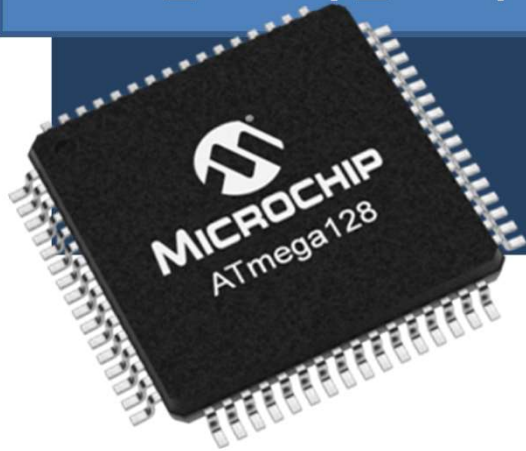


2023년 IoT기반 스마트 솔루션 개발자 양성과정



# Embedded Application

## 11-PSD Sensor

담당 교수 : 윤 종 이

010-9577-1696

[ojo1696@naver.com](mailto:ojo1696@naver.com)

<https://cafe.naver.com/yoons2023>

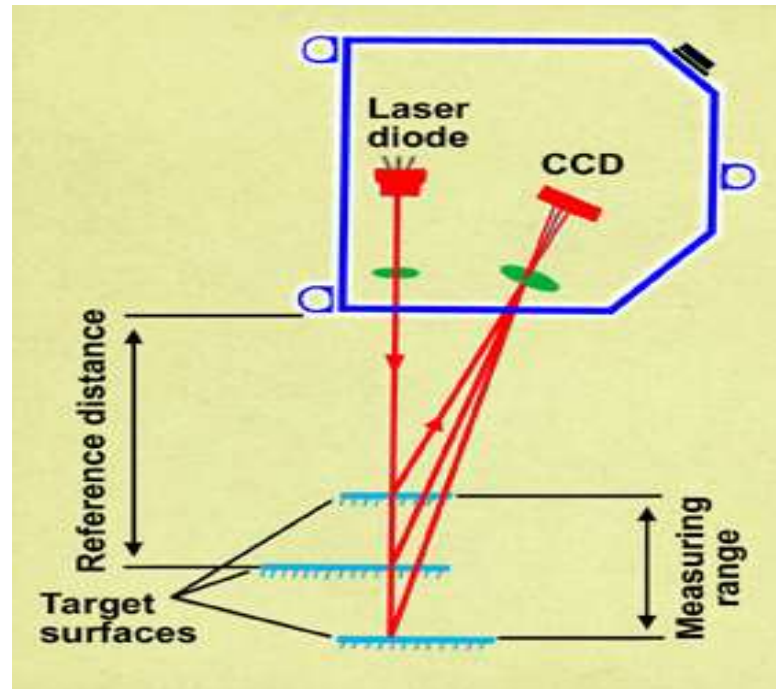


