



&lt; DTPM11 &gt;



&lt; DTPM81 &gt;



&lt; DTPM151 &gt;

- 비접촉 온도 측정
- 방사율 조절
- 원거리 온도 측정
- High Accuracy
- Digital Interface : SPI

## ▶ 제품 설명

- DTPM Series는 접촉을 하지 않고 원하는 물체 표면의 온도를 500ms 이내에 정확하게 측정할 수 있는 온도센서모듈입니다.
- DTPM Series는 온도계산 프로세서를 내장하고 있어 정확한 온도 값을 출력합니다. (Master Controller에 온도계산 알고리즘이 필요하지 않습니다.)
- 방사율 조절이 가능합니다(초기값 0.95ε 값으로 출하).
- DTPM Series는 디지털 통신(SPI)으로 온도값을 출력합니다.
- 주변 온도와 대상 온도를 동시에 측정합니다.



## ▶ 특징

- DTPM11 측정 온도 구간 : -20 °C~100°C
- DTPM81 측정 온도 구간 : -20 °C~200°C
- DTPM151 측정 온도 구간 : -20 °C~270°C
- 동작 온도 구간 : -20 °C ~ 70°C
- 분해능 : 0.1°C
- DTPM11 DS ratio : No lens(측정각 :100도)
- DTPM81 DS ratio : 8:1
- DTPM151 DS ratio : 15:1
- 정확도 : ±2%
- 입력 전압 : 2.4V ~ 3.6V
- 통신 인터페이스 : SPI

## ▶ 응용분야

- 과열방지 시스템
- 산업용 온도 측정 장치
- 체온 측정을 통한 인체 감지
- 가전기기

### ▶ Absolute Maximum Ratings

- Absolute Maximum Rating 값을 초과하는 조건에서 DTPM을 동작시킬 경우 치명적인 손상을 가할 수 있습니다.

Parameter	Symbol	Conditions	min	Typ	Max	Unit
Supply Voltage	Vcc	Measured Versus GND	-0.2		4.0	V
Storage temperature	Tstor		-40		85	°C
Operating temperature	Top		-20		70	°C

### ▶ Electrical Requirements

Parameter	Symbol	Conditions	min	Typ	Max	Unit
공급전압	Vcc	Measured versus GND	2.4	3.3	3.6	V
방사율(Emission Coefficient)	$\epsilon$		0.1		1	
공급 전류		Full ambient temp. range, Typical value, no output load		1.25		mA
SPI Clock					1	MHz
<b>INPUT High Level</b>			<b>2.1</b>		<b>3.6</b>	V
<b>INPUT Low Level</b>					<b>0.9</b>	V
<b>OUTPUT High Level</b>			<b>VDD-0.3</b>		<b>VDD</b>	V
<b>OUTPUT LOW Level</b>			<b>VSS</b>		<b>VSS+0.3</b>	V

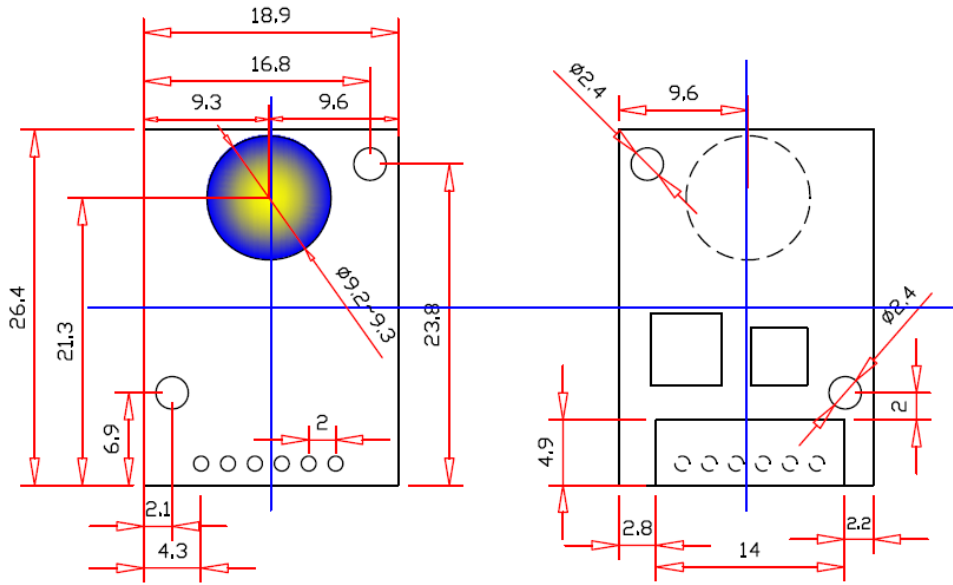
### ▶ Operational Characteristics

- if not otherwise noted, 25°C ambient temperature, 3.3V supply voltage and object with  $\epsilon = 0.95$  were applied

Parameter	Symbol	Conditions	min	Typ	Max	Unit
DS ratio : DTPM11 DTPM81 DTPM151				No lens 8:1 15:1		
온도측정범위: DTPM11 DTPM81 DTPM151	Tobj		-20 -20 -20		100 200 270	°C
동작온도(주변온도)	Tamb		-20		70	°C
온도측정 시간	Fout			0.5		sec
정확도	AccT			±2		%
Resolution Digital				0.1		°C
Standard Start-UP Time	tStart			3		sec
Stabilization Time	tStab			1		min

