SmartElex Ambient Light Sensor Breakout - TEMT6000

Light sensors have all sorts of practical uses in the modern era, most notably in devices with auto-brightness for their screens and in digital cameras to adjust exposure. With the Ambient Light Sensor Breakout, it's be a breeze to interface with the TEMT6000 Light Sensor so you can bring the ability to detect light levels to any project.

As the name suggests, the TEMT6000 Light Sensor will detect the brightness of its surroundings. While there are many properties of light that can help us categorize its brightness, the TEMT6000 measures illuminance (measured in lux (lx), often denoted E_v). Don't worry if illuminance is new to you though, the TEMT6000 is very intuitive to use: brighter = more current, darker = less current.

In this guide, we'll show you how to quickly get the ambient light sensor breakout up and running, then discuss some of the more technical details of how it functions.

Hardware Overview

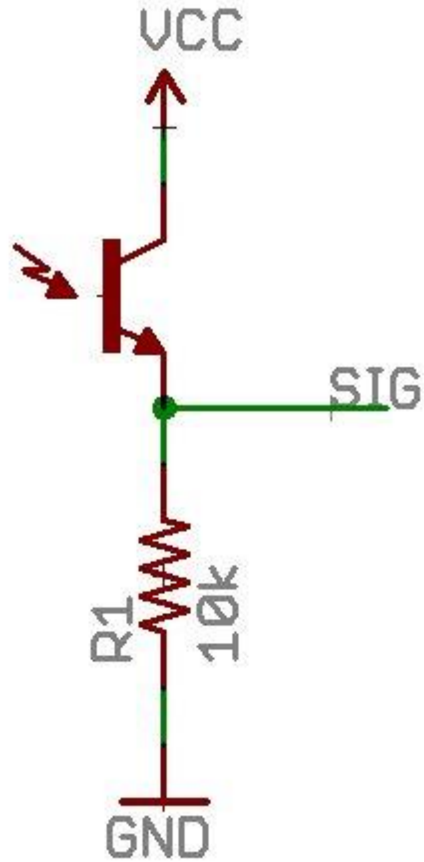
As you can see from the image below, the TEMT6000 is about as simple as it gets when it comes to breakout boards. The three pins broken out are labeled on the top of the board.The function of each pin can be found in the table below.

Symbol	**Description**
SIG	Output Voltage from the divider circuit
GND	GND (0V)
VCC	Collector Voltage (**should not exceed 6V**)

Being a phototransistor, this sensor acts just like any other NPN transistor -- the greater the incoming light on the **Base**, the the more current that can flow from the **Collector** to the **Emitter**. Only light that falls within the visible spectrum (**390–700 nm**) will alter the Base. Infrared, ultraviolet, or any other light we can't directly see will have no effect on the sensor.

This sensor can handle voltages from both **5V and 3.3V** devices.

To make taking light measurements as easy as possible, this sensor has been designed into a voltage divider circuit. The TEMT600 acts as one of the resistors in the divider, and, as the light hitting it changes, so too does the voltage on the SIG pin. To read that voltage, simply connect the SIG pin on the TEMT6000 to any analog to digital conversion pin on your microcontroller.



The TEMT6000 Breakout schematic.

The voltage value returned from the SIG pin will vary depending on what voltage is being used to power the sensor and depending on the resolution of your ADC.