충북대학교 공동훈련센터

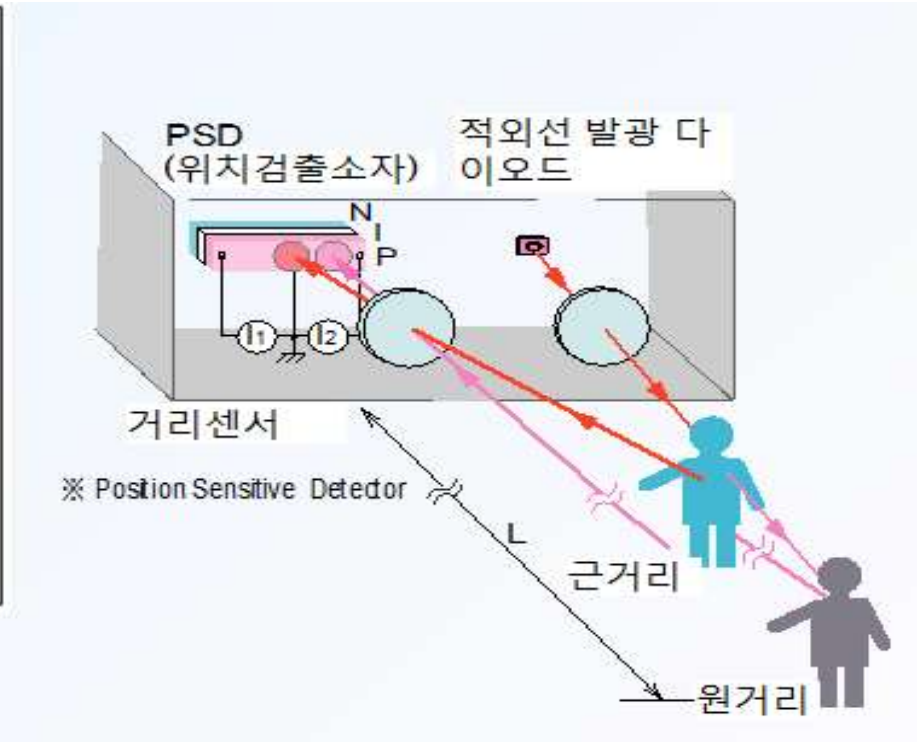
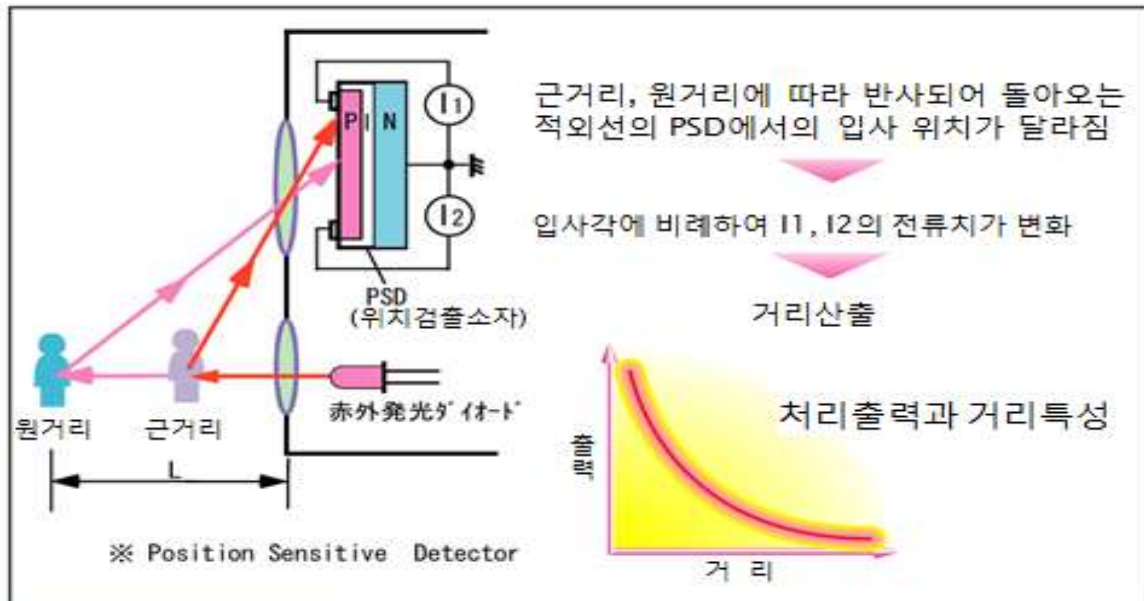
# 거리 감지 센서

- 거리 감지 센서 소개

- 거리감지 센서는 위치검출소자(Position Sensitive Device :PSD)를 사용
- 반도체 표면저항을 이용해서 1개의 PN접합으로 입사광의 단장거리 위치를 검출하는 반도체 소자
- 적외선을 물체에 발사하여 돌아오는 시간을 가지고 거리를 측정하는 센서