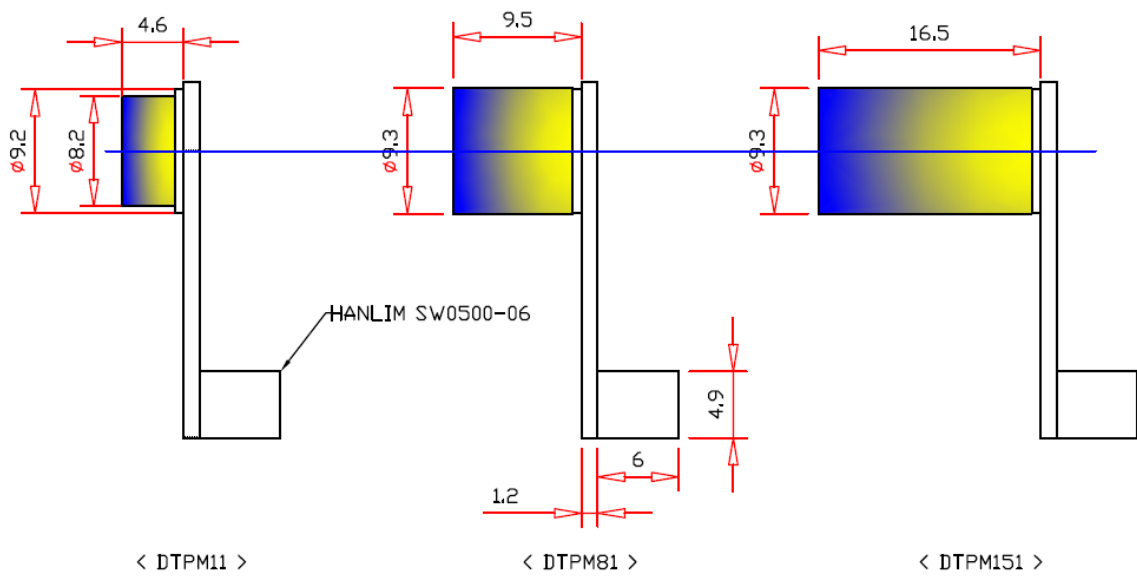
▶ Mechanical Dimensions

단위 : mm



<Top View>

<Bottom View>



< DTPM11 >

< DTPM81 >

< DTPM151 >

<Side View>

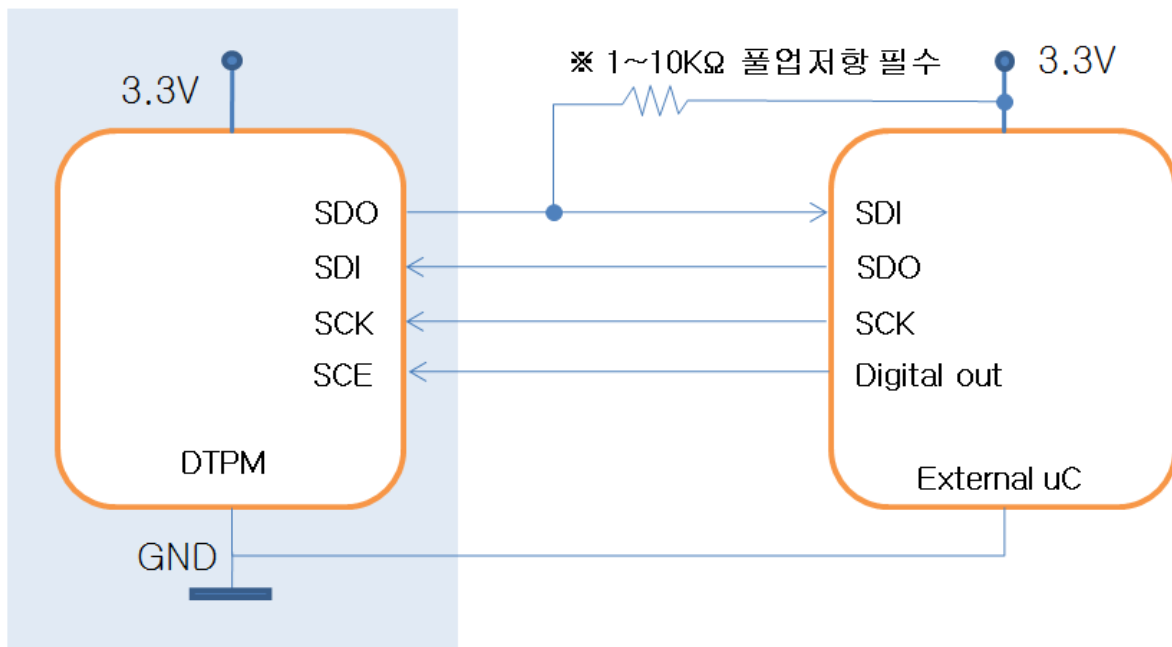
## ▶ Pin Assignment

number	Name	Description	Type
1	GND	Ground	Ground
2	SCE	ENABLE	Input
3	SCK	CLOCK	Input
4	SDI	Signal Input	Input
5	SDO	Signal Output	Output
6	VCC	Supply Voltage	Supply

- Connector : HANLIM SW0500-06( 반대편 Connector : HANLIM CH0500-06 )

