PHOTORESISTORS AND HOW TO USE THEM

Professor Jon E. Froehlich
@jonfroehlich | jonf@cs.uw.edu

DESIGN USE BUILD

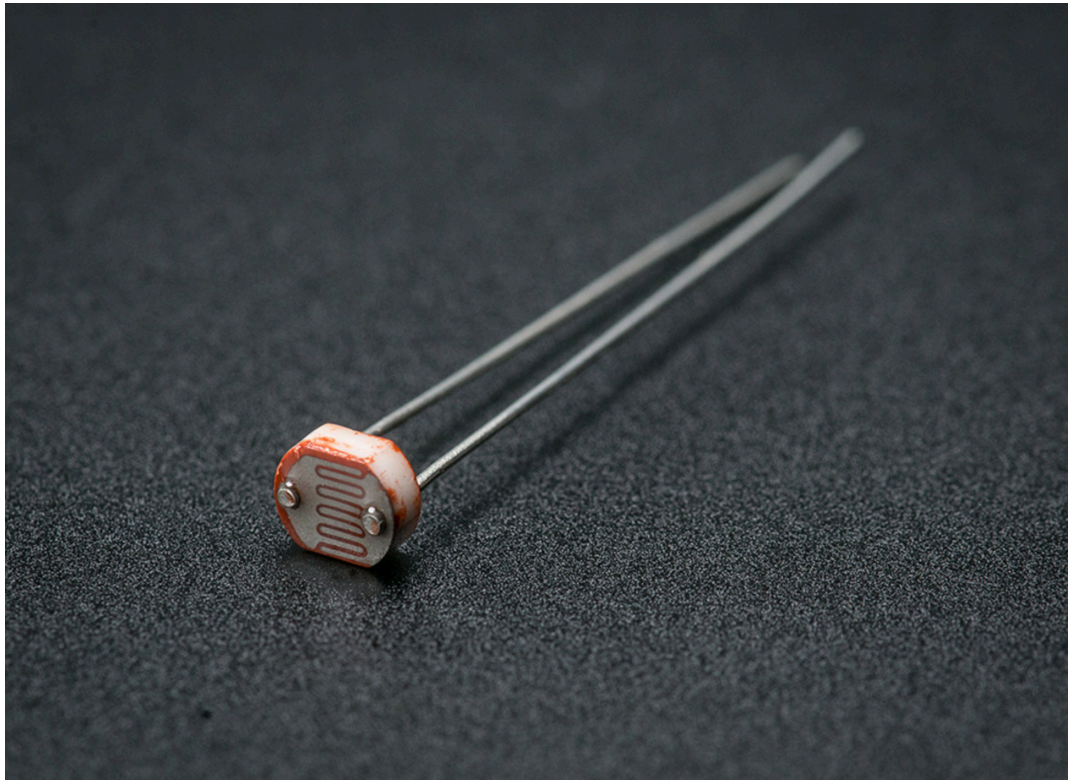
MAKEABILITY LAB

PAUL G. ALLEN SCHOOL
OF COMPUTER SCIENCE & ENGINEERING

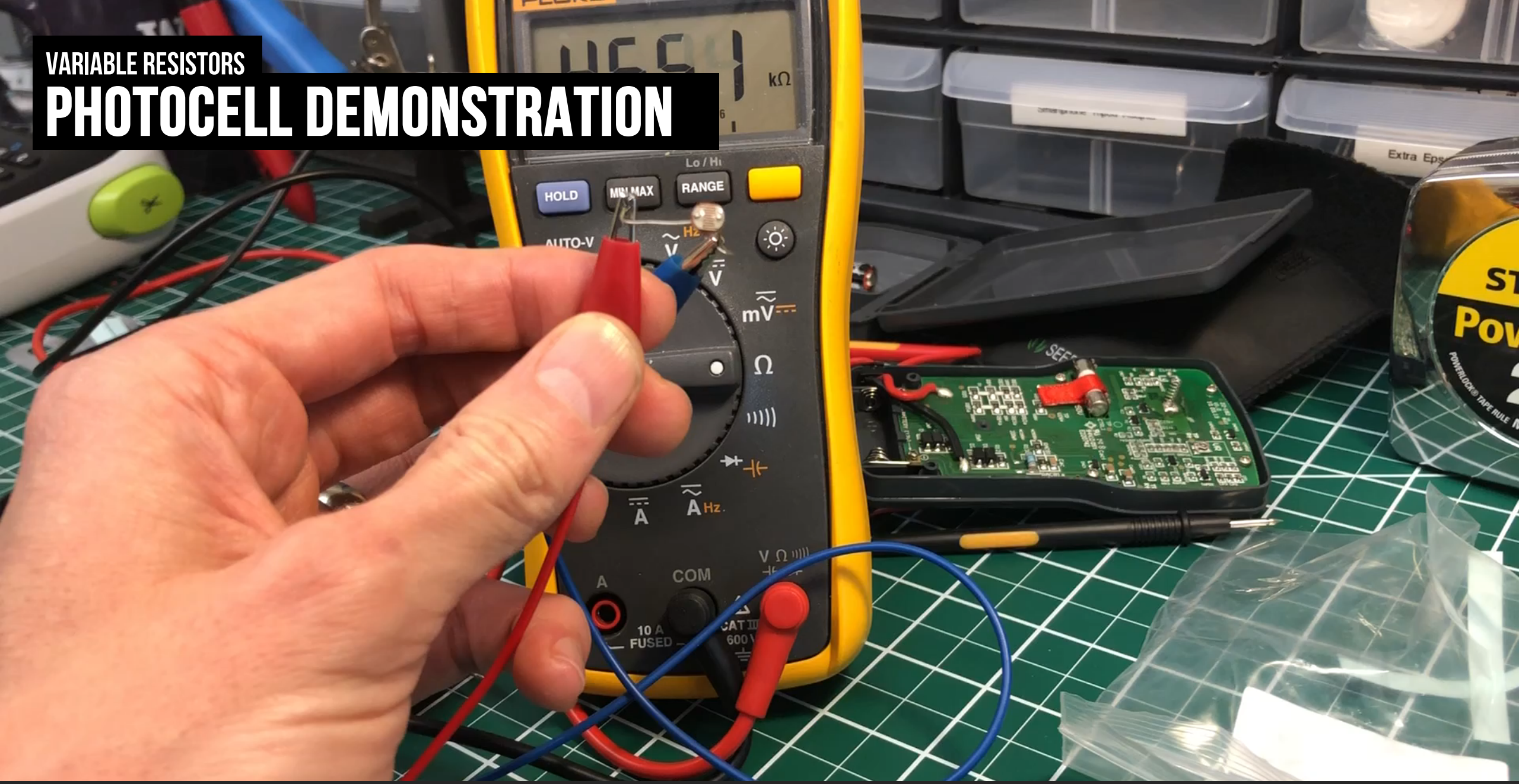
UNIVERSITY of WASHINGTON

ANALOG INPUT

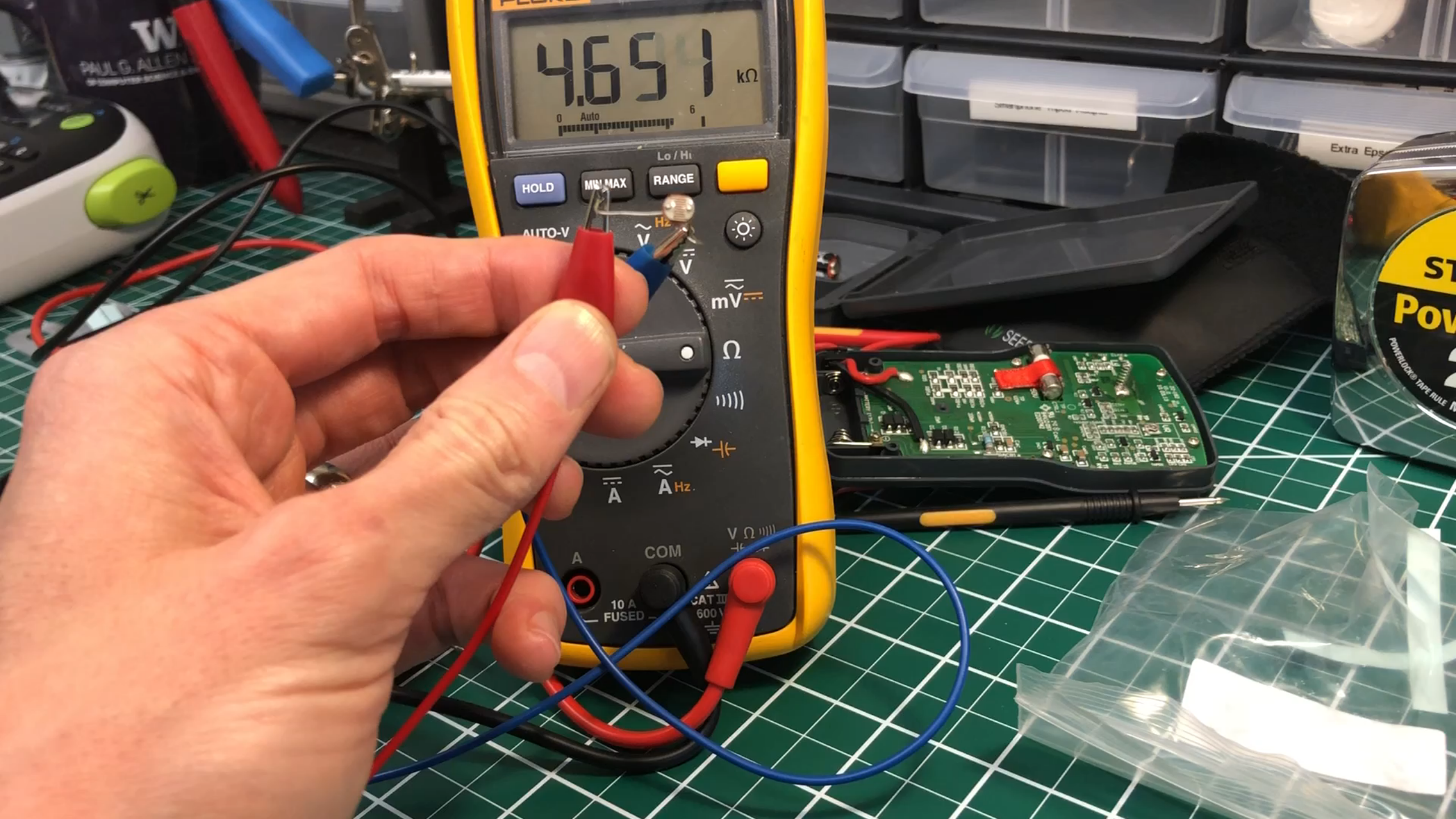
THE PHOTOSENSITIVE RESISTOR



Photosensitive resistors or photocells **change their resistance based** on **light levels**. When light levels go up, resistance goes down.



Video by Jon Froehlich

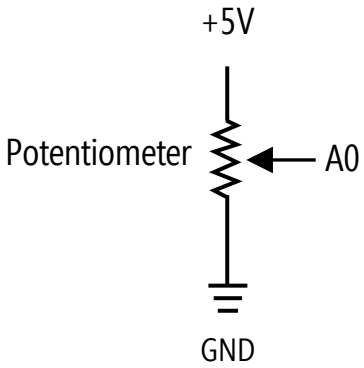

4.651 kΩ
0 Auto 6
Lo / Hi