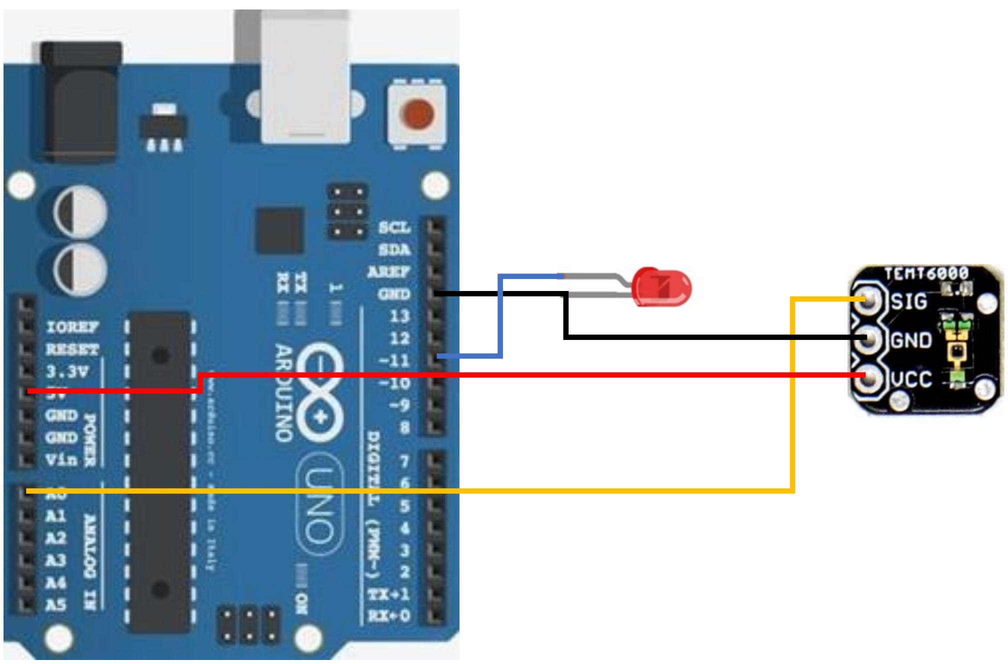
Wiring

We recommend soldering male header-pins to the breakout to make it simpler to prototype with:

The rest is just plugging parts into the UNO. Start by taking any basic LED of your preferred color and placing its anode leg (the long leg) into a pin whose label is followed by a `~`. This mark on the means the pin supports pulse-width modulation (PWM), which is just a technical way of saying we can control the voltage output of the pin digitally; in this case it lets us control the apparent brightness of our LED. We want this because our LED is going to show us the relative brightness of the world through the eyes of our TEMT6000. If you're using a different board, be sure to read its documentation closely to see which of its pins support PWM. Place the other leg into ground (`GND`).

We used pin 11 on the UNO because it is the PWM pin closest to a ground pin. Thus, we didn't have to deform our LED's legs too much. This is the pin we'll be using in the sample code, so be sure to modify that constant if you're using a different physical pin. Next comes the light sensor breakout. Start by connecting the female ends of some male-to-female jumper cables to the pins we soldered to it earlier. Connect VCC on the sensor to the 5V pin on the UNO, GND to GND, and SIG to any analog pin (we'll be using A0 in the sample code). Use the table below to aid in wiring.

Arduino	TEMT6000
A0	SIG
GND	GND
5V	VCC



Example Code:

```
#define LEDPIN 11         //LED brightness (PWM) writing
#define LIGHTSENSORPIN A0 //Ambient light sensor reading
```

```
void setup() {
  pinMode(LIGHTSENSORPIN,  INPUT);
  pinMode(LEDPIN, OUTPUT);
  Serial.begin(9600);
}

void loop() {
  float reading = analogRead(LIGHTSENSORPIN); //Read light level
  float square_ratio = reading / 1023.0;      //Get percent of maximum value (1023)
  square_ratio = pow(square_ratio, 2.0);      //Square to make response more obvious
delay(250);
  analogWrite(LEDPIN, 255.0 * square_ratio);  //Adjust LED brightness relatively
  Serial.println(reading);                    //Display reading in serial monitor
}
////////////////////////////////////////END//////////////////////////////////////////////
```
Once everything's uploaded, your LED should respond to the relative brightness that's exposed to the TEMT6000.

