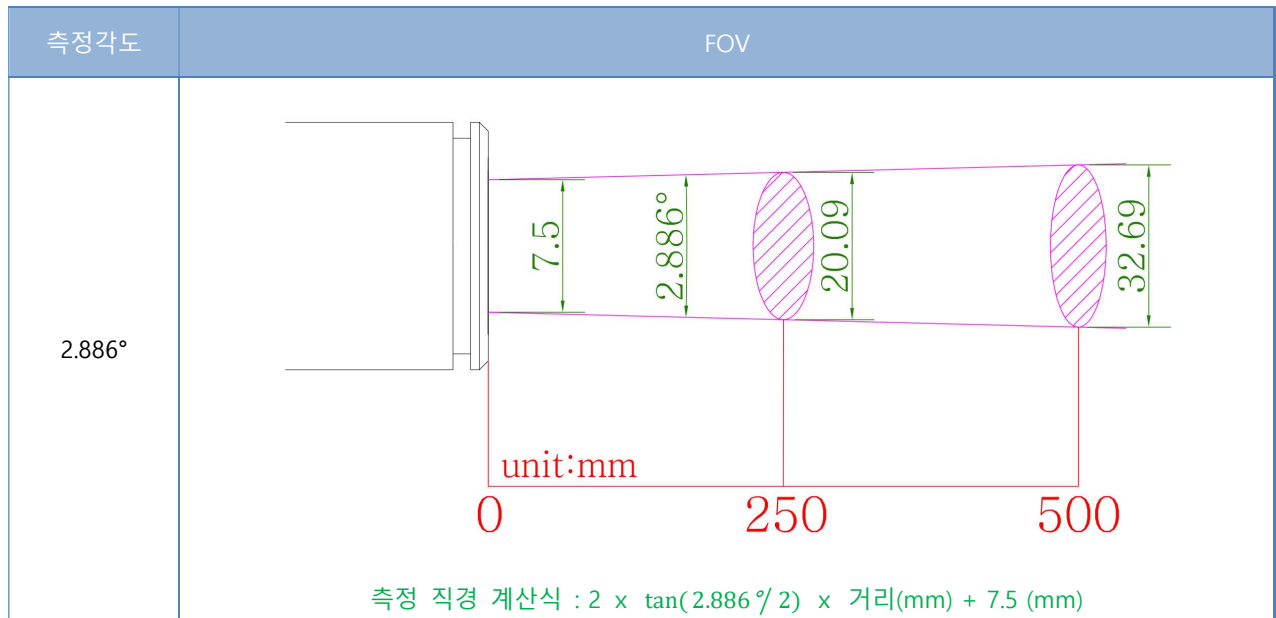
Parameter		min	Typ	Max	Unit
측정각도(FOV)			2.886		°
측정 거리 (정확도 보장 거리)		11	12	13	cm
온도 측정 범위 (Measurement Range)	RAW Data 출력 모드 (정확도 보장 온도)	20 (30)	-	45 (41)	°C
	보상 모드	20	-	45	°C
정확도(Accuracy)	RAW Data 출력 모드		±0.3	±0.5	°C
동작 온도 (Operating temp)		16		35	°C
측정 파장 대역 (Spectral Range)		5.5	-	14	μm
IR refresh rate			10		Hz
Resolution Digital			0.01		°C
Standard Start-UP Time			1	2	sec
Stabilization Time		1			min

## ▶ Absolute Maximum Ratings

- Supply voltage : 5.5V
- Operating Temperature Range : 10°C ~ 70°C
- Storage Temperature Range : -20°C ~ 85°C

위 조건을 넘어서게 되면 제품의 수명을 보장할 수 없습니다. 반드시 Electrical Requirements 를 지켜주세요.

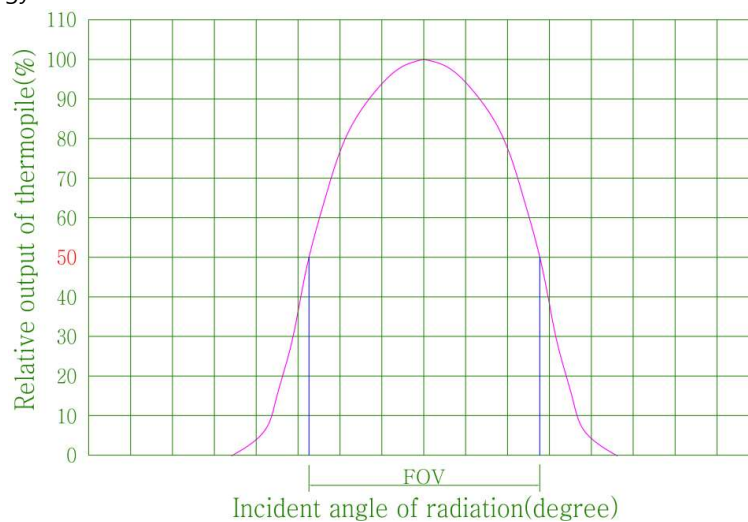
## ▶ Optical field of view (FOV)



※ 측정 거리 표시는 거리에 따른 측정 면적을 표시하는 것으로 정확도가 보장되는 거리를 의미하는 것이 아닙니다. 거리가 멀어질수록 측정값은 실제 온도와 차이가 발생합니다.