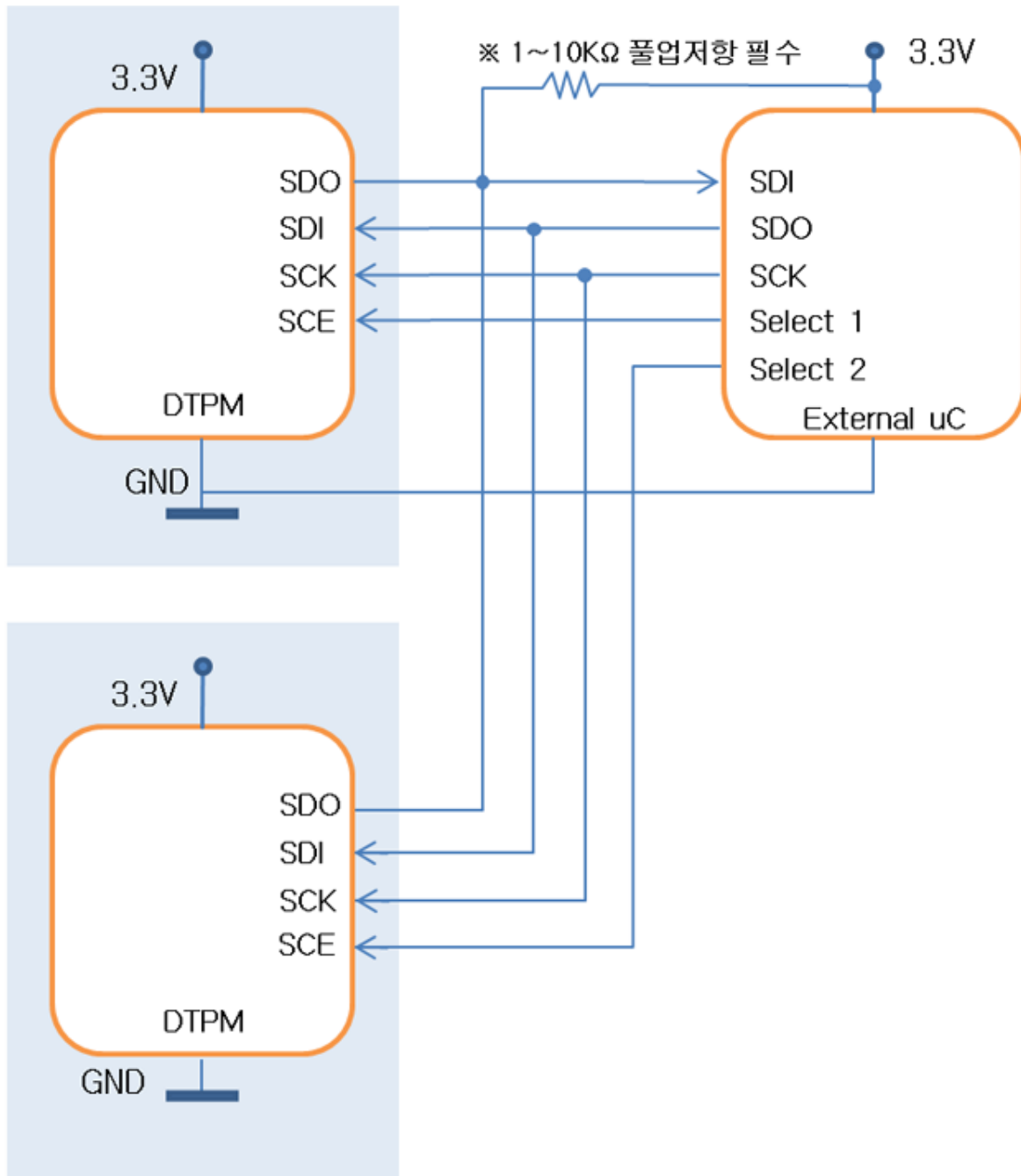
## ▶ 단일 모듈 연결방법 ( Single Module )

- Module 기준 SDO 포트에 반드시 풀업저항을 연결해 주세요. 아래 그림 참고.
- CT-Testboard-Plus 구매하신분은 테스트보드에 이미 풀업저항이 연결돼 있으니 참고하십시오.