How Light Detection Works

Now that our sensor is working, let's take a more in-depth look at what is going on inside the senor. As mentioned earlier, the TEMT6000 measures illuminance. If you're unfamiliar with illuminance, it is a measure of the total quantity of visible light emitted by a source (referred to as luminous flux, measured in lumens (lm) divided by an area in square meters. More notationally, 1 lx = 1 lm/m². Along with these, there are other properties of light that are unfortunately all named using the same Latin root for light, so it can be hard to keep them straight. Here's a diagram to hopefully *elucidate* the differences:

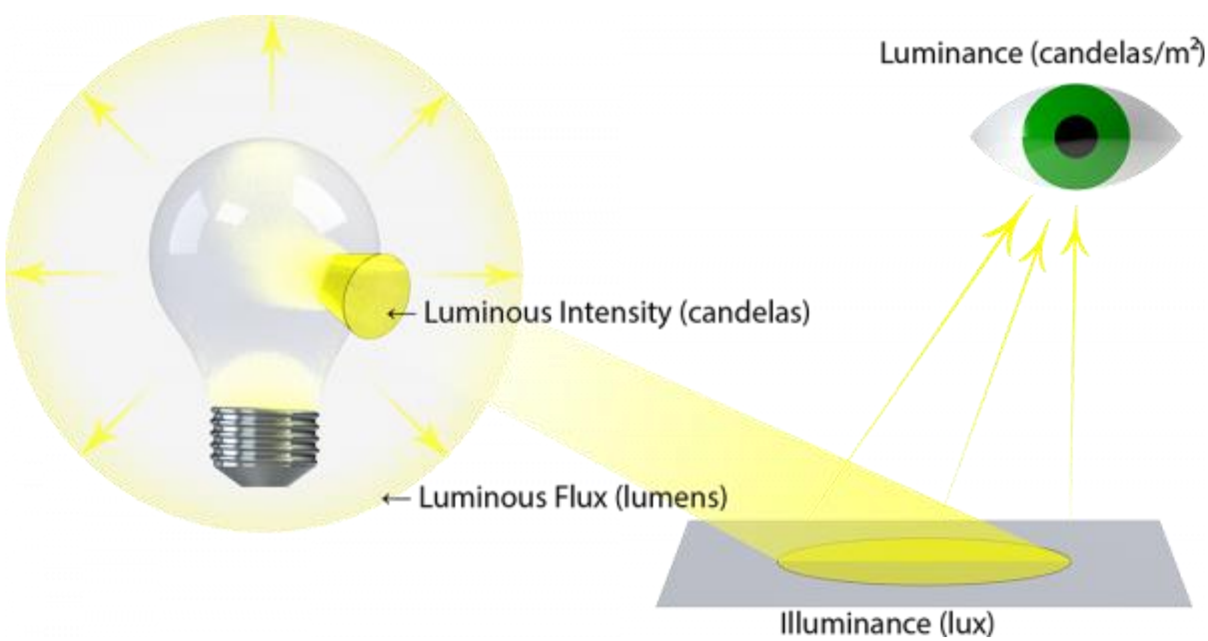


Diagram depicting the nuances between the various measurements of light.

Why does the TEMT6000 measure illuminance? In most practices, measuring the intensity of light without factoring in distance is very difficult, and puzzled early astronomers for a long time. In short, there is **apparent** magnitude (how bright a source appears) and **absolute** magnitude (how bright the source actually is). Two sources of different absolute magnitudes can have the same apparent magnitude depending on their distance from the observer.

For example, if you have a bright source far away and a dim source very close, they can appear to have the same brightness because the brighter source's light will have to dissipate over a larger volume. This is why the sensor will read a smaller value if you move the same source of light farther away from it, essentially increasing the amount of space that the same amount of light has to fill between the source and the sensor (i.e. reducing the illuminance, as you're dividing by a larger surface area of the light-sphere generated by the source).