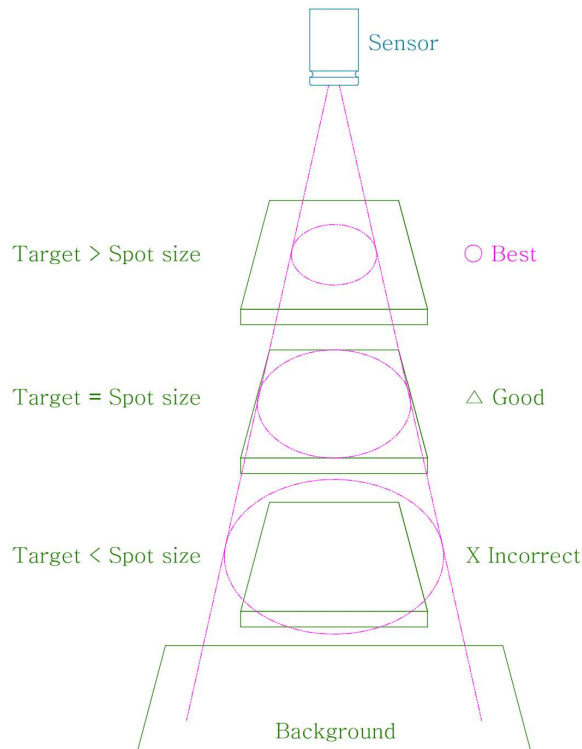
※ 측정하고자 하는 물체의 크기는 위 계산식의 spot size보다 충분히 더 커야 측정이 용이합니다.  
다음 페이지 Distance and spot size 그림을 참고하십시오.

The optical chart below indicates the nominal target spot diameter at any given distance from the sensing head and assumes 50% energy.



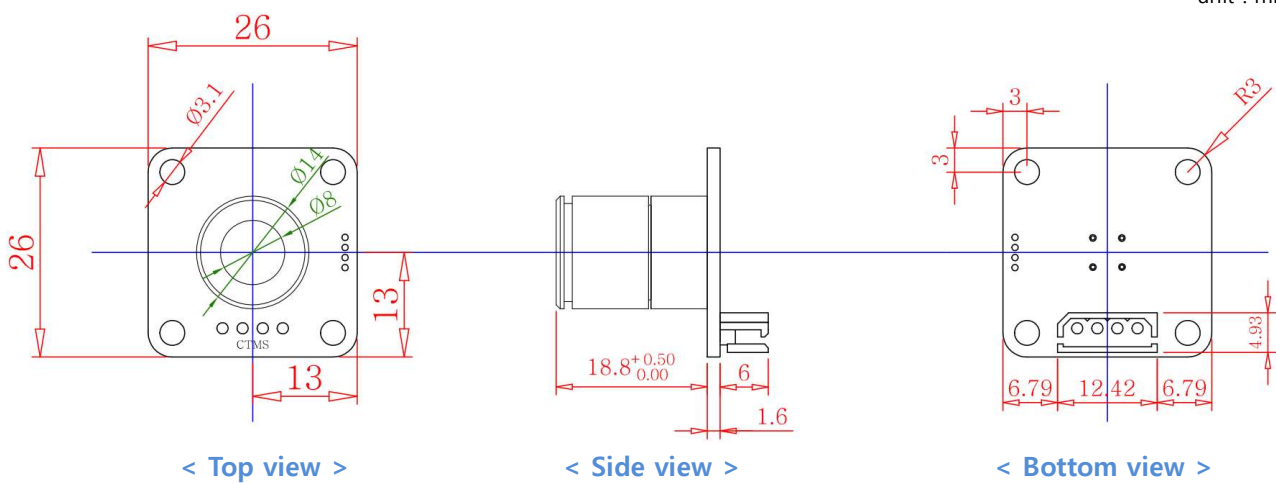
## ▶ DISTANCE AND SPOT SIZE

Spot Size는 아래 그림에서와 같이 측정하고자 하는 대상의 면적보다 반드시 작아야 합니다.

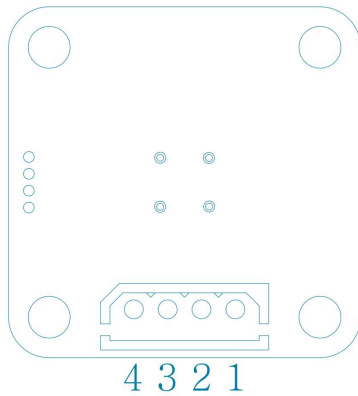


## ▶ Mechanical Dimensions

unit : mm



## ▶ Pin Assignment



Pin Number	통신 방식
	UART
1	5V
2	TX (3.3V TTL)
3	RX (3.3V TTL)
4	GND

- ※ 제품 헤더 정보 : molex 5267-04
- ※ 결합용 하우징 정보 : molex 5264-04
- (주의) 제품 본체만 구매시 케이블은 미포함입니다.  
TestKit 구매시에만 케이블이 같이 출고됩니다.

## ▶ 기본 구성품



본체 - 1ea

## ▶ Accessories (별도 구매)



Data cable

## ▶ 온도 측정 결과표

※ 주의 : 모든 제품이 동일한 오차를 갖진 않습니다

### ■ CTMS-3

챔버온도 (°C)	Mode 온도 기준(°C)	RAW Data 출력 모드	
		측정 결과	오 차
16	30	29.79	-0.21
	38	37.75	-0.25
	41	40.71	-0.29
25	30	29.55	-0.45
	38	37.89	-0.11
	41	41.13	+0.13
35	30	30.39	+0.39
	38	38.37	+0.37
	41	41.45	+0.45

측정거리 : 12cm, 정확도 기준 : Max ±0.5°C 이내

센서 흑체 접근 후 1초후에 측정 1회

각 온도 구간 안정화 시간 30분 후 측정

※ 온도 정확도는 아래의 조건 등에 영향을 받아 변화될 수 있습니다.