HOLD
MIN MAX
RANGE
AUTO-V
mV
Ω
A
A Hz.
A
COM
10 A FUSED
V Ω
CAT III
600 V

VARIABLE RESISTORS

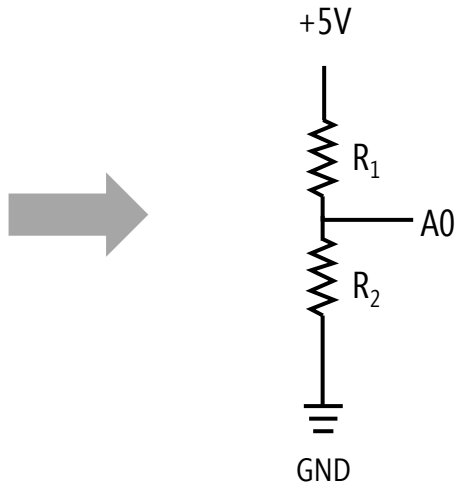
PHOTOCELLS ARE EVERYWHERE!


Melissa & Doug
~crafted by hand~
SOUND PUZZLE
FIRE TRUCK
COUNTY
FIRE DEPT
6
9-1-1

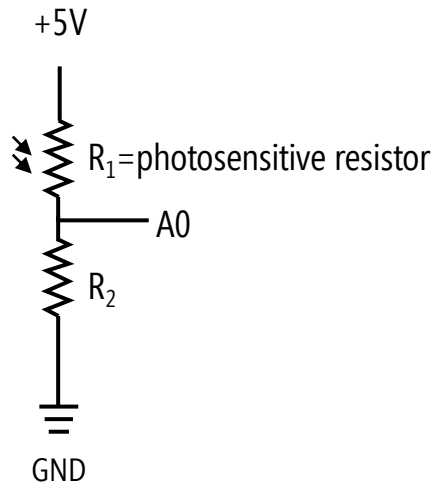
REPLACE POT WITH PHOTOSENSITIVE RESISTOR



Potentiometer
input circuit

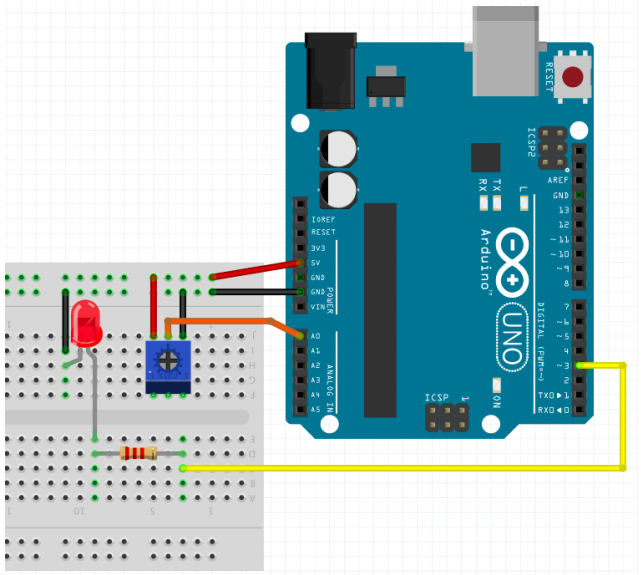


Potentiometer
input circuit is, by
design, a voltage divider

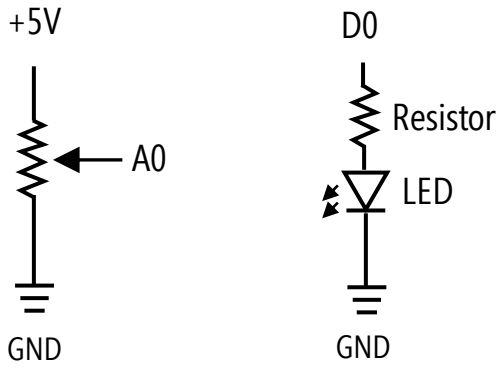


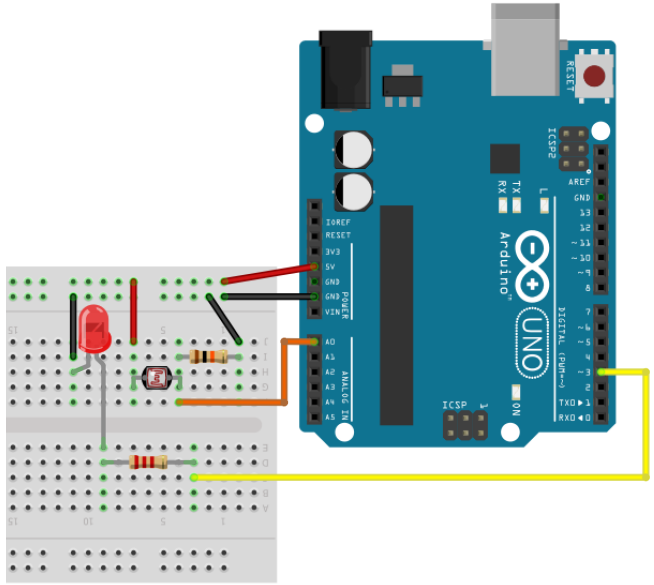
Let's make a similar
circuit but with a
photosensitive resistor