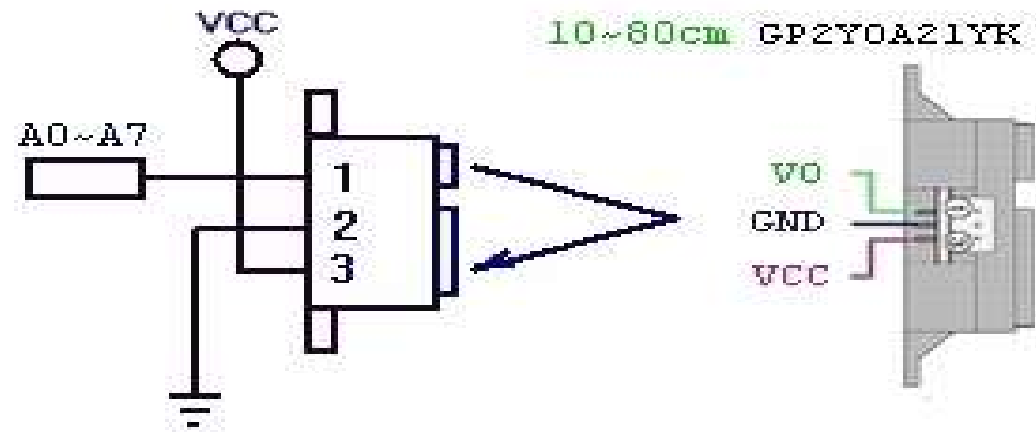


# 거리 측정 원리

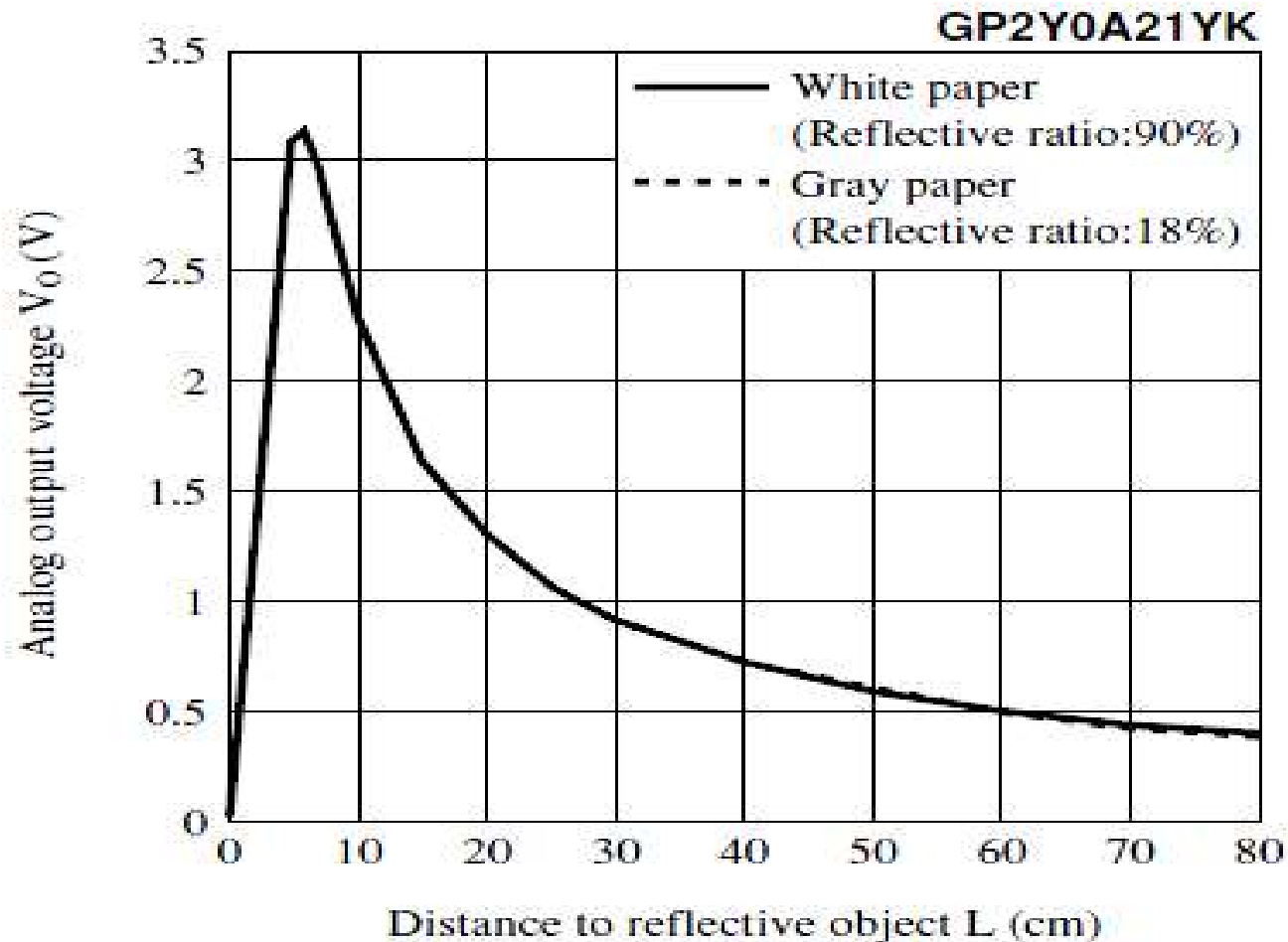


# PSD 센서(GP2Y0A21YK)

- 적외선 송신, 수신 소자와 IC를 내장하여 3~7V의 동작 전압에서 10~80cm의 거리를 검출
- Vo-아날로그 출력, VCC-구동전압, GND-그라운드
- 적외선 거리 센서의 출력은 10~80cm 거리에서, 최대 2.4V 에서 최소 0.25V로 반비례값을 출력
- 10cm 이하의 거리는 발광과 수광소자의 간격 문제로 검출이 안됨