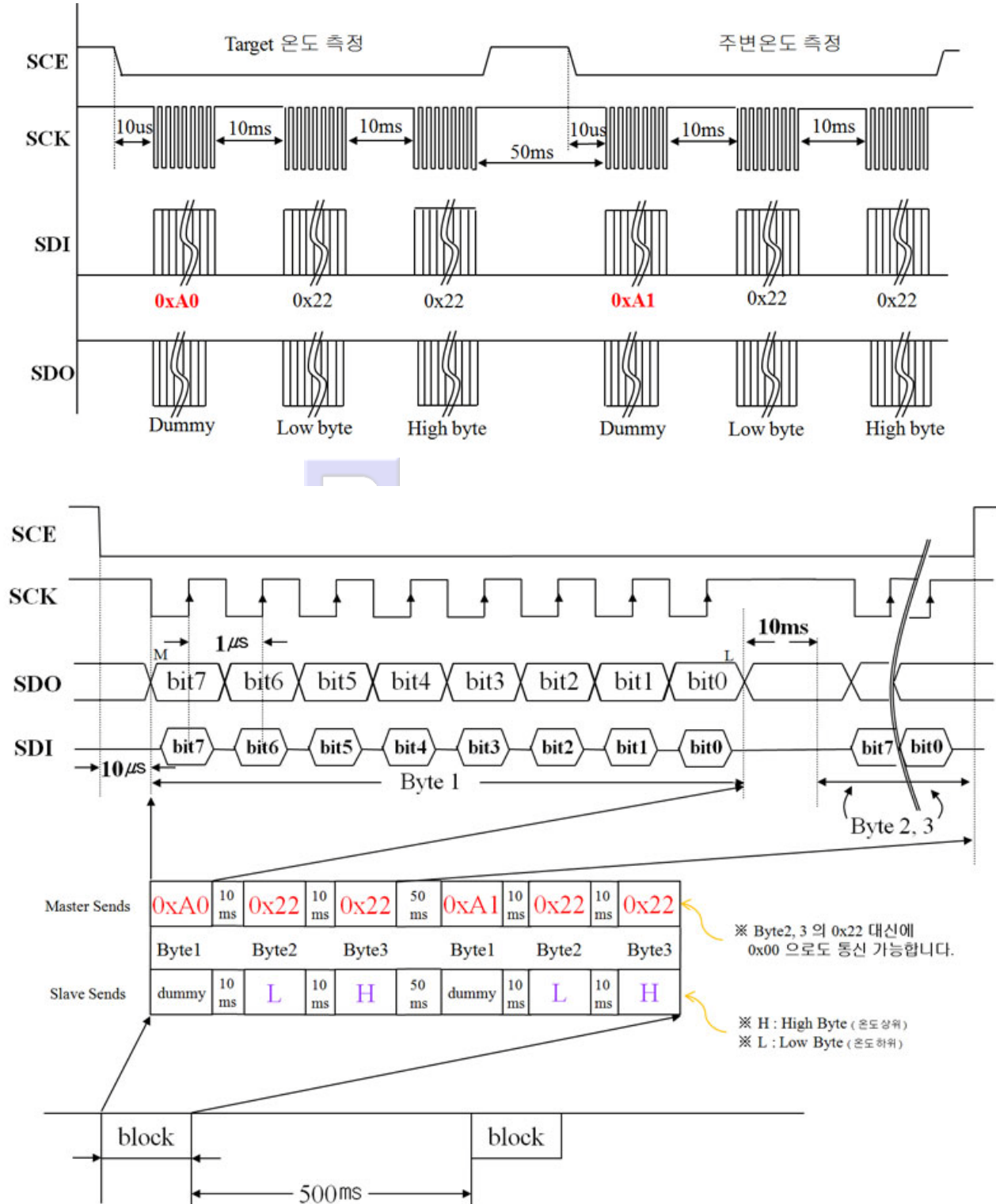
### ▶ 복수 모듈 연결 방법 ( Multiple Modules )

- Module 기준 SDO 포트에 반드시 **풀업저항**을 연결해 주세요.
- 모듈이 여러개라도 풀업저항은 한개만 연결하면 됩니다. 아래 그림 참고.

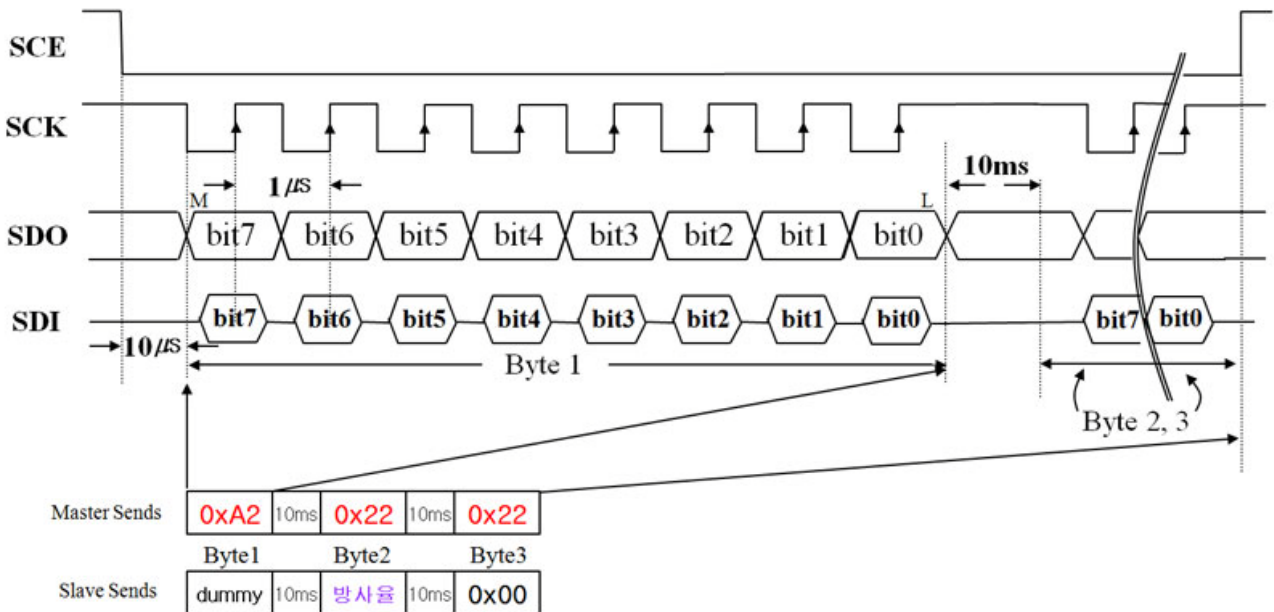
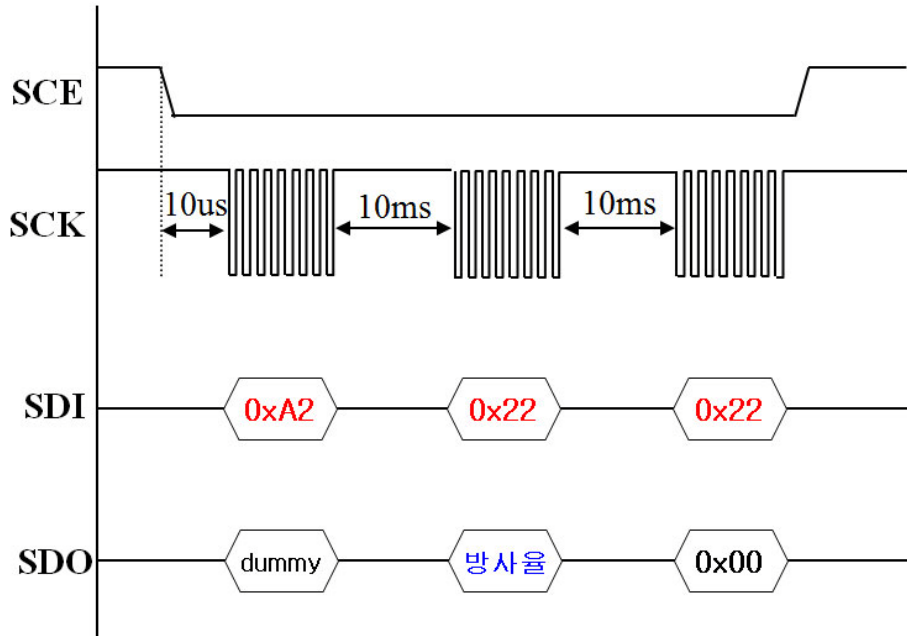