REPLACE POT WITH PHOTOSENSITIVE RESISTOR

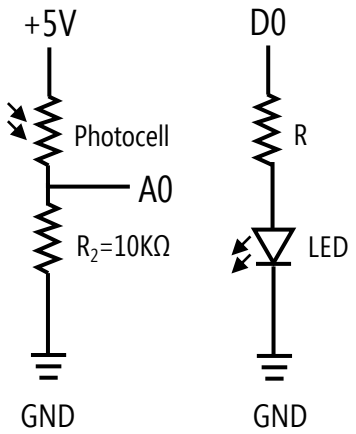


Old Circuit with Potentiometer



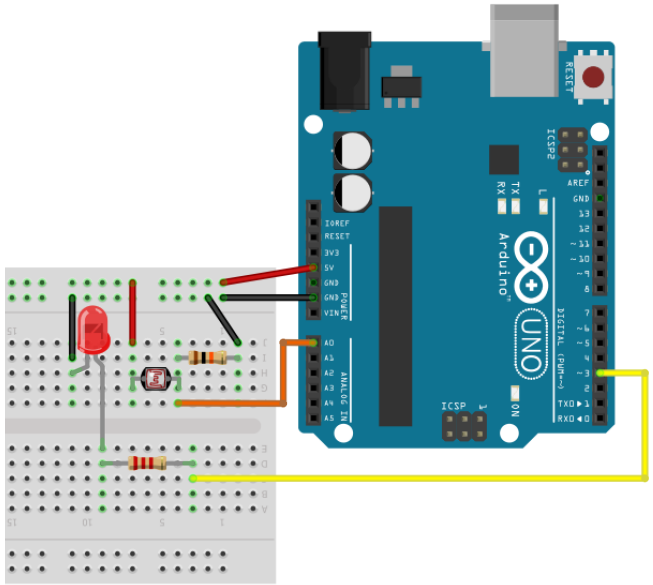


New Circuit with Photocell

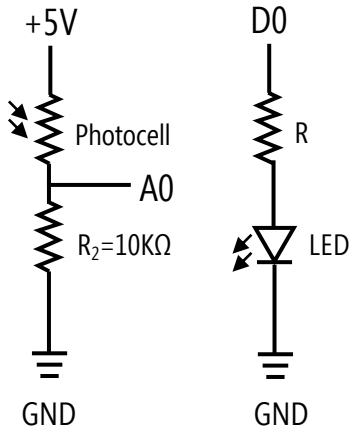


ANALOG INPUT

PHOTOCELL HOOKUP DIAGRAM

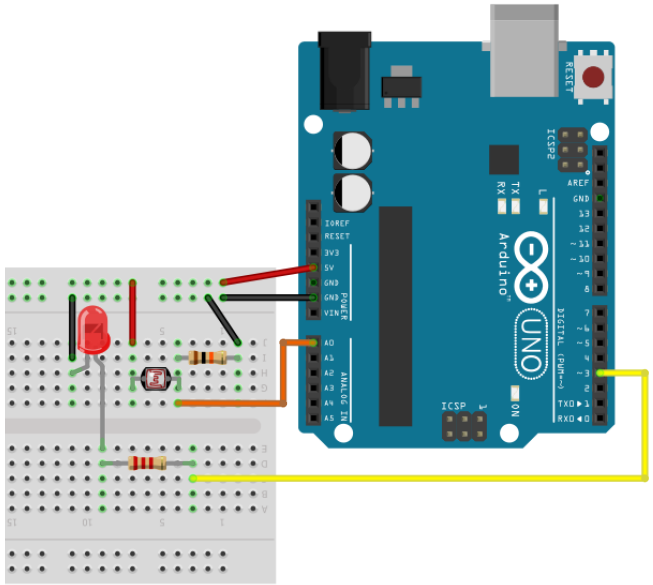


Correct Circuit with Photocell

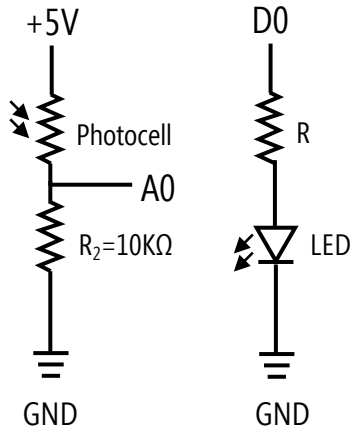


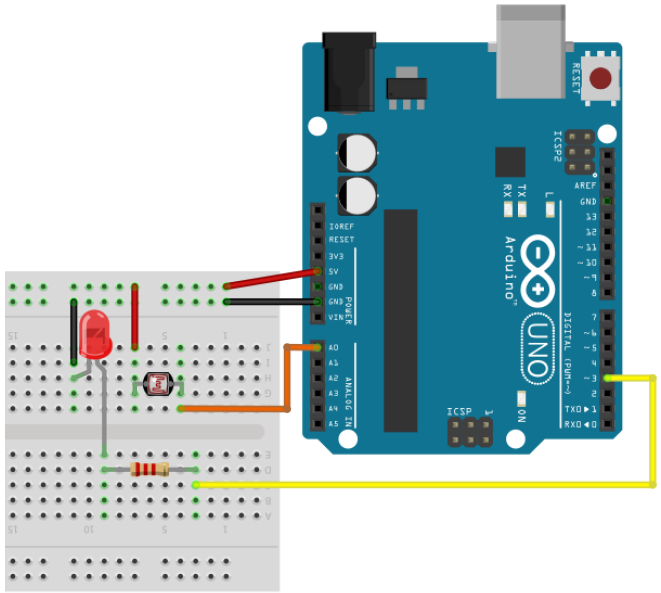
ANALOG INPUT

WHY CAN'T WE DO THIS?

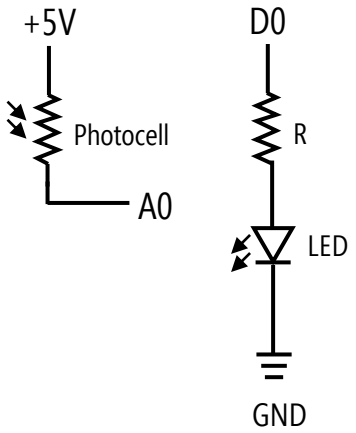


Correct Circuit with Photocell

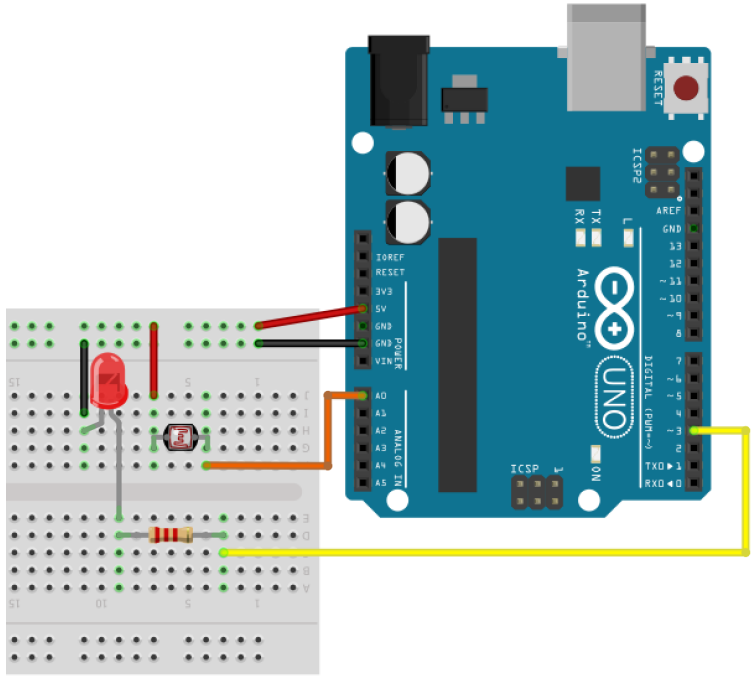




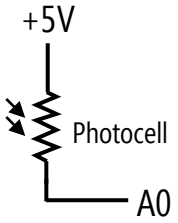
Incorrect Circuit with Photocell



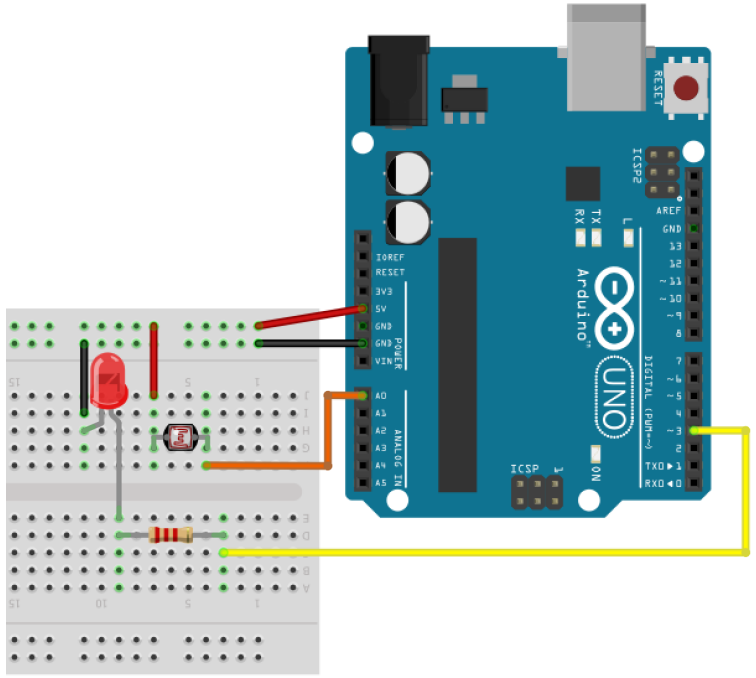
WHY CAN'T WE JUST DO THIS?



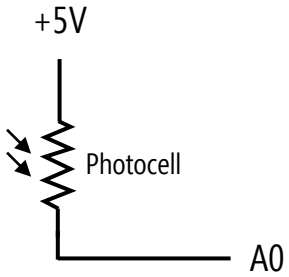
What's wrong with this circuit?



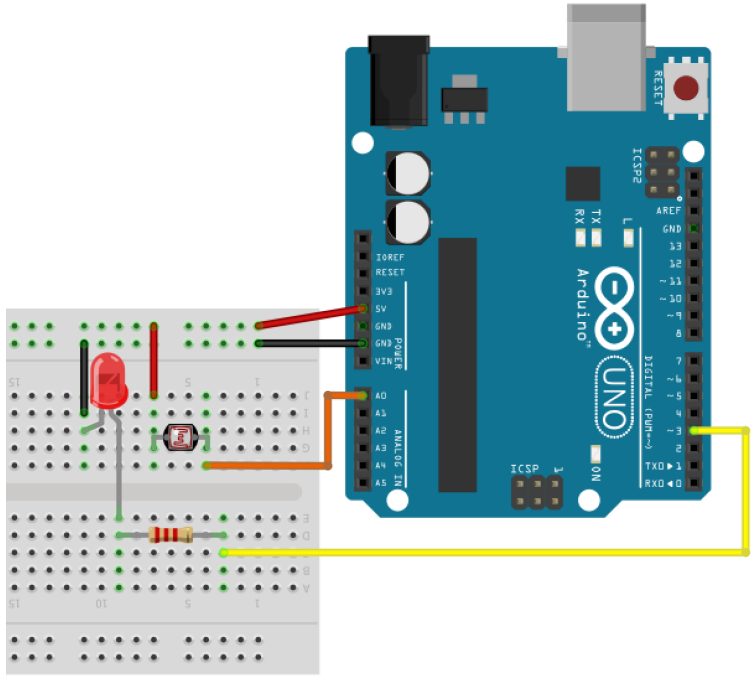
WHY CAN'T WE JUST DO THIS?



