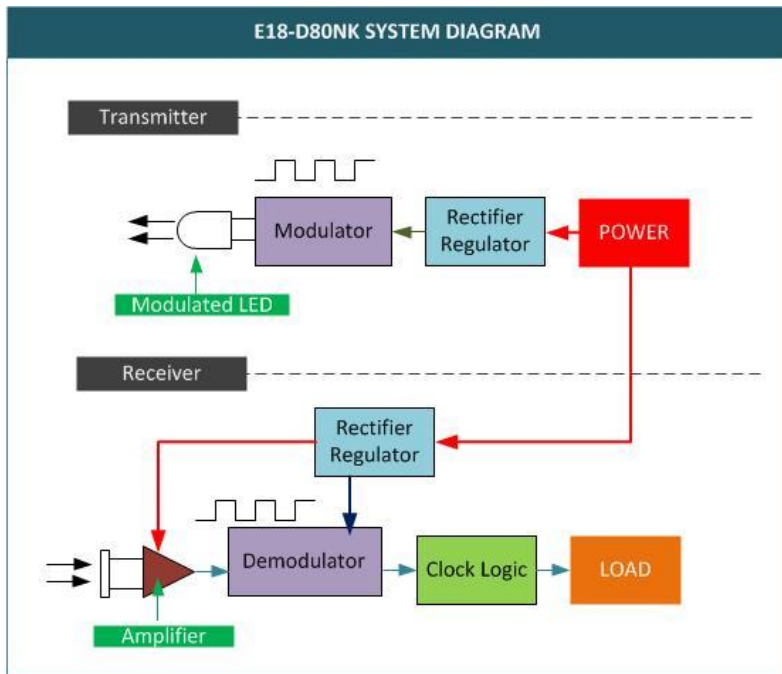


Wiring the E18-D80NK Infrared Distance Ranging Sensor

In this illustration we will going to wire the Infrared distance switch, a high-sensitive photo reflector to detect distance function, ranging from 3cm to 80cm. When the infrared emitted by the emitter it will get reflected on a surface blocked it then the phototransistor will pick up the signal for a distance calculation. This device has integrated with potentiometer to adjust the range for easy and clear to use. Best usage on this device is for robotics, interactive media, industrial and automotive, etc.

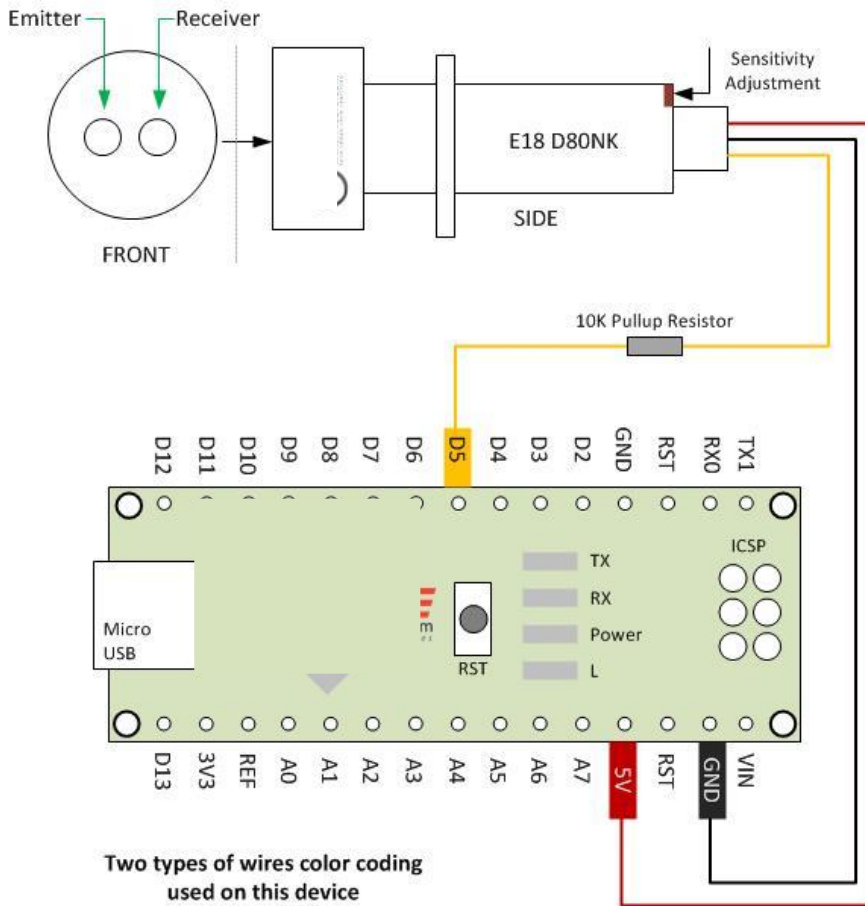


As you can see the diagram above the technology behind this effective distance sensor is the modulation from the infrared light, the receiver functionality is sense only if infrared light and only when it receives the correct signal. Therefore it is not very sensitive to ambient light compared to normal infrared detectors. Below are the wiring diagram for testing and how to use this device with Arduino MCU.







Device Characteristics

- Power Supply: 5VDC
- Supply current DC <25mA
- Maximum load current 100mA (Open-collector NPN pulldown output)
- Response time <2ms
- Diameter: 17MM
- Pointing angle: $\leq 15^\circ$, effective from 3-80CM Adjustable
- Detection of objects: transparent or opaque
- Working environment temperature: $-25^\circ\text{C}+55^\circ\text{C}$
- Case Material: Plastic
- Lead Length: 45CM

Wiring Diagram Schematics



Two types of wires color coding used on this device

| TYPE - 1 | TYPE - 2 |
|--|--|
|  +5 Vcc |  Gnd |
|  Gnd |  Vcc |
|  Signal |  Signal |

Placing a 4.7k to 10k Pullup Resistor to +5v

```
/*
E18-D80NK Infrared Distance Ranging Sensor
*/
```

```
void setup() {
  Serial.begin(9600); //Start serial communication boud rate at 9600
  pinMode(5,INPUT); //Pin 5 as signal input
}
```

```
void loop() {
  while(1) {
    delay(500);
    if(digitalRead(5)==LOW) {
      // If no signal print collision detected
      Serial.println("Collision Detected.");
    }
    else {
      // If signal detected print collision detected
      Serial.println("No Collision Detected.");
    }
  }
}
```