- ▷ 센서의 열평형이 이루어져 있을 것.
- ▷ 센서 패키지에 열이 가해지거나, 온도 차이가 없을 것.
- ▷ 측정 물체가 센서의 측정 영역에 꼭 채워질 것.
- ▷ 측정 영역에서 대상 표면의 온도가 균일하게 분포할 것.
- ▷ 측정 영역에서 대상 표면 재질의 방사율이 일정할 것.
- ▷ 측정할 때마다 거리가 변화되지 않을 것.

※ 정확도에 관한 이해(필독)

온도 정확도는 모든 환경에서 항상 만족하는 것은 아닙니다. 센서의 열평형 상태가 유지되지 않는 환경에서는 정확도 유지가 어렵다는 것을 가장 중요하게 이해하고 있어야 합니다. 예를 들어, 센서 패키지에 순간적으로 온도 변화를 일으킬 수 있는 외부 요소(히터, 에어컨, 센서 근처의 발열 IC, 손으로 잡는 행위 등)가 미친다면, 센서는 열충격을 받기 때문에 측정 오차가 발생할 수 있습니다.

이러한 온도 차이를 temperature gradient 라 부르며, **CTMS 제품은 내부적으로 열충격을 active하게 보정하는 기능이 추가되었습니다.** 따라서 CTMS는 이러한 보정 기능이 없는 센서들과 비교해 좀 더 안정적인 측정이 가능합니다.

하지만 완벽하게 제거되는 것은 아니기 때문에, 애초에 이런 온도차이가 발생되지 않도록 외부 요소를 피하거나 센서를 보호해야 합니다.

## ▶ 거리 변화에 따른 온도 측정

※ 주 의 : 모든 제품이 동일한 오차를 갖진 않습니다

※ CTMS 측정을 위한 적정 거리는 12cm 입니다. 따라서 적정 거리를 벗어날 경우 오차가 발생할 수 있습니다. 다만, 오차가 발생해도 이를 감안해서 사용하시려는 업체가 있을 수 있으니, 참고용으로 대략 50~55cm 거리에서 측정한 결과를 첨부합니다. 데이터는 RAW Data 출력 모드 수치 입니다. 측정 범위에 장애물(천, 유리, 투명아크릴, 렌즈 등)이 있다면 해당 영역에 따라 오차가 더 발생할 수 있습니다. (주의) 측정시 적정 거리가 아닐 경우 보상 모드 사용을 하지 마십시오.

실험 조건 < Room temp : 25°C, 흑체 size : 직경 12cm >

모델명	흑체 온도	측정 거리 : 12cm (적정거리)	측정 거리 : 50~55cm	
		측정 결과	측정 결과	오차 (흑체온도 - 측정결과)
CTMS-3	38°C	38.0°C	37.4°C	-0.6°C

## ▶ 통신 프로토콜

### 1. UART 통신

※ UART 3.3V TTL (5V로 동작하는 MCU와는 통신포트를 직접 연결하면 안됩니다.)

※ 통신 규격

통신속도(bps)	DATA bit	Stop bit	Parity
19,200	8	1	none

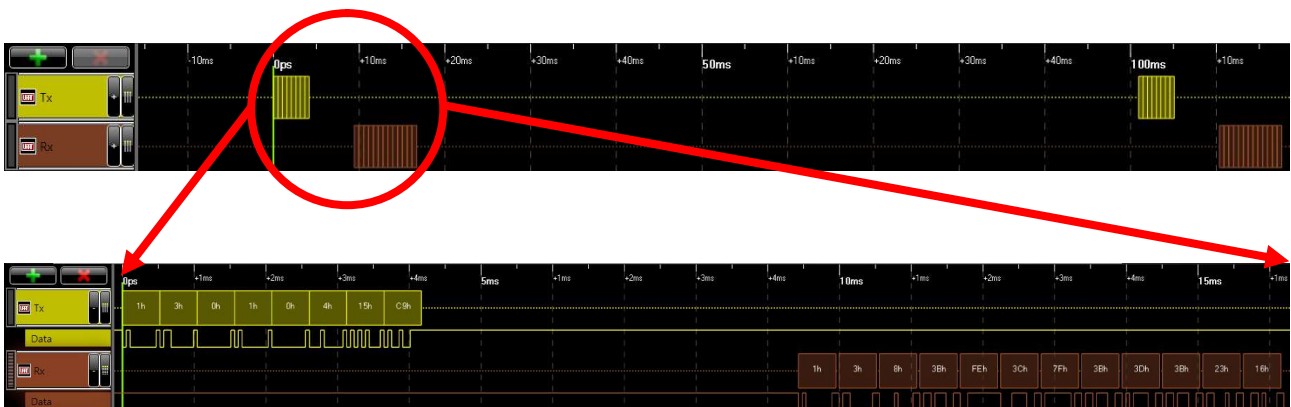
### 2. 온도 데이터 읽기

※ Request 주기 : 반드시 최소 100ms 이상

※ 전원 공급 후 첫 Request Timing : 최소 1초 이후

※ Request 후 첫 Byte 응답 시간 : 3~6ms 이후

(센서의 Start-up 시간보다 빠른 Request 는 응답이 없거나 데이터 수치 0 이 전송됩니다.)