Here is a graphical relationship between the current (in µA) and illuminance of the immediate vicinity perceived by the sensor:

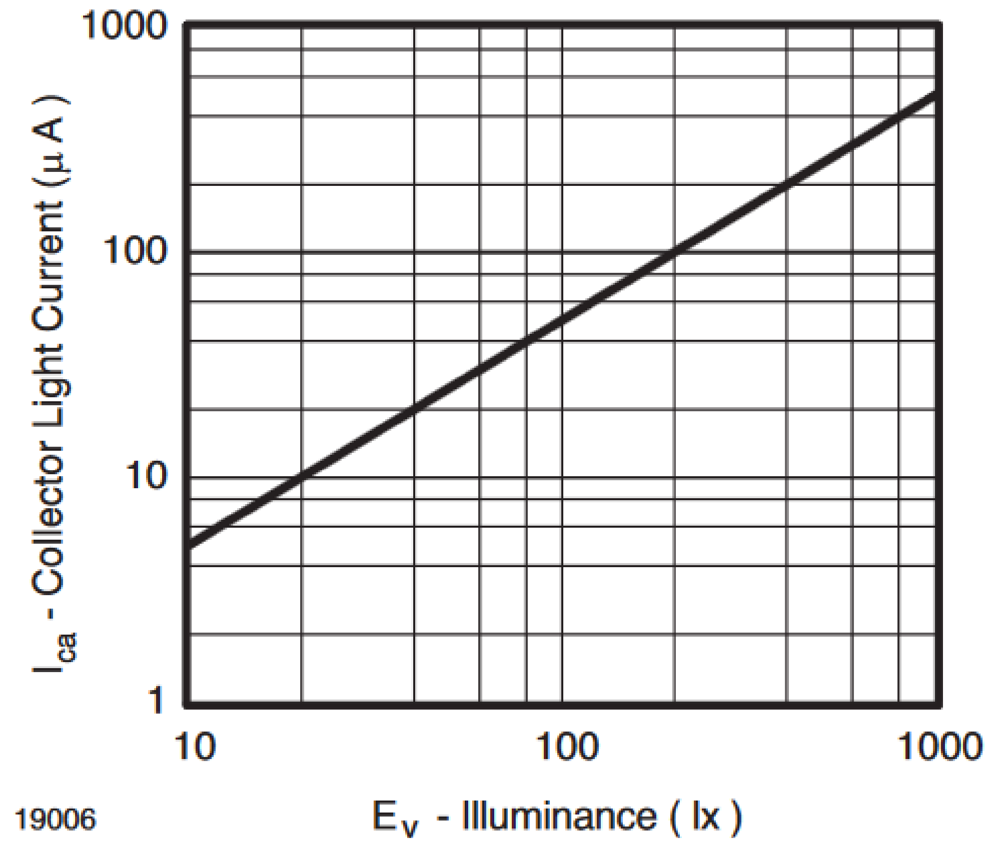


Figure 1. Collector Light Current vs. Illuminance

Found in the TEMT6000 datasheet.

The TEMT6000 only recognizes light with wavelengths in the range of **390–700 nm**, which roughly covers the entire spectrum of visible light. In other words, this won't pick up infrared, ultraviolet, or any other light we can't directly see.

Here's a table of the typical illuminance from common sources of visible light:

Examples	
Illuminance	**Surfaces illuminated by:**
0.0001 lux	Moonless, overcast night sky (starlight)[3]
0.002 lux	Moonless clear night sky with airglow[3]
0.27–1.0 lux	Full moon on a clear night[3][4]
3.4 lux	Dark limit of civil twilight under a clear sky[5]
50 lux	Family living room lights (Australia, 1998)[6]
80 lux	Office building hallway/toilet lighting[7][8]
100 lux	Very dark overcast day[3]
320–500 lux	Office lighting[6][9][10][11]
400 lux	Sunrise or sunset on a clear day.
1000 lux	Overcast day;[3] typical TV studio lighting
10 000–25 000 lux	Full daylight (not direct sun)[3]
32 000–100 000 lux	Direct sunlight

Lux Wikipedia article / CC BY

Now that we understand the TEMT6000 a little better, let's use it in something more interesting and build ourselves a night light that turns itself on and off!