# GY2Y0A21YK의 거리에 따른 출력 전압



# Linearization [선형화]

$$1 / (d + k) = a * ADC + b$$

**d** - distance in centimeters.

**k** - corrective constant (found using trial-and-error method)

**ADC** - digitalized value of voltage.

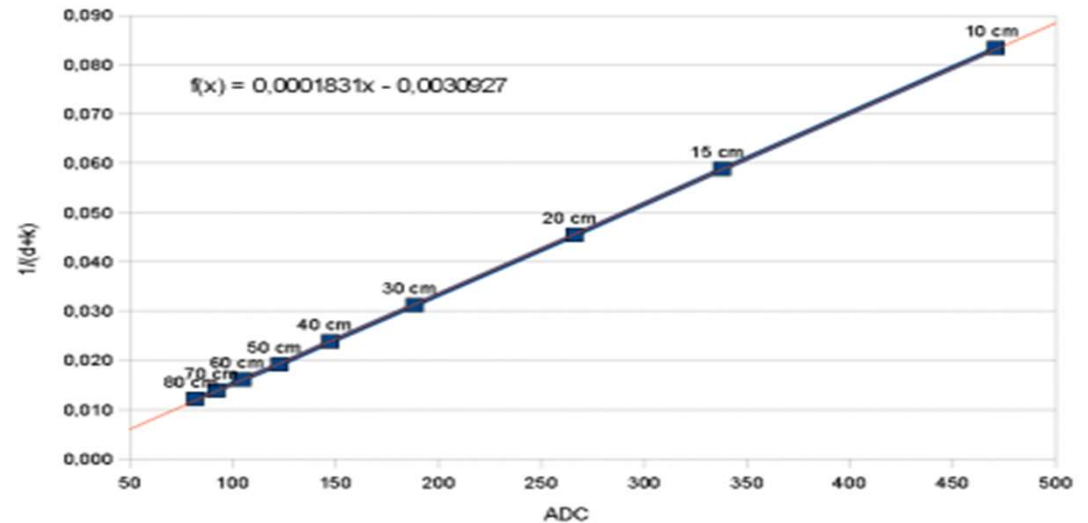
**a** - linear member (value is determined by the trend line equation)