## 2-1. Protocol table

Request ( Master → Sensor )			Response ( Sensor → Master )		
Field Name	Hex	DEC	Field Name	Hex	DEC
Request 1	0x01	1	Start	0x01	1
Request 2	0x03	3	Function	0x03	3
Request 3	0x00	0	Byte Count	0x08	8
Request 4	0x01	1	Data Value 1 Hi	RAW Data	
Request 5	0x00	0	Data Value 1 Lo		
Request 6	0x04	4	Data Value 2 Hi	보상 Data	
Request 7 CRC	page 10~11 참고 (※ <b>Note1</b> )		Data Value 2 Lo		
Request 8 CRC			Data Value 3 Hi	reserved	※ <b>Note3</b>
			Data Value 3 Lo		
			Data Value 4 Hi	Sensor 온도	
			Data Value 4 Lo		
			CRC	page 10~11 참고 (※ <b>Note2</b> )	
			CRC		
8 Byte Request			13 Byte Response		

※ **Note 1, 2** : CRC 는 통신과정에서 오류가 없는지 검증하는 데이터 입니다.

※ **Note 3** : 데이터에 의미는 없습니다. dummy 처리하십시오.

※ 모든 Request 및 Response Data 는 CRC 데이터가 포함됩니다. 임의 데이터를 넣으면 응답을 하지 않습니다. 시리얼 통신은 외부 노이즈 또는 다양한 원인에 의해 데이터 수신에 원활하지 않을 경우가 발생할 수 있습니다. 이럴 경우 비정상적인 온도로 표시될 수 있기 때문에 수신된 데이터에 오염이 없는지 반드시 CRC 체크를 통해 검증을 한 후, 데이터를 활용해야 합니다.

※ CTMS 제품은 UART 통신이며, 센서와 마스터는 반드시 1:1로 연결해야 합니다.

※ 동시에 여러 제품을 연결해야 하는 환경은 RS-485 통신을 지원하는 CTM series 제품을 선택하십시오.



## 2-2. 온도 계산 법

모든 온도 데이터는 각각 아래와 같은 계산 과정을 거치면 됩니다.

uTemp : unsigned 2 바이트 크기 변수 선언

dTemp : float 또는 double 형 선언

$$\text{uTemp} = (\text{Data Hi byte}) * 256 + (\text{Data Lo byte})$$

$$\text{dTemp} = (\text{double})\text{uTemp} * 0.02 - 273.15$$

### ■ 계산 예제 1

Data Value Hi = 0x3B,

Data Value Lo = 0xFB

$$0x3B * 256 + 0xFB = 15,355$$

$$(15,355 * 0.02) - 273.15 = 33.95$$

즉, 33.95°C 가 됩니다.

### ■ 계산 예제 2

Data Value Hi = 0x33

Data Value Lo = 0x65

$$0x33 * 256 + 0x65 = 13,157$$

$$(13,157 * 0.02) - 273.15 = -10.01$$

즉, -10.01°C 가 됩니다.

## 2-3. output data limit.

데이터의 출력 범위는 19.99°C ~ 45.01°C 입니다. 해당 범위를 제외한 구간은 출력 수치에 제한이 걸립니다.

→ 센서에서 출력하는 최소 수치(min) : 14,657(dec) = 19.99°C

→ 센서에서 출력하는 최대 수치(Max) : 15,908(dec) = 45.01°C

즉, 5°C, 10°C 를 측정하여도 19.99°C 로 출력합니다.

50°C, 60°C 를 측정하여도 45.01°C 로 출력합니다.

**Sensor 온도**는 별도 출력 수치에 제한이 없습니다. Sensor 온도는 센서가 동작하기에 적합한 환경인지 체크하는 용도로 활용하십시오. Sensor 동작 온도 16 ~ 35°C 구간에서만 RAW Data의 정확도가 보장됩니다.

### 3. CRC란? (개발자 분은 반드시 아래 내용을 확인해서 CRC 값을 적용해야 합니다.)

CRC(Cyclic Redundancy Check) 는 데이터 통신시 전송 데이터에 오류가 있는지 확인 하기 위해 체크값을 결정하는 방식입니다. 모든 Request/ Response 메시지에 2byte 크기의 CRC 값이 붙어서 전송됩니다. 각 전송/ 수신 데이터에 따라 CRC값이 맞는지 반드시 검증 한 후 데이터를 활용해야 합니다.

Polynomial 값은 0xA001 이지만, 아래 예제 코드는 비트단위 연산 부하를 줄이기 위해 CRC 테이블을 활용하는 예제 입니다. 요즘 시스템들은 메모리가 작지 않기 때문에 테이블 적용을 추천합니다.