## ▶ SPI Communication and Timings :

### ● Target(대상) & Ambient(주변) 온도 읽기

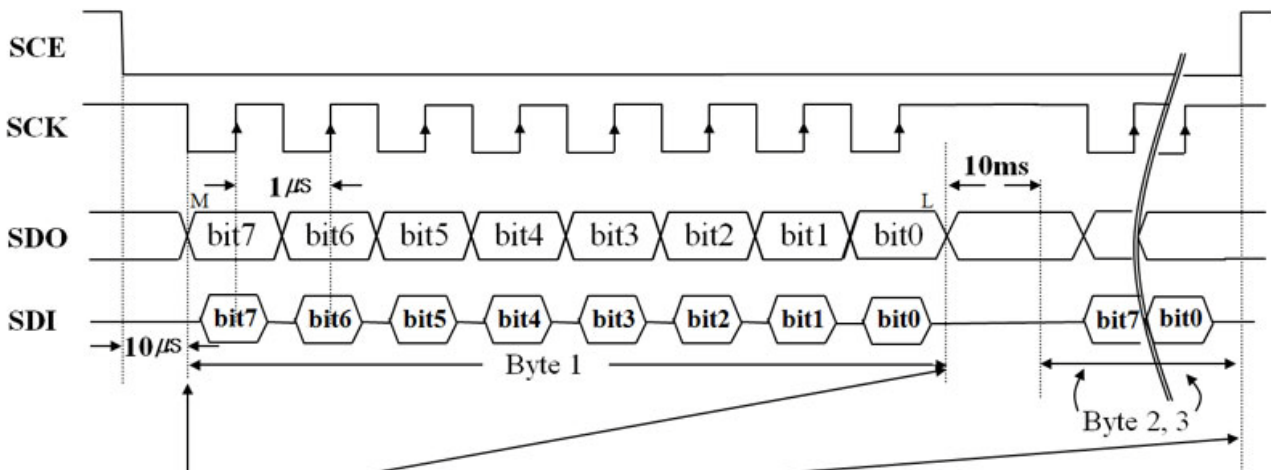
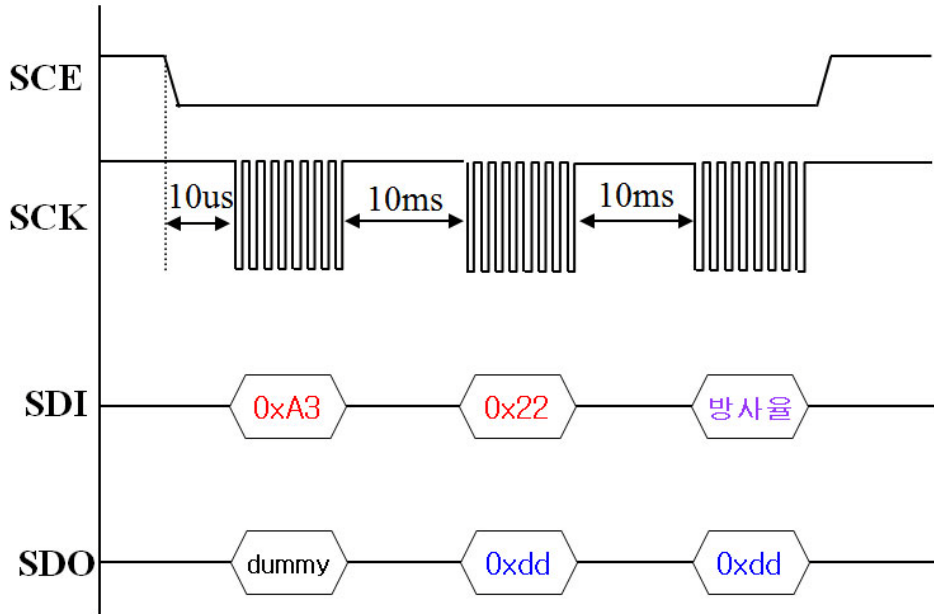


● 방사율 READ



※ 방사율 : range(1~100)  
 예) 방사율 0.95 는 95 로 표현됩니다.  
 방사율 0.83 는 83 로 표현됩니다.