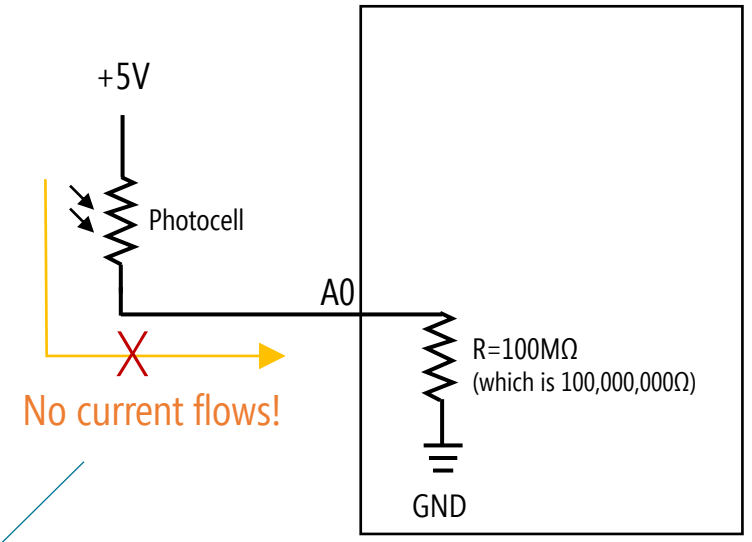
What's wrong with this circuit?



WHY CAN'T WE JUST DO THIS?



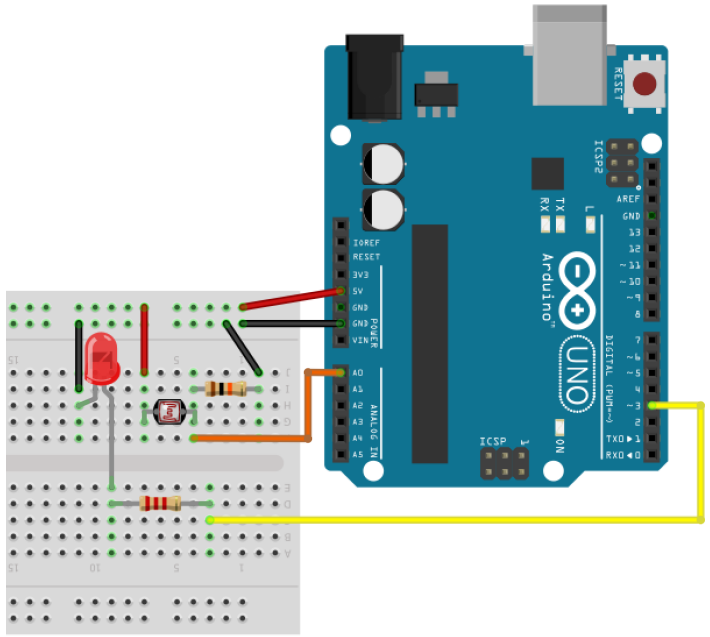
What's wrong with this circuit?



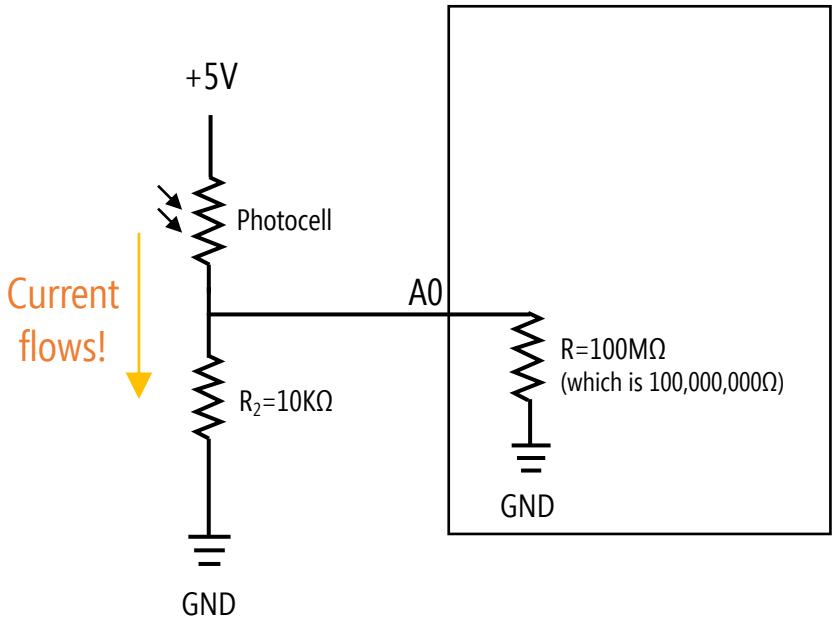
1. No current flows into the input pin because of extremely high resistance in the microcontroller