※ CRC 계산 예제 코드.

```
const uint16_t TableCRC16[256]={
    0x0000, 0xC0C1, 0xC181, 0x0140, 0xC301, 0x03C0, 0x0280, 0xC241, 0xC601, 0x06C0,
    0x0780, 0xC741, 0x0500, 0xC5C1, 0xC481, 0x0440, 0xCC01, 0x0CC0, 0x0D80, 0xCD41,
    0x0F00, 0xCF41, 0xCE81, 0x0E40, 0x0A00, 0xCAC1, 0xCB81, 0x0B40, 0xC901, 0x09C0,
    0x0880, 0xC841, 0xD801, 0x18C0, 0x1980, 0xD941, 0x1B00, 0xDB41, 0xDA81, 0x1A40,
    0x1E00, 0xDEC1, 0xDF81, 0x1F40, 0xDD01, 0x1DC0, 0x1C80, 0xDC41, 0x1400, 0xD4C1,
    0xD581, 0x1540, 0xD701, 0x17C0, 0x1680, 0xD641, 0xD201, 0x12C0, 0x1380, 0xD341,
    0x1100, 0xD1C1, 0xD081, 0x1040, 0xF001, 0x30C0, 0x3180, 0xF141, 0x3300, 0xF3C1,
    0xF281, 0x3240, 0x3600, 0xF6C1, 0xF781, 0x3740, 0xF501, 0x35C0, 0x3480, 0xF441,
    0x3C00, 0xFCC1, 0xFD81, 0x3D40, 0xFF01, 0x3FC0, 0x3E80, 0xFE41, 0xFA01, 0x3AC0,
    0x3B80, 0xFB41, 0x3900, 0xF9C1, 0xF881, 0x3840, 0x2800, 0xE8C1, 0xE981, 0x2940,
    0xEB01, 0x2BC0, 0x2A80, 0xEA41, 0xEE01, 0x2EC0, 0x2F80, 0xEF41, 0x2D00, 0xEDC1,
    0xEC81, 0x2C40, 0xE401, 0x24C0, 0x2580, 0xE541, 0x2700, 0xE7C1, 0xE681, 0x2640,
    0x2200, 0xE2C1, 0xE381, 0x2340, 0xE101, 0x21C0, 0x2080, 0xE041, 0xA001, 0x60C0,
    0x6180, 0xA141, 0x6300, 0xA3C1, 0xA281, 0x6240, 0x6600, 0xA6C1, 0xA781, 0x6740,
    0xA501, 0x65C0, 0x6480, 0xA441, 0x6C00, 0xACC1, 0xAD81, 0x6D40, 0xAF01, 0x6FC0,
    0x6E80, 0xAE41, 0xAA01, 0x6AC0, 0x6B80, 0xAB41, 0x6900, 0xA9C1, 0xA881, 0x6840,
    0x7800, 0xB8C1, 0xB981, 0x7940, 0xBB01, 0x7BC0, 0x7A80, 0xBA41, 0xBE01, 0x7EC0,
    0x7F80, 0xBF41, 0x7D00, 0xBDC1, 0xBC81, 0x7C40, 0xB401, 0x74C0, 0x7580, 0xB541,
    0x7700, 0xB7C1, 0xB681, 0x7640, 0x7200, 0xB2C1, 0xB381, 0x7340, 0xB101, 0x71C0,
    0x7080, 0xB041, 0x5000, 0x90C1, 0x9181, 0x5140, 0x9301, 0x53C0, 0x5280, 0x9241,
    0x9601, 0x56C0, 0x5780, 0x9741, 0x5500, 0x95C1, 0x9481, 0x5440, 0x9C01, 0x5CC0,
    0x5D80, 0x9D41, 0x5F00, 0x9FC1, 0x9E81, 0x5E40, 0x5A00, 0x9AC1, 0x9B81, 0x5B40,
    0x9901, 0x59C0, 0x5880, 0x9841, 0x8801, 0x48C0, 0x4980, 0x8941, 0x4B00, 0x8BC1,
    0x8A81, 0x4A40, 0x4E00, 0x8EC1, 0x8F81, 0x4F40, 0x8D01, 0x4DC0, 0x4C80, 0x8C41,
    0x4400, 0x84C1, 0x8581, 0x4540, 0x8701, 0x47C0, 0x4680, 0x8641, 0x8201, 0x42C0,
    0x4380, 0x8341, 0x4100, 0x81C1, 0x8081, 0x4040
};
```

```
uint16_t CalcCRC16(uint8_t *pdata, uint16_t DataLen)
{
    uint16_t AccumCRC16 = 0xFFFF;
    uint8_t i, j;
    for(j=0; j<DataLen; j++)
    {
        i = (AccumCRC16 ^ *(pdata++) ) & 0xFF;
        AccumCRC16 = ((AccumCRC16>>8) ^ TableCRC16[i]) & 0xFFFF;
    }
    return AccumCRC16;
}
```

※ 온도 읽는 명령 CRC 구하는 과정 예제.

온도 읽는 명령의 CRC 값을 구해보도록 하겠습니다.

먼저 데이터를 전송할 적당한 크기의 전역 변수를 선언합니다.

```
uint8_t RequestData[8];          // 전송할 배열
uint16_t crc;                    // 연산한 crc값 임시 저장 변수
```

본문 (2-1. Protocol table) 를 참고하여 메인 함수에서 배열 요소에 값을 할당합니다.

```
RequestData[0] = 0x01;
RequestData[1] = 0x03;
RequestData[2] = 0x00;
RequestData[3] = 0x01;
RequestData[4] = 0x00;
RequestData[5] = 0x04;
```

여기까지가 보낼 데이터이며, 여기에 CRC 2byte를 계산하여 추가해야 합니다.

이전 페이지의 CalcCRC16 함수를 사용합니다.

```
crc = CalcCRC16(RequestData, 6); // RequestData 배열 0~5 까지, 6 바이트 데이터를 연산합니다.
```

```
RequestData[6] = (unsigned char)((crc >> 0) & 0x00FF); // CRC 값의 하위 바이트
RequestData[7] = (unsigned char)((crc >> 8) & 0x00FF); // CRC 값의 상위 바이트
```

연산을 거친 crc 값은 아래와 같습니다.

```
crc = 0xC915 (dec : 51,477)
RequestData[6] 변수에는 0x15 값이 저장됩니다. (CRC 값의 하위 바이트)
RequestData[7] 변수에는 0xC9 값이 저장됩니다. (CRC 값의 상위 바이트)
```

CRC를 포함한 전송할 데이터 준비는 끝났으며, RequestData 배열 0~7 의 값을 전송하면 됩니다.

수신 측(모듈)에서는 (RequestData[6], RequestData[7] 의 데이터) 와 (수신받은 데이터의 crc 연산값)을 비교한 후, 일치하면 응답, 일치하지 않으면 응답을 하지 않습니다.