● 방사율 WRITE



※ 방사율 : range(1~100)  
 예) 정수 95 를 write 하면 방사율 0.95 를 의미합니다.  
 정수 83 를 write 하면 방사율 0.83 을 의미합니다.

- range(1~100) 를 벗어난 값을 입력하면 방사율은 0.95로 리셋됩니다.  
 범위를 꼭 지켜주세요.

- 제품 출고시 기본 방사율은 0.95 입니다.



## ▶ 온도 계산 방법

- 영상온도 계산(주의 : 소수점 첫째 자리까지 출력 됩니다. 결과값에 10을 나눠주면 됩니다.)

Target	0x6D	0x01	주변	0xFA	0x00
--------	------	------	----	------	------

\* 타겟 온도 계산 : 상위 Byte(0x01) + 하위 Byte(0x6D) = 0x016D  
 => 365(HEX → 10진수) 즉 36.5 도입니다.

\* 주변 온도 계산 : 상위 Byte(0x00) + 하위 Byte(0xFA) = 0x00FA  
 = >250 (HEX → 10진수) 즉 25.0 도입니다.

- 영하온도 계산(영하(0도 미만)일 때는 2의 보수 값으로 전송됩니다.)

Target	0xF1	0xFF	주변	0x9C	0xFF
--------	------	------	----	------	------

\*타겟 온도 계산 : 상위 Byte(0xFF) + 하위 Byte(0xF1) = 0xFFFF1  
 0xFFFF1 = 1111 1111 1111 0001 (1의 보수 값 + 1 의 연산을 합니다)  
 0000 0000 0000 1110 □ 1의 보수값  
 0000 0000 0000 1111 □ +1 = 0x000F  
 0x000F = 15 즉, -1.5도 입니다.

\*주변 온도 계산 : 상위 Byte(0xFF) + 하위 Byte(0x9C) = 0xFF9C => 즉, -10.0도입니다.

## ▶ 방사율 READ

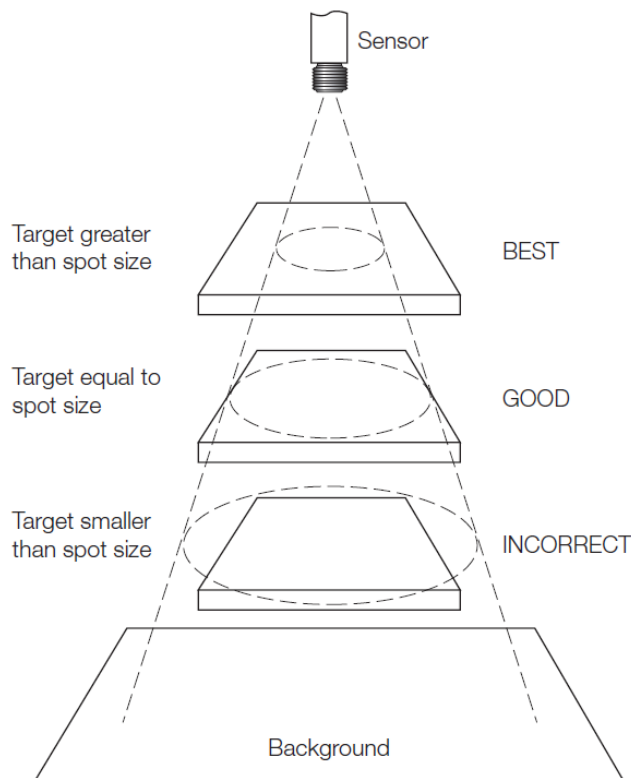
- DTPM 모듈의 방사율 설정을 읽어옵니다. 수치는 방사율 x 100 입니다.
- 예) 읽어온 값이 0x5F(95) 이면 방사율 0.95를 의미합니다.
- Request : 0xA2, 0x22, 0x22
- Response : dummy, 방사율, 0x00
- 데이터 범위 : 1~100
- page17. 방사율표 참고

## ▶ 방사율 WRITE

- DTPM 모듈에 방사율을 저장합니다. 수치는 방사율 × 100 입니다.
- 예) 방사율 0.95를 쓰고자 하면 0x5F(95) 값을 통신하면 됩니다.
- Request : 0xA3, 0x22, 방사율
- Response : dummy, 0xdd, 0xdd
- 데이터 범위 : 1~100
- page17. 방사율표 참고

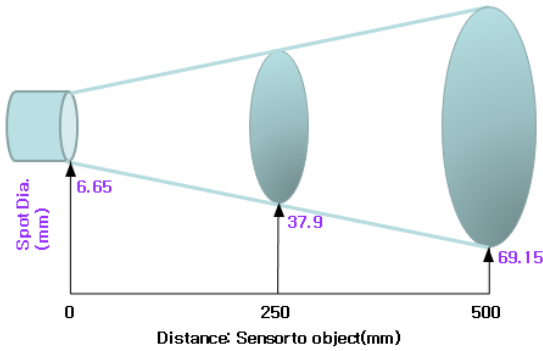
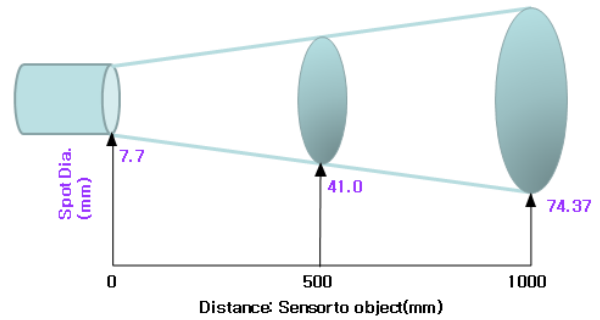
## ▶ DISTANCE AND SPOT SIZE