2. Microcontrollers **read changes in voltage** & not resistance or current

THE CORRECT CIRCUIT DESIGN

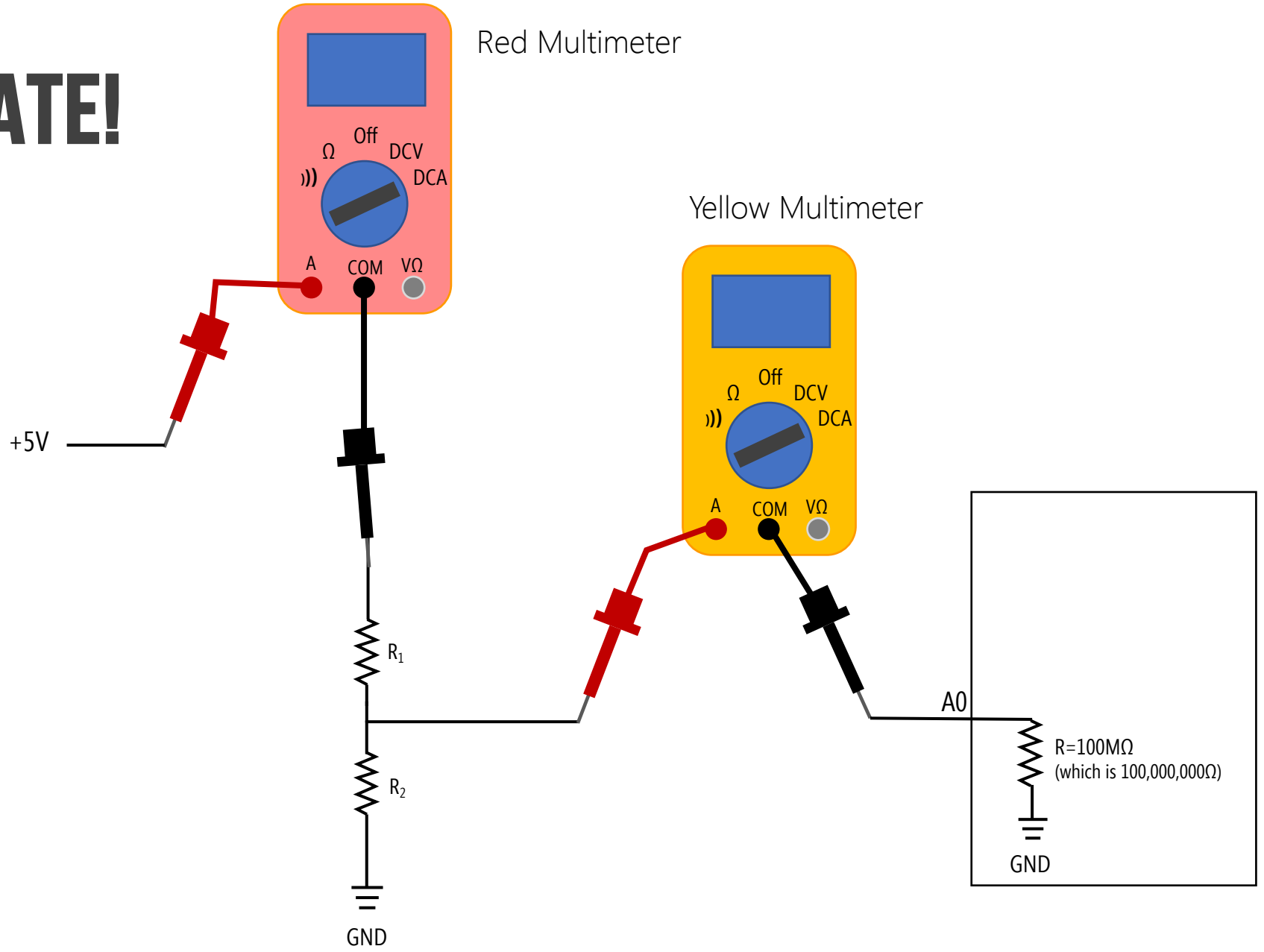


The correct circuit design



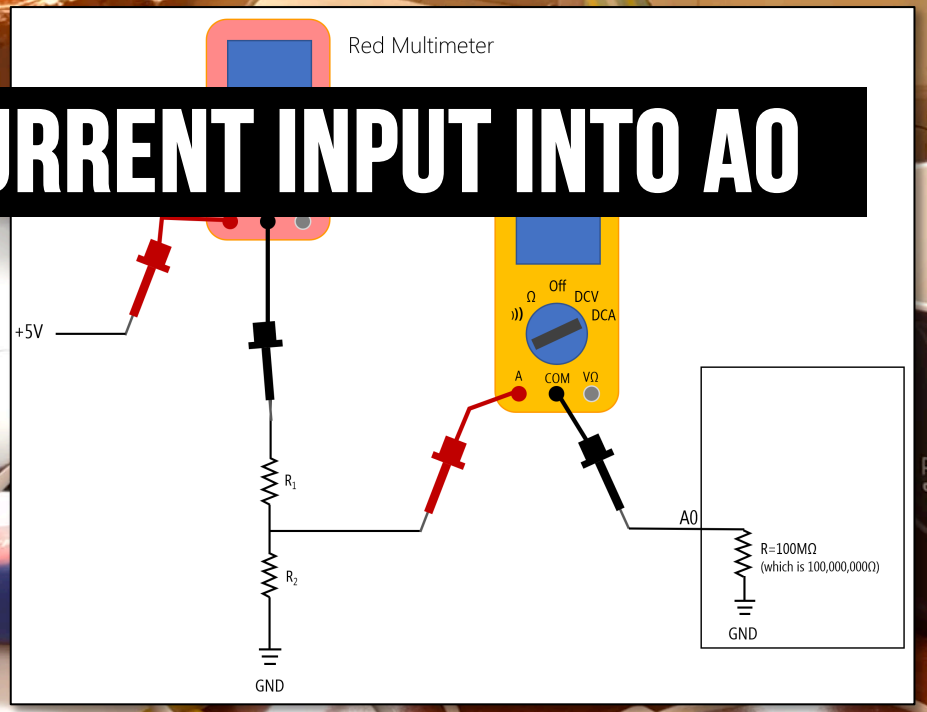
ANALOG INPUT

LET'S INVESTIGATE!

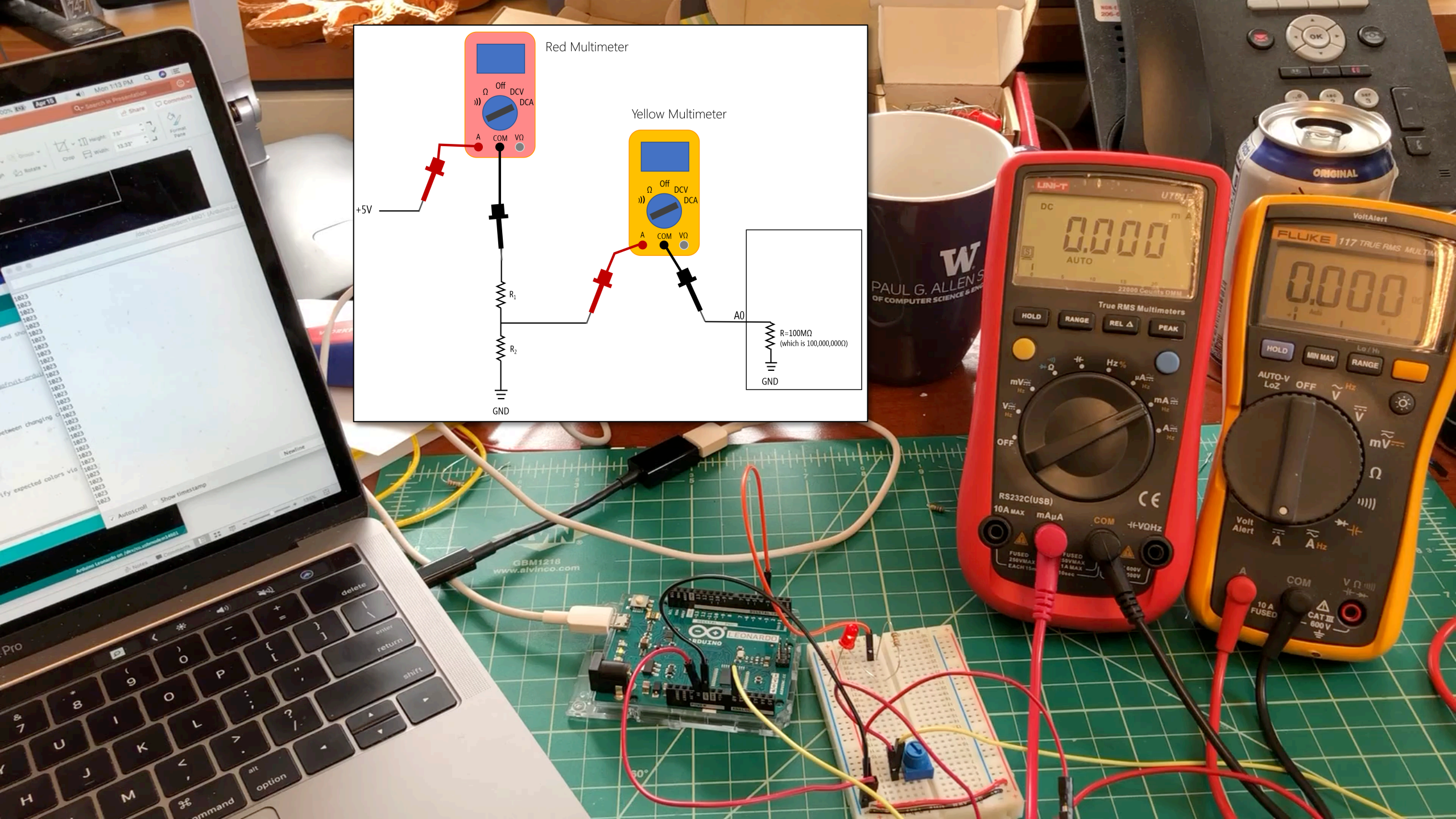


ANALOG INPUT

MEASURING CURRENT INPUT INTO A0

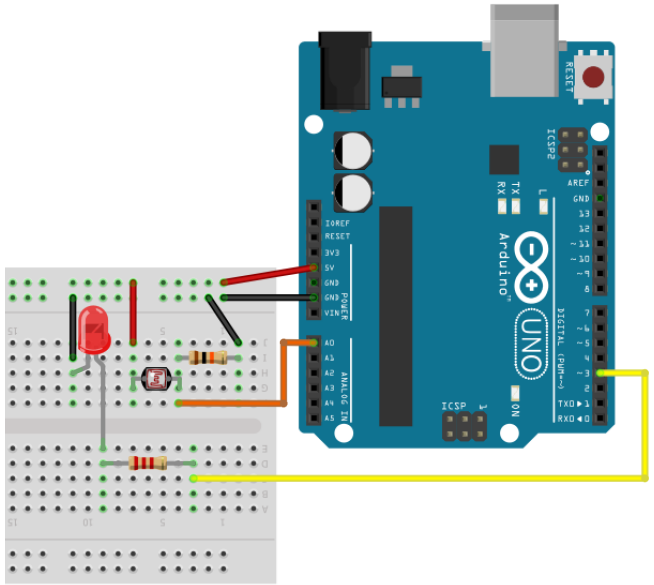


Video by Jon Froehlich

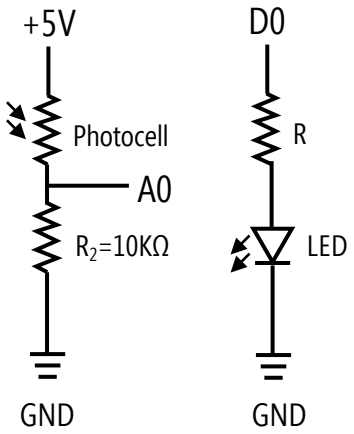

Red Multimeter
Off
Ω
DCV
DCA
A
COM
VΩ
Yellow Multimeter
+5V
R1
R2
GND
A0
R=100MΩ
(which is 100,000,000Ω)
GND