반대로 마스터는 센서로부터 수신된 데이터를 위와 같은 연산을 거쳐, 데이터에 오류가 없는지 반드시 검증한 후 온도 데이터를 활용해야 합니다.

## ▶ 자주 묻는 질문(필독)

### Q1. 정확도 보장거리가 짧은데 보다 원거리에서 사용이 가능한가요?

A1. 정확도는 12cm에서  $\pm 0.3^{\circ}\text{C}(\text{typ})$  를 보장합니다만, 거리가 멀어질수록 제품별 오차가 발생합니다. 따라서 보다 원거리에서 사용하고자 할 경우, 특정 거리에서의 수치를 측정해 본 후 오차를 보정하는 방법을 사용해야 합니다.

예를 들어 12cm에서 측정치보다 50cm거리에서 0.6도가 낮았다면 50cm거리에서는 0.6도를 보상해 주면 됩니다. 다만 옅셋 개념으로 모든 온도 구간에 0.6도를 더해 주는 것보다는, 온도 구간을 쪼개서 구간별 보상 수치를 달리 적용하면 좀 더 정확한 수치에 근접할 수 있으니 참고 바랍니다.

다만 위 방법을 적용하기 위해서는 측정 대상과의 거리를 판단할 수 있는 환경이어야 합니다.

### Q2. 실외에서 사용이 가능한가요?

A2. 실외에서 사용할 경우 일차적으로 센서에 직사광선이 직접적으로 미치게 되어 측정치에 오류가 발생합니다. 허공(하늘방향)을 바라보게 설치 해서도 안됩니다. 그리고 한여름이나 겨울 같은 경우는 실외 온도가 제품 동작에 적합한 온도가 아니기 때문에 측정값에 오차가 발생합니다. 부득이 하게 건설 현장 같은 실외에서 사용할 경우에는 상온이 유지되는 별도의 컨테이너 박스 등 가급적 내부에서 제품을 설치/ 사용해야 합니다. 또한 히터나 에어컨을 가동한 직후 급격한 실내 온도 변화가 이뤄지고 있는 상황에서의 사용도 오차가 발생할 수 있습니다.

어느 정도 온도 안정화(열평형)가 이뤄진 이후에만 정확한 온도를 얻을 수 있습니다.

### Q3. 측정 시간은 어느 타이밍에 이뤄져야 하나요?

A3. IR refresh 는 10Hz, 즉 1초에 약 10번 통신이 가능하지만, 정확한 온도 반영에는 1초가 걸립니다. 따라서 정확도가 보장되는 온도의 측정 시간은 1초~2초 사이의 데이터 입니다.

■ 대상 접근 → 1초 wait → Data Read → 2초 전까지의 Data 활용

측정 시작 이후에는 측정 대상이 멀어지거나 움직이면 안됩니다.

### Q4. 센서 보호를 위해 센서 앞에 투명한 아크릴이나 유리등으로 막아도 될까요?

A4. 투명한 재질이라도 적외선 통과에 영향을 주기 때문에 어떠한 물리적 차단물 하면 안됩니다.

### Q5. 제품을 손으로 들고 측정해도 되나요?

A5. 반드시 Bracket등을 활용하여 고정해서 사용하시기 바랍니다. 손으로 제품을 잡고 있으면 결국 손의 열이 센서에 전도 되기 때문에 열충격으로 인한 정확도에 오차가 발생할 수 있습니다.

**Q6. 소비전류를 줄이고자 제품을 ON/OFF 해가면서 측정해도 될까요?**

A6. 센서는 전원이 공급되면 제품 안정화 시간이 필요합니다. Start-up time 은 최소 1초, 25°C 기준 안정화 시간은 최소 1분 이상이 필요합니다.  
공급 전원은 항상 유지될 수 있도록 해주세요.

**Q7. 제품 설치시 고려해야 할 주의 사항이 있나요?**

A7. 제품이 바라보는 방향에는 어떠한 열원(히터, 에어컨, 전자레인지, 햇빛이 들어오는 창문 등)이 있으면 안됩니다. 만약 열원이 있다면 배경의 열원에 따라 열이 있다고 감지할 수 도 있습니다. 가급적 최소 5m 이내에는 위 열원이 없게끔 하시고 부득이 하게 설치를 해야 할 경우에는 대상이 측정 범위에 있는지 판단 후에 그때의 데이터만 활용하시기를 바랍니다.

- 창문을 통해 제품에 직사광선이 내리쬐면 안됩니다. 창가 옆에는 설치하지 마세요.