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



## ▶ Optical field of view (FOV)

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


**DTPM81**

**DTPM151**

## ▶ Ordering Guide

DTPM

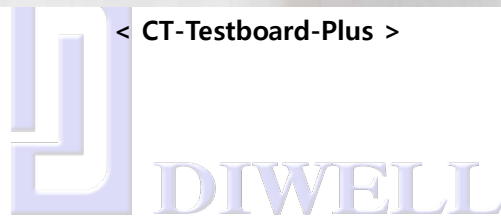
	DS ratio	측정 온도 구간
<b>11</b>	1 : 1	-20 °C ~ 100°C
<b>81</b>	8 : 1	-20 °C ~ 200°C
<b>151</b>	15 : 1	-20 °C ~ 270°C

## ▶ 테스트 보드(별도 구매) - CT-Testboard-Plus

- 손쉽게 측정하여 온도값을 다른 device(MCU, PC, embedded등) 로 전송(RS-232)할 수 있습니다.
- 기존 CT-Testboard 와 CT-Testboard-Plus 와의 차이점/ 주의사항
  1. 아래 그림에서와 같이 **Plus** 에는 **R7에 10K옴 풀업저항이 달려있습니다.**  
DTPM 시리즈 제품에는 풀업저항 연결이 필수
  2. DTPM의 방사율을 Read/Write 하는 기능.
  3. 구버전의 **CT-Testboard** 와는 호환되지 않습니다. 반드시 **Plus** 를 확인하세요.



&lt; CT-Testboard-Plus &gt;



## ▶ Additional Information

- manufacturer : Diwell Electronics Co., Ltd. <(주)디웰전자>
- Homepage : [www.diwell.com](http://www.diwell.com)
- shopping mall : [www.diwellshop.com](http://www.diwellshop.com)
- Phone : +82-70-8235-0820
- Fax : +82-31-429-0821
- Technical support : [expoeb2@diwell.com](mailto:expoeb2@diwell.com), [dsjeong@diwell.com](mailto:dsjeong@diwell.com)

## ▶ Appendix - A (Example Code - Sonix SPI 레지스터 이용)

아래 소스코드는 DTPM 모듈과 통신을 위한 참고용 소스 코드입니다.

### SPI 초기값 세팅

- Clock 주파수 최대 1Mhz
- Internal SPI Clock(Master Mode)
- SCK data transfer edge : Rising Edge
- MSB first data transfer
- SCK idle status : High

사용하는 MCU 환경에 따라 다르므로 내용을 이해 하신 후 적용하고자 하는 컴파일러에서 레지스터 설정해 주시면 됩니다. 레지스터 설정이 힘들다면 Appendix - B 의 IO 제어 코드를 참고하시길 바랍니다. 예제 코드는 SONIX사의 MCU 예제입니다. SPI 초기화 함수는 MCU마다 다르니 생략합니다. 전체적인 흐름을 참고하십시오.

```
long Check_Temp(unsigned char datum) // Sonix 컴파일러에는 Long 이 2byte 입니다.
{
    long temp_bank=0;
    SIOB = datum; // Buffer 레지스터에 저장
    NOP(1);

    EN_LOW; // Enable Low (중요)
    delay_us(10); // 10us delay (중요)
    FSTART = 1; // SPI 전송 시작
    while(FSTART==1); // 전송 완료 flag check

    buffer2=SIOB; // dummy
    SIOB = 0x22; // 0x22 or 0x00 가능합니다.
    delay_ms(10); // 10ms delay(중요)
    FSTART = 1; // SPI 전송 시작
    while(FSTART==1); // 전송 완료 flag check

    buffer=SIOB; // 하위 Byte 저장
    SIOB = 0x22; // 0x22 or 0x00 가능합니다.
    delay_ms(10); // 10ms delay(중요)
```

```
FSTART = 1;           // SPI 전송 시작
while(FSTART==1);    // 전송 완료 flag check

buffer1=SIOB;        // 상위 Byte 저장
EN_HIGH;            // Enable High (중요)

temp_bank=(buffer1*256)+buffer; // 상위, 하위 Byte 온도 계산식.
// temp_bank : 최종 온도

return temp_bank;
}

void main(void)
{
    Long Target_Value, Ambient_Value; //부호 있는 2byte 선언
    Port_init();           // PORT 초기화
    LCD_init();           // LCD 초기화
    EN_HIGH;             // CS idle High
    SPI_init();          // SPI 초기화

    while(1)
    {
        Target_Value = Check_Temp(0xa0); // SPI 통신(타겟온도)
        delay_ms(50);
        Ambient_Value = Check_Temp(0xa1); // SPI 통신(주변온도)
        LCD_view();           // 온도값 LCD 표시
        delay_ms(500);       // 500 delay(필수)
        // 온도값 read 명령 후 다음 명령까지 500ms delay
    }
}
```

## ▶ Appendix - B (Example Code - I/O)

하단 코드는 컨트롤러에서 SPI 레지스터 설정이 아닌 I/O 포트 제어를 통해 통신하는 예제 코드입니다. I/O 포트 전압 레벨이 3.3V 인지를 꼭 확인하세요.

- **SPI.H**

```
#define SCK_HIGH    FP16=1
#define SCK_LOW    FP16=0
#define SDO_HIGH   FP14=1
#define SDO_LOW    FP14=0
#define EN_HIGH    FP17=1
#define EN_LOW     FP17=0
long CHECK(unsigned char datum);
```

- **Main.C**

```
#include "SN8F27E65.h"
#include "delay.h"
#include "SPI.H"
Long Target_Value, Ambient_Value;
// sonix 컴파일러는 long이 2byte 입니다. 해당하는 컴파일러에 맞게 2byte변수 선언하세요
void main(void)
{
    While(1)
    {
        Target_Value = CHECK(0xa0); // 대상온도
        delay_ms(50);
        Ambient_Value = CHECK(0xa1) // 주변온도
        delay_ms(500);
        // LCD View CODE here
    }
}
```

- **SPI.C**

```

unsigned char buffer_Lo, buffer_Hi, p02; //1byte 선언

long CHECK(unsigned char datum) //2 byte return 함수
{
    unsigned char i=0;
    buffer_Lo=0;
    buffer_Hi = 0;
    EN_LOW;
    delay_us(10);
    for(i=0; i<8; i++)
    {
        if((0x80 >>i)&datum)==0){SDO_LOW;}
        else {SDO_HIGH;}
        SCK_LOW;
        delay_us(1);
        SCK_HIGH;
        delay_us(1);
    }
    SDO_LOW; // 0x22 or 0x00 전부 가능합니다만 코드간결을 위해 0x00 사용
    delay_ms(10);

    for(i=0; i<8; i++) // Low byte read
    {
        buffer_Lo = buffer_Lo <<1;
        SCK_LOW;
        delay_us(1);
        SCK_HIGH;
        delay_us(1);
        p02=FP02; // 포트의 상태 읽는 문장
        if(p02 == 1){buffer_Lo = buffer_Lo|0x01;}
        else{buffer_Lo = buffer_Lo&0xFE;}
    }

    SDO_LOW;
    delay_ms(10);

    for(i=0; i<8; i++) // High byte read
    {
        buffer_Hi = buffer_Hi <<1;
        SCK_LOW;
        delay_us(1);
        SCK_HIGH;
        delay_us(1);
        p02=FP02; // 포트의 상태 읽는 문장
        if(p02 == 1){buffer_Hi = buffer_Hi|0x01;}
        else{buffer_Hi = buffer_Hi&0xFE;}
    }
    EN_HIGH;
    return (buffer_Hi*256+buffer_Lo);
}
    
```



## ▶ Appendix - C (방사율표)

방사율이란 물체가 외부 적외선 에너지를 흡수, 투과 및 반사하는 비율을 말하는데, 이론적으로 외부에너지를 흡수만 하고 반사하지 않는 물체를 흑체라 하여 이때의 방사율은 "1"입니다. 하지만, 일반적으로 물체의 표면상태(광택, 거친, 산화여부 등) 에 따라 흡수, 반사하는 에너지량이 변합니다. 재질에 따른 방사율 값은 하단의 "방사율표" 를 참고하여 변경하면 됩니다. 단, "방사율표"의 값은 절대적인 값이 아니며 표면 상태와 그 외 복합적인 환경 요인에 따라 오차가 있을 수 있으니, 이점 참고하십시오.

대상	방사율	대상	방사율	대상	방사율
산화아연	0.1	에나멜	0.9	구리(연마된)	0.5
아연도금철	0.3	페인트	0.95	구리(산화된)	0.8
주석도금철	0.1	라 카	0.9	니켈(순수)	0.1
금(연마된)	0.1	고무(smooth)	0.9	니켈(산화된)	0.4~0.5
은(연마된)	0.1	고무(Rough)	0.98	니켈크롬	0.7
크롬(연마된)	0.1	플라스틱	0.8~0.95	니켈크롬(산화된)	0.95
붉은 벽돌	0.75~0.9	플라스틱필름	0.5~0.95	식물	0.9
흙	0.92~0.96	주철(연마)	0.2	피부	0.98
석면	0.95	Steel	0.6	가죽	0.75~0.8
콘크리트	0.7	산화 Steel	0.9	얼음	0.96~0.98
대리석	0.9	목재	0.8~0.9	모래	0.9
모르타르	0.89~0.91	스테인레스(연마된)	0.1	아스팔트	0.9~0.98
석고	0.85	스테인레스(기타)	0.2~0.6	유리	0.8~0.9
시멘트	0.96	알루미늄(연마된)	0.1	물	0.8~0.9
규토(정제된)	0.4	알루미늄(합금)	0.1~0.25	종이	0.9
세라믹	0.90~0.94	황동(연마된)	0.1	실리콘	0.7
석 영	0.9	황동(거친)	0.2	주철(부식된)	0.95
석 탄	0.75	황동(산화된)	0.6	Mild Steel	0.3~0.5
Fe(부식된)	0.7~0.85				

## ▶ DTPM Revision History

Version	Date	Description
1.0	2014-06-10	First version is released.
1.3	2014-09-11	Page11. FOV 그림 오류 수정