ANALOG INPUT

HOOKING UP PHOTOCELL



Correct Circuit with Photocell

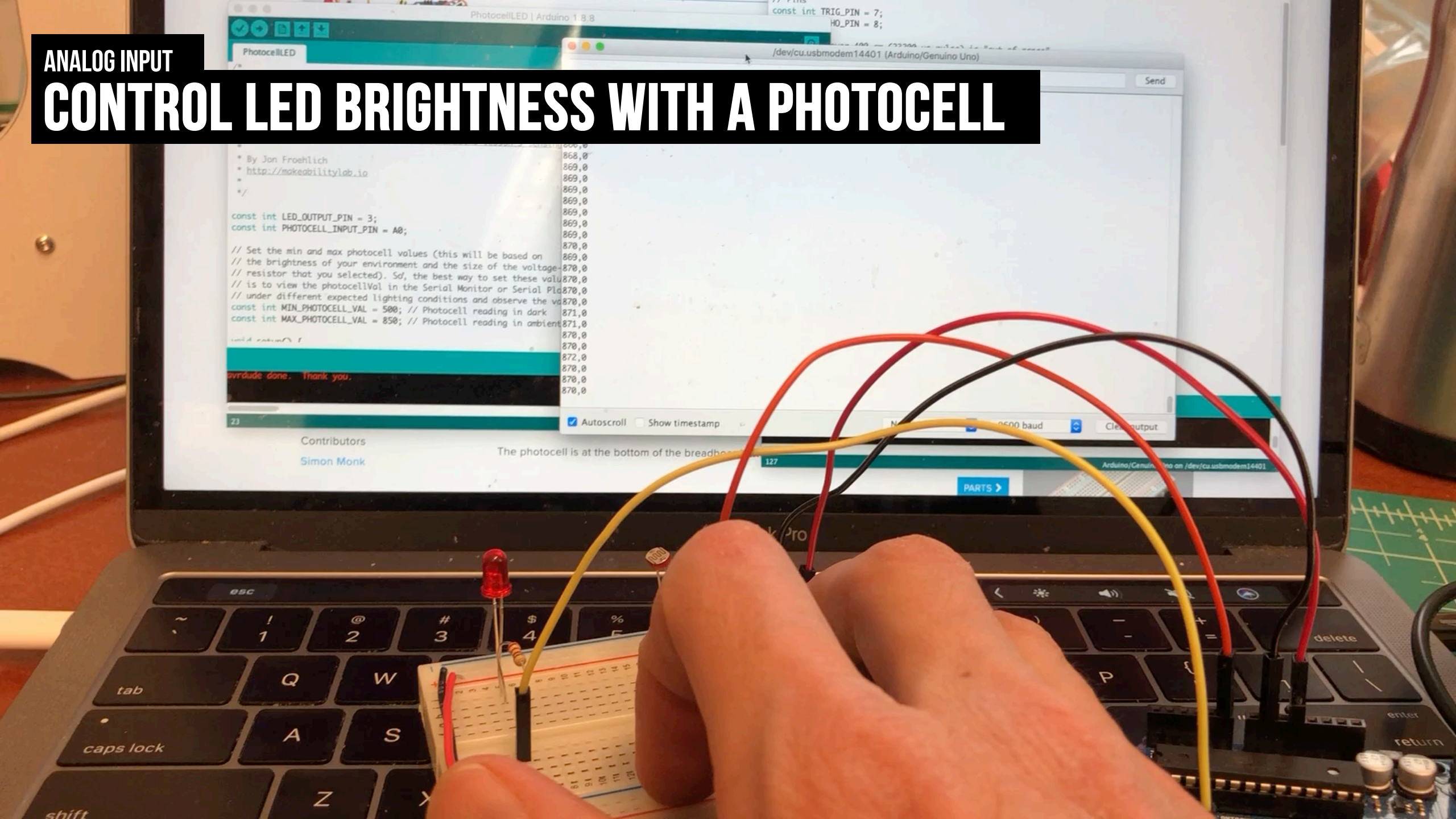


PhotocellLED

```
const int LED_OUTPUT_PIN = 3;
const int PHOTOCELL_INPUT_PIN = A0;

// Set the min and max photocell values (this will be based on
// the brightness of your environment and the size of the voltage-divider
// resistor that you selected). So, the best way to set these values
// is to view the photocellVal in the Serial Monitor or Serial Plotter
// under different expected lighting conditions and observe the values
const int MIN_PHOTOCELL_VAL = 150; // Photocell reading in dark
const int MAX_PHOTOCELL_VAL = 1023; // Photocell reading in ambient light

void setup() {
  pinMode(LED_OUTPUT_PIN, OUTPUT);
  pinMode(PHOTOCELL_INPUT_PIN, INPUT);
  Serial.begin(9600);
}

void loop() {

  // Read the photo-sensitive resistor value. If you have the photocell resistor hooked
  // up as Rtop in the voltage divider (that is, one leg of the photocell is connected to 5V),
  // then higher values correspond to brightness. If you ahve the photocell hooked up as Rbottom
  // in the voltage divider (that is, one leg of the photocell is connected to GND), then
  // higher values correspond to darkness.
  int photocellVal = analogRead(PHOTOCELL_INPUT_PIN);

  // Remap the value for output.
  int ledVal = map(photocellVal, MIN_PHOTOCELL_VAL, MAX_PHOTOCELL_VAL, 0, 255);

  // The map function does not constrain output outside of the provided range
  // so, we need to make sure that things are within range for the led
  ledVal = constrain(ledVal, 0, 255);

  // We want to invert the LED (it should be brighter when environment is darker)
  // This assumes the photocell is Rtop in the voltage divider
  ledVal = 255 - ledVal;

  // Print the raw photocell value and the converted led value (e,g., for Serial Plotter)
  Serial.print(photocellVal);
  Serial.print(",");
  Serial.println(ledVal);

  // Write out the LED value.
  analogWrite(LED_OUTPUT_PIN, ledVal);

  delay(100);
}
```

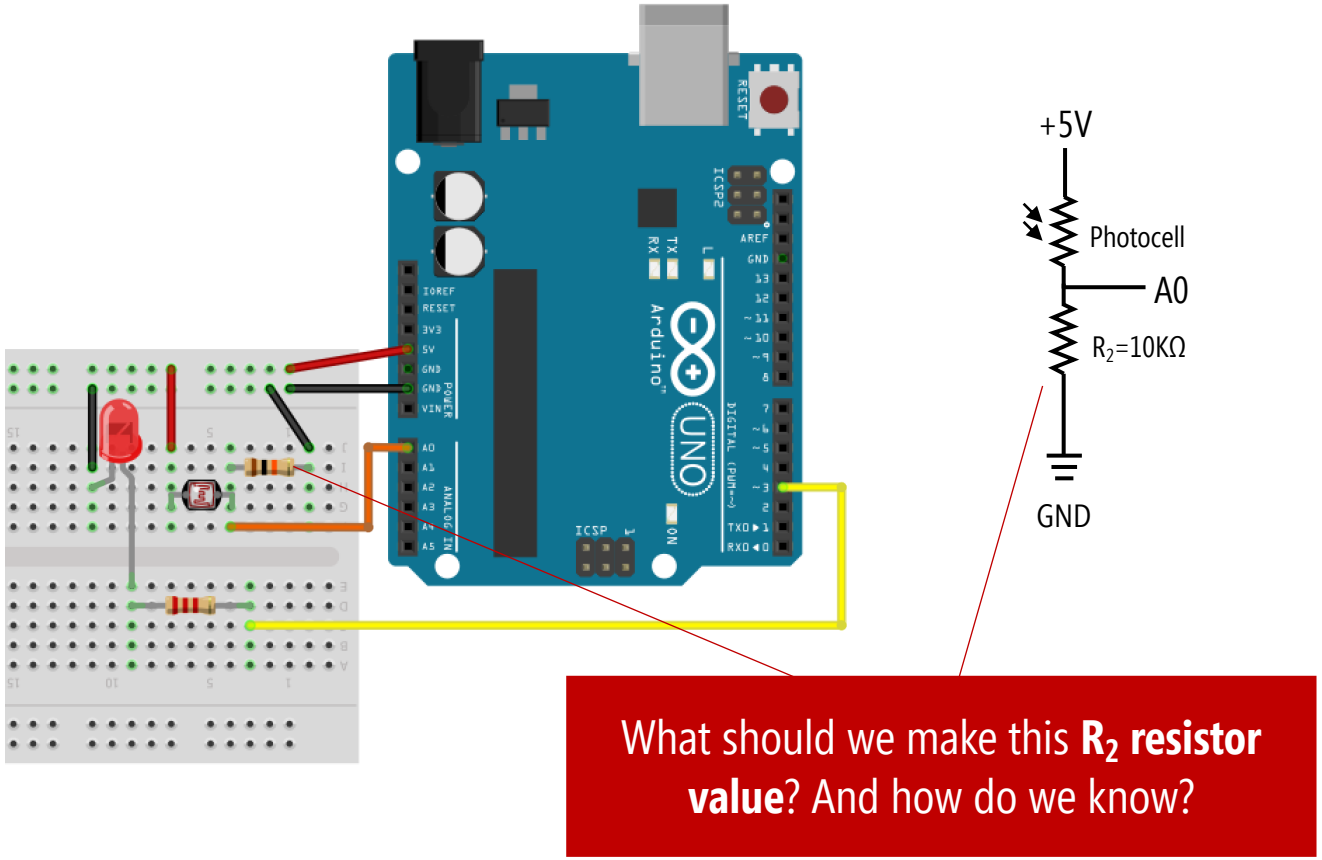
ANALOG INPUT

CONTROL LED BRIGHTNESS WITH A PHOTOCELL



$$V_{out} = Vin * \frac{R_2}{R_1 + R_2}$$

HOW DO I KNOW WHAT TO SELECT FOR R2?



What should we make this **R_2 resistor value**? And how do we know?

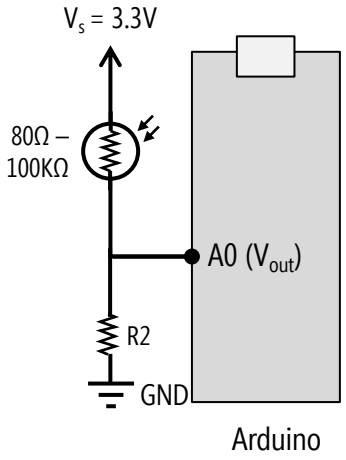
It depends on the **resistance range** of your variable resistor (R1)

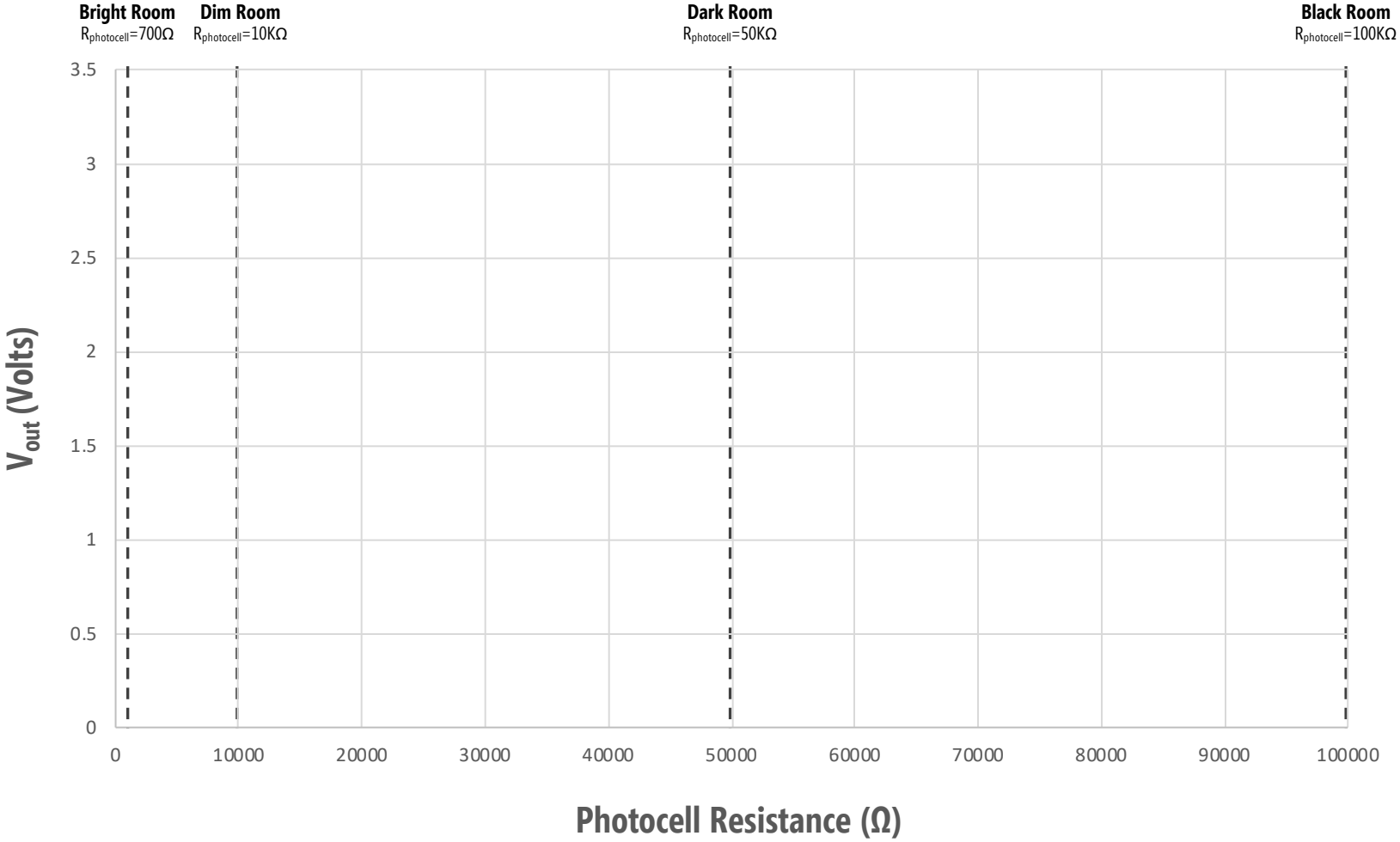
It depends on the **context of use** (what's the expected resistance range of R1)

USING A VARIABLE RESISTIVE SENSOR

GRAPHING VOUT FOR PHOTOCELL

$$V_{out} = Vin * \frac{R_2}{R_1 + R_2}$$

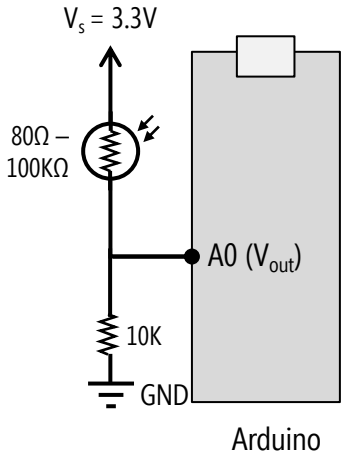


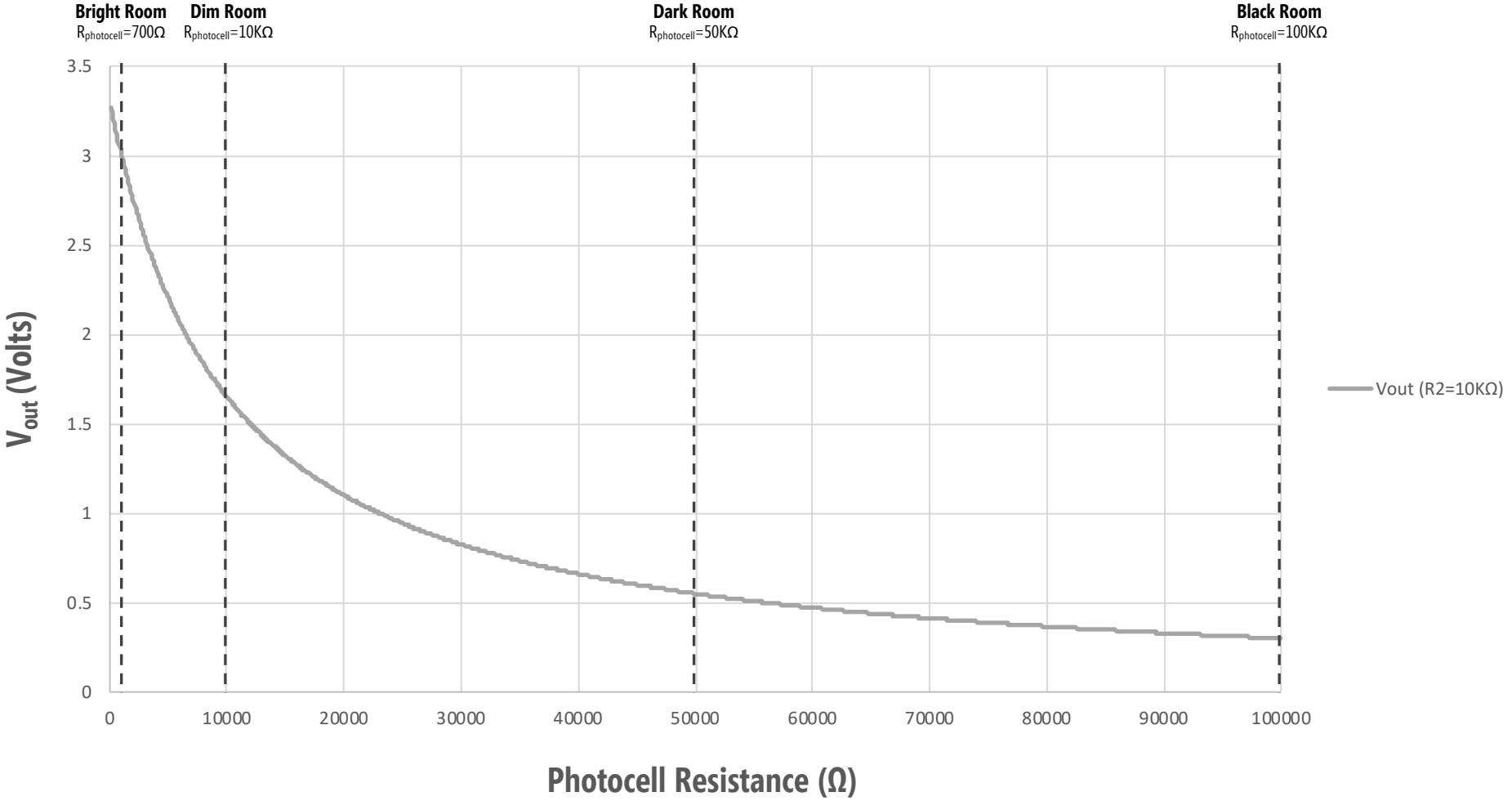


USING A VARIABLE RESISTIVE SENSOR

GRAPHING VOUT FOR PHOTOCELL

$$V_{out} = Vin * \frac{R_2}{R_1 + R_2}$$

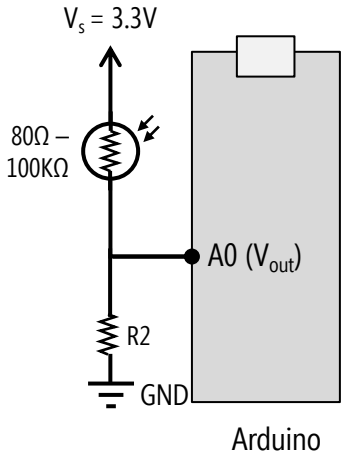


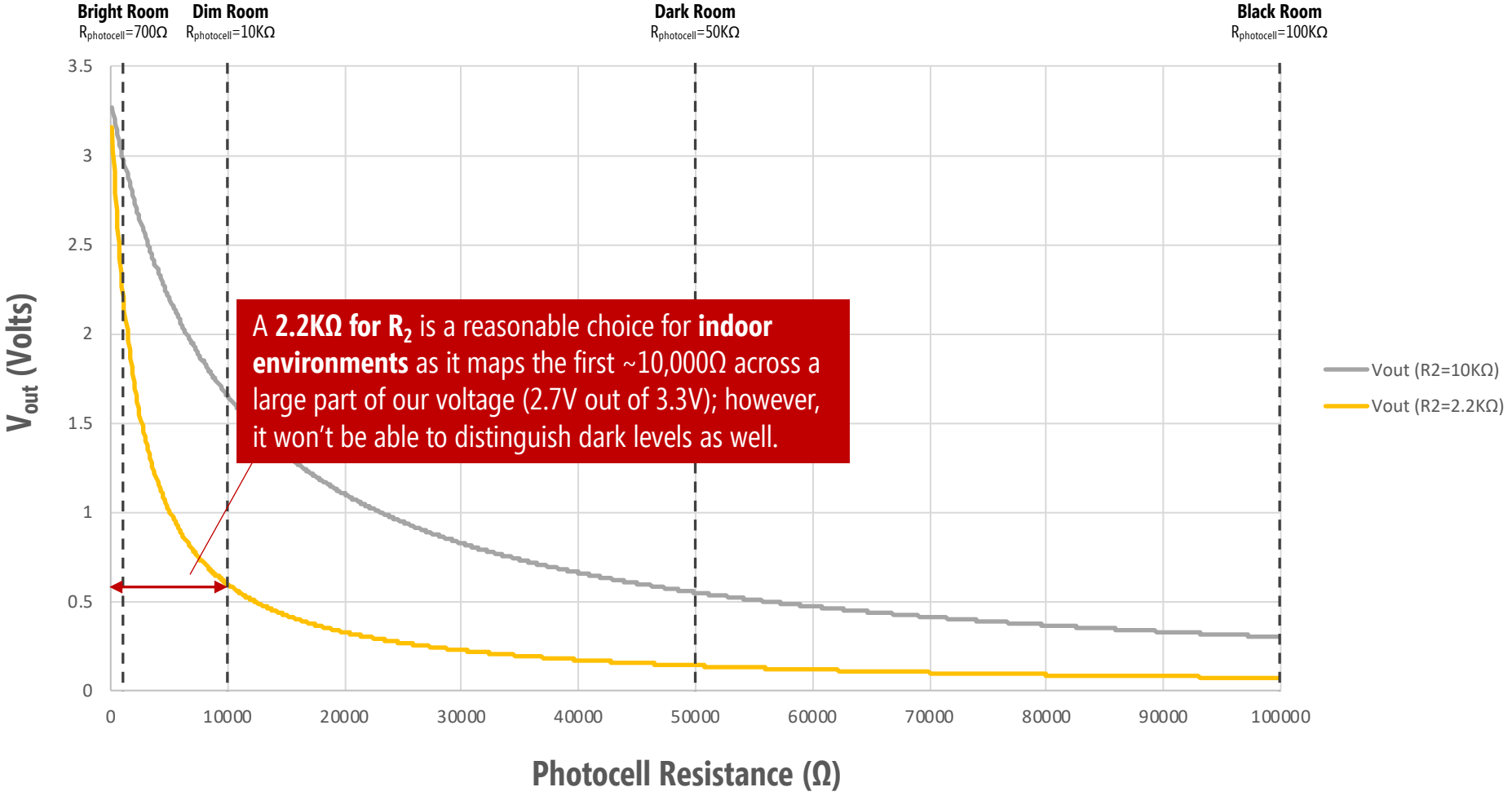


USING A VARIABLE RESISTIVE SENSOR

GRAPHING VOUT FOR PHOTOCELL

$$V_{out} = Vin * \frac{R_2}{R_1 + R_2}$$

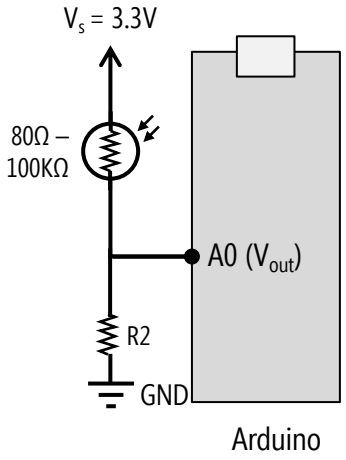


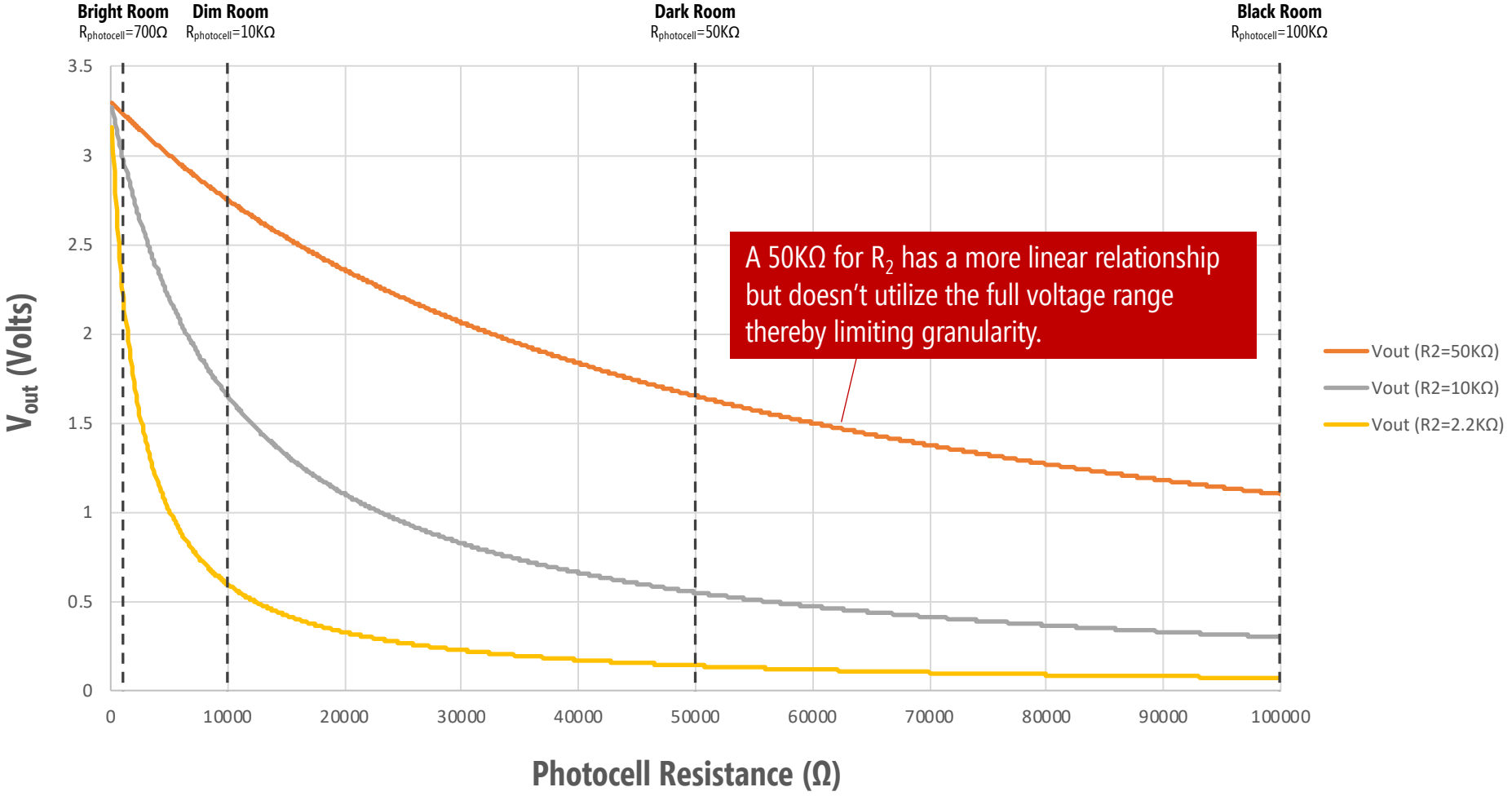


USING A VARIABLE RESISTIVE SENSOR

GRAPHING VOUT FOR PHOTOCELL

$$V_{out} = Vin * \frac{R_2}{R_1 + R_2}$$

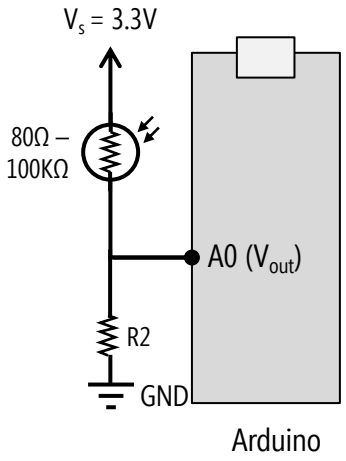