## ▶ 청소 및 관리

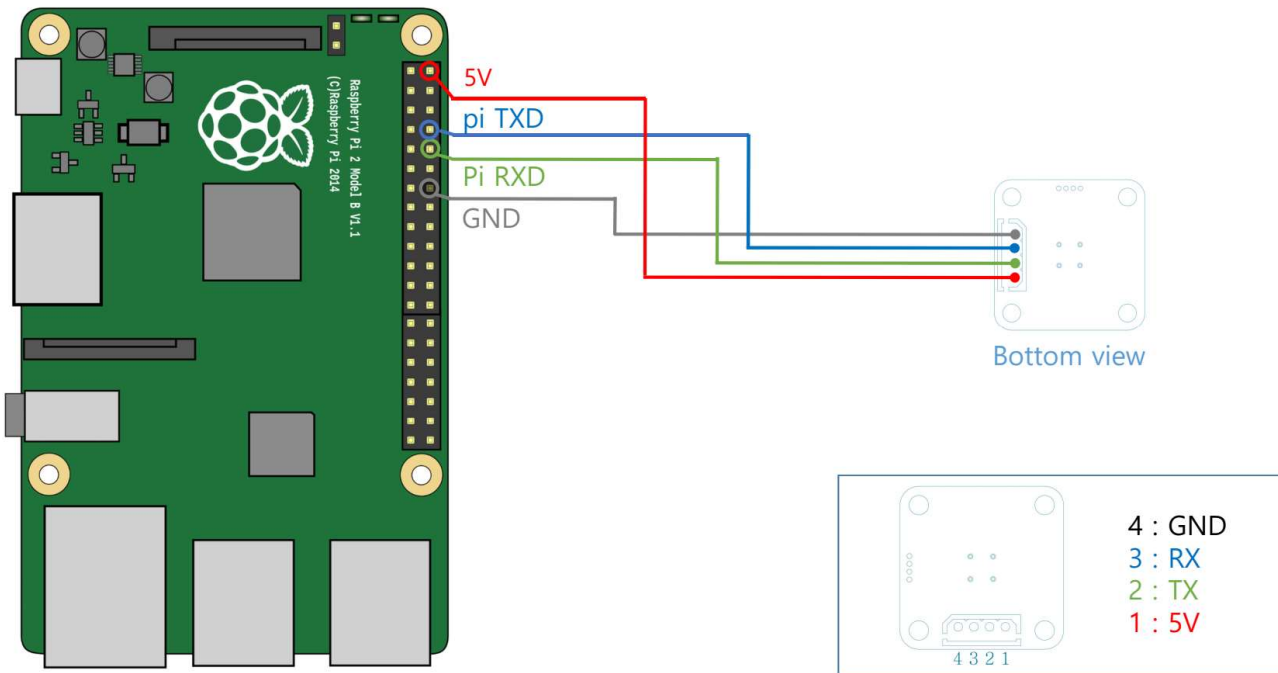
- ※ 제품의 렌즈가 오염되면 오염원에 따라 예상치 못한 온도가 나올 수 있습니다.
- ※ 렌즈 오염시 청소(일반 먼지)
  - 부드러운 천(안경 천)으로 파손되지 않게 주의하여 이물질을 제거하세요.
  - 청소 후에 바로 온도 측정을 하면 온도가 부정확할 수 있습니다.
  - 청소 후에는 상온에서 최소 30분 정도 보관 후 측정을 시작하세요.
  - 가벼운 먼지 정도는 카메라 먼지 제거용 에어브러시 등을 활용하세요.
  - 고압의 에어브러시는 사용을 금지합니다. 렌즈 파손의 우려가 있습니다.
- ※ 렌즈 오염시 청소(기름때)
  - 유증기가 발생하는 곳에서의 사용은 추천하지 않습니다.
  - 기름때의 정도에 따라 온도가 부정확해 질 수 있습니다.
  - 오염이 심하다면 광학 렌즈 전용 세척액을 사용하는 것을 추천합니다만, 힘주어서 닦아 내면 안됩니다. 렌즈 코팅이 손상될 수 있습니다.
  - 이소프로필 알코올(IPA) 이 광학 렌즈의 세정에 많이 쓰이니 참고하십시오.
  - 에탄올, 메탄올, 아세톤, 그 외 렌즈 코팅을 손상시킬 수 있는 산업용 물질은 사용을 금합니다.
  - 전용 세척액을 사용하더라도 오랜 기간 동안 반복 세척시 코팅이 벗겨질 수 있습니다. 렌즈 코팅이 손상되어 온도 차이가 심해지면, 신규 구매를 하셔야 합니다.
- ※ 특정 조건(거리에 따른 온도 보상)으로 설정하여 사용할 경우, 항상 같은 조건이 유지 되어야 합니다.
- ※ 독성 화학 물질이 있는 곳에서의 사용을 금합니다. 렌즈 코팅 및 제품이 부식될 수 있습니다.
- ※ 태양을 바라보면서 측정하면 안됩니다. 직사광선에 제품이 노출되지 않도록 하세요.
- ※ 방수가 되지 않으니 물이 튀는 환경 또는 비를 맞는 곳에서의 사용을 금합니다.
- ※ 전자파 간섭을 최소화 하기 위해서 가급적 모터나 generator 와 가까운 곳에 설치하지 마십시오.
- ※ 외력에 의한 제품 파손, 임의 개조 사용, 잘못된 회로 연결로 인한 내부 보드 파손 등 주의사항을 지키지 않아 발생한 문제에 대하여 AS 는 불가능 하오니 유의 바랍니다.

## ▶ 케이스 설계시 주의 사항.

- ※ 제품 케이스 제작시 주의 사항을 안내합니다.
- ※ 모듈의 히트싱크(센서 부분)가 케이스에 가급적 닿지 않게 설계하세요.  
구조상 반드시 밀착을 해야 한다면 고무 재질의 평와셔를 케이스와 히트싱크 사이에 위치하도록 하여 센서가 케이스에 직접 닿지 않도록 하세요.
- ※ 모듈 고정은 반드시 제품 PCB의 홀을 이용하세요.
- ※ 케이스 내부에 열이 많이 나는 부품을 가급적 같이 배치하지 마십시오.  
부득이하게 내부에 배치를 할 경우에는, 열이 나는 부품은 최대한 센서에서 멀리 위치해야 합니다.  
또한 해당 부품의 열이 센서에 미치지 않고 바로 빠져나갈 수 있도록 케이스 설계시 고려해야 합니다.
- ※ 센서 모듈 앞 부분은 어떠한 물리적 차단이 있으면 안됩니다. (유리, 아크릴 등등)
- ※ 케이스 및 센서부를 손으로 잡으면 안 된다는 문구를 제품에 꼭 삽입하여 안내해 주세요.
- ※ 가급적 통신선(UART) 길이는 1m를 넘지 않도록 하십시오. 선이 길 수록 외부의 노이즈 영향을 크게 받으며, 데이터 오류가 발생할 수 있습니다.  
설치되는 환경(외부 노이즈, 연결 케이블 종류, 전원부 안정성)에 따라 통신 가능 거리가 달라집니다.  
컨트롤러와 제품 설치 장소와의 거리가 수 미터 이상이 될 경우에는 CTM-3 제품을 구매하십시오.  
RS-485 통신이 기본 적용돼 있기 때문에 원거리 통신이 가능합니다.