**b** - free member (value is determined by the trend line equation)

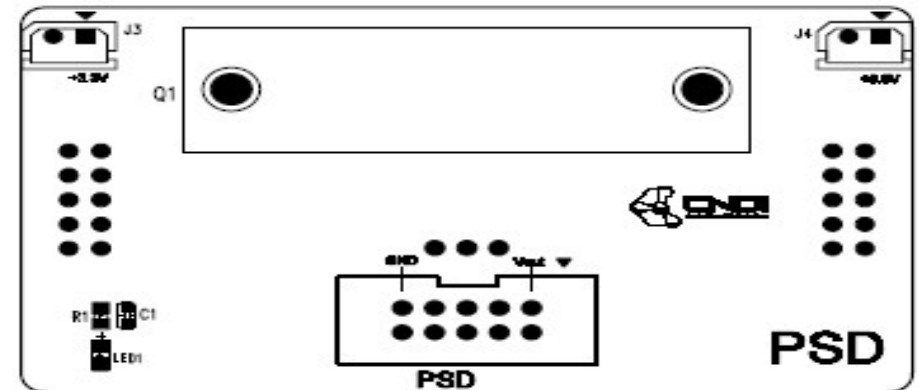
$$d = (1 / (a * ADC + b)) - k$$

$$d = (1 / a) / (ADC + b / a) - k$$

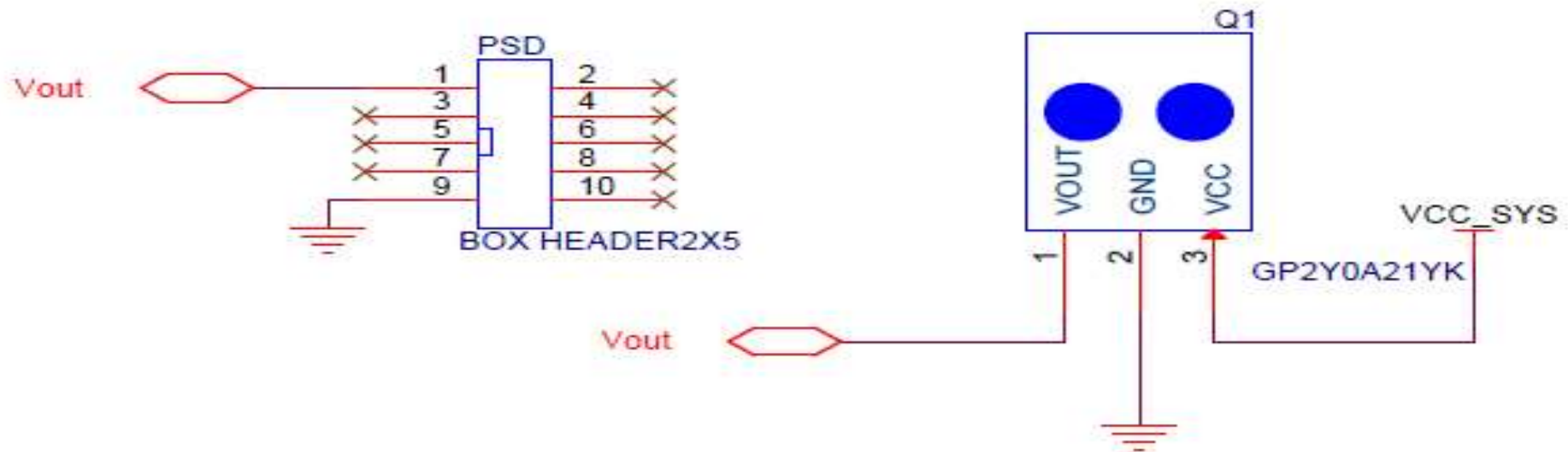
$$d = 5461 / (ADC - 17) - 2$$



# Sensor Module



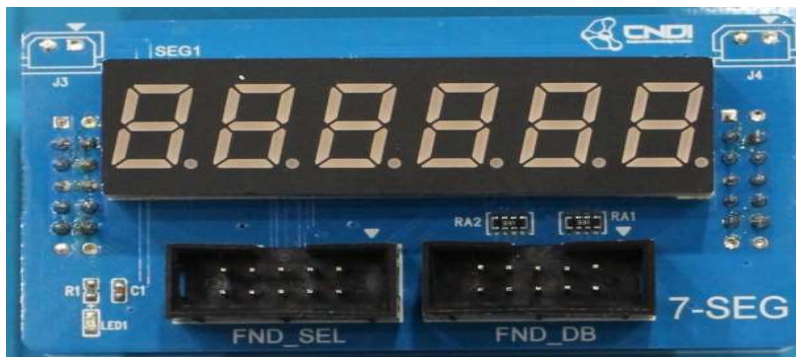
# Sensor Circuit



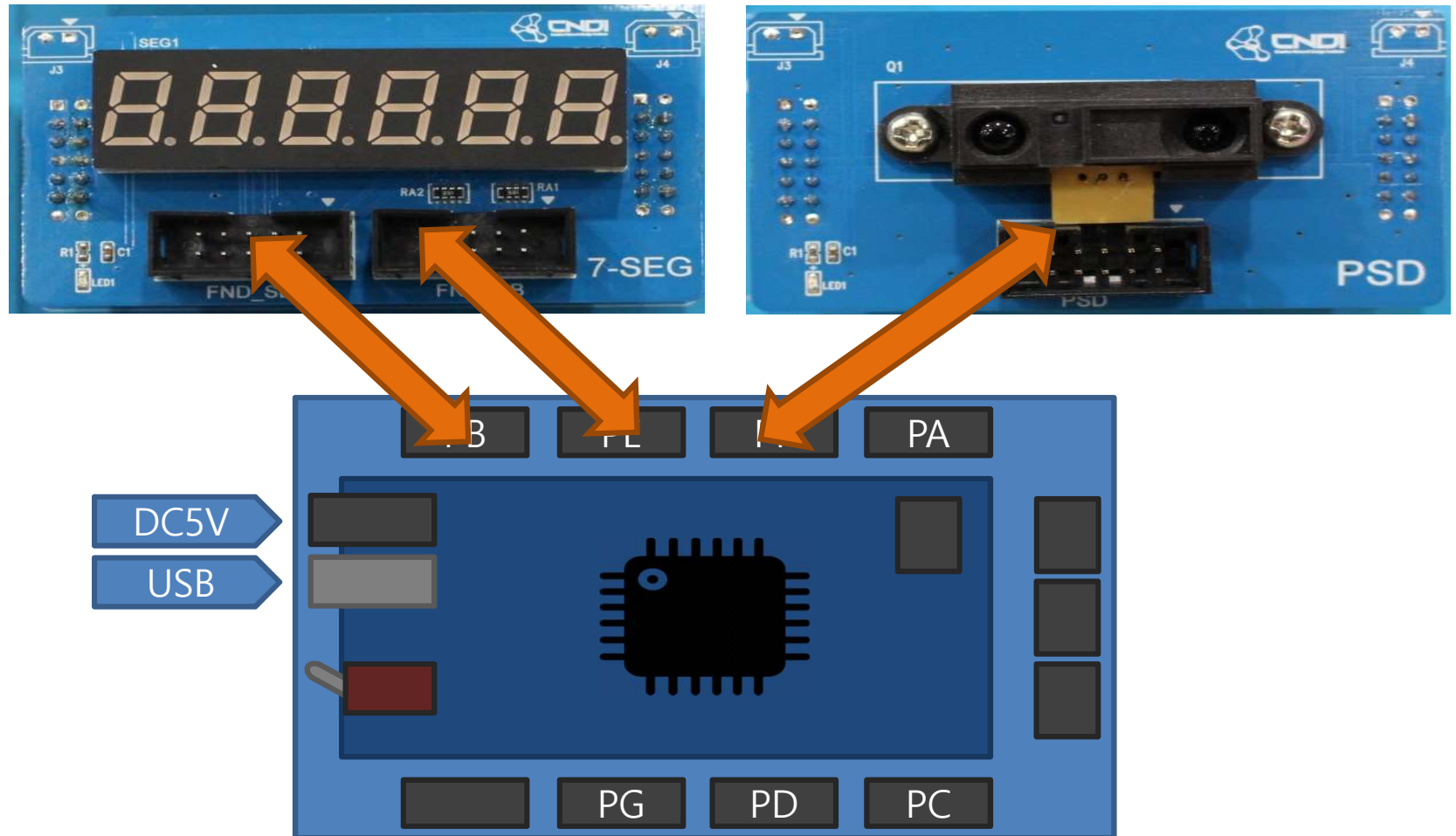


# Ex-1 : 거리 측정

- PSD 센서로 거리를 측정하여 FND에 표시하여 보자



# Ex-1 : Wiring



# Ex-1 : Define

```
#define F_CPU 14745600UL
#define FND_SEL PORTB
#define FND_DB PORTE
#define dTime 3

#include <avr/io.h>
#include <util/delay.h>

unsigned char FND[17]={0x3f, 0x06, 0x5b, 0x4f, 0x66, 0x6d, 0x7d, 0x27, 0x7f, 0x6f, 0x77, 0x7c, 0x58, 0x5e,
0x40, 0x49, 0x40};
unsigned char DGT[6]={0xfe, 0xfd, 0xfb, 0xf7, 0xef, 0xdf};
unsigned char NUM[6]={0x00, 0x00, 0x00, 0x00, 0x00, 0x00};

unsigned int ADV=0;
unsigned int Distance=0;

unsigned int DistanceCalc(int Val){
    return 5461/(Val-17)-2;
}
```



# Ex-1 : sub

```
void Hex2Dec(unsigned int No){
    unsigned int tmp=No;
    NUM[5]=8;///'8'
    NUM[4]=16;///'-'

    NUM[3]=tmp/1000;
    tmp=tmp%1000;
    NUM[2]=tmp/100;
    tmp=tmp%100;
    NUM[1]=tmp/10;
    NUM[0]=tmp%10;
}

void FND_Display(){
    for (unsigned char k=0; k<6; k++) {
        FND_SEL=DGT[k];
        FND_DB=FND[ NUM[k] ];
        _delay_ms(dTime );
    }
}
```



# Ex-1 : main

```
void CPU_Setup( ) {  
    DDRB=0xff;  
    DDRE=0xff;  
    DDRF=0xF0;  
}  
  
int main(void) {  
    CPU_Setup();  
  
    while (1) {  
        ADV=ADConv(0);  
        Distance=DistanceCalc(ADV);  
        Hex2Dec(Distance);  
        FND_Display();  
    }  
}
```