## ▶ 라즈베리파이 예제 코드 지원

※ 쇼핑몰 상세 페이지에서 예제 코드 및 회로도 다운로드가 가능합니다.



### ※ 동작 화면

```

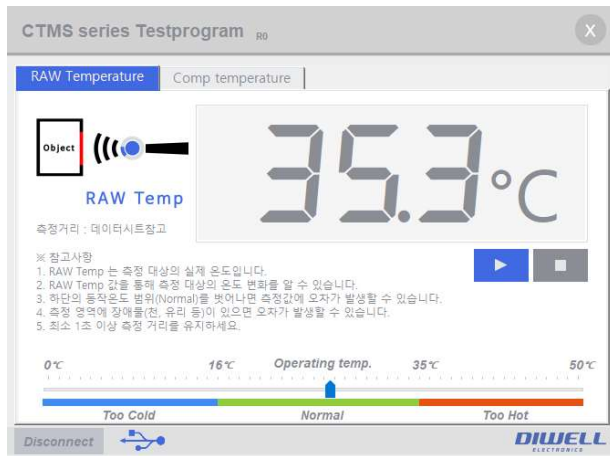
1- Raw Temp : 33.59  Comp Temp : 36.43  Sensor Temp : 29.99
2- Raw Temp : 33.61  Comp Temp : 36.43  Sensor Temp : 29.99
3- Raw Temp : 33.65  Comp Temp : 36.45  Sensor Temp : 29.99
4- Raw Temp : 33.69  Comp Temp : 36.47  Sensor Temp : 29.99
5- Raw Temp : 33.71  Comp Temp : 36.49  Sensor Temp : 29.99
6- Raw Temp : 33.71  Comp Temp : 36.47  Sensor Temp : 29.99
7- Raw Temp : 33.69  Comp Temp : 36.47  Sensor Temp : 29.99
8- Raw Temp : 33.69  Comp Temp : 36.47  Sensor Temp : 29.99
9- Raw Temp : 33.71  Comp Temp : 36.47  Sensor Temp : 30.01
10- Raw Temp : 33.73  Comp Temp : 36.49  Sensor Temp : 30.01
11- Raw Temp : 33.77  Comp Temp : 36.51  Sensor Temp : 30.01
    
```



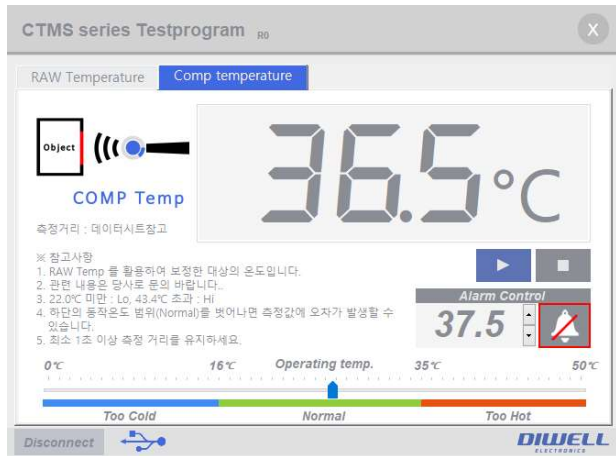
## ▶ CTMS series TestKit

- 온도센서는 설치 환경(거리, 면적, 움직임 여부 등)의 복합적인 영향을 받기 때문에 응용분야에 사용이 가능한지를 판단하려면 우선 사용 환경에서 측정을 해보아야 합니다. 하지만 이를 위해 통신 프로토콜을 분석하고 회로를 구현하는 시간/ 비용이 소요됩니다. CTMS testKit은 USB 연결만으로 PC에서 간편하게 온도 측정이 가능한 제품입니다. 제품 구매와 더불어 PC(윈도우10전용)에서 각 측정 모드에 따른 온도 확인이 가능한 윈도우 프로그램을 쇼핑몰 상세 페이지에서 무료로 다운로드 할 수 있습니다. 자세한 사항은 TestKit 매뉴얼을 확인하세요.

CTMS-3 TestKit : [http://diwellshop.com/product/detail.html?product\\_no=945](http://diwellshop.com/product/detail.html?product_no=945)



< RAW Data 출력 모드 >



< 보상 모드 >



< 보상 모드 - 고온 알람 >

## ▶ Additional Information

- manufacturer : Diwell Electronics Co., Ltd. <(주)디웰전자>
- Homepage: <http://www.diwell.com>
- Shopping mall : <http://www.diwellshop.com>
- Phone : 070-8235-0820
- Fax : 031-429-0821
- Technical support : [expoeb2@diwell.com](mailto:expoeb2@diwell.com), [dsjeong@diwell.com](mailto:dsjeong@diwell.com)
- 본 데이터시트 내용은 별도 공지 없이 변경될 수 있습니다. 최신 버전의 데이터 시트는 디웰전자 쇼핑몰 상세 페이지에서 확인 바랍니다.

## ▶ Revision History

Version	Date	Description
V1.0	2020-06-30	First version is released.
V1.1	2020-08-27	문구 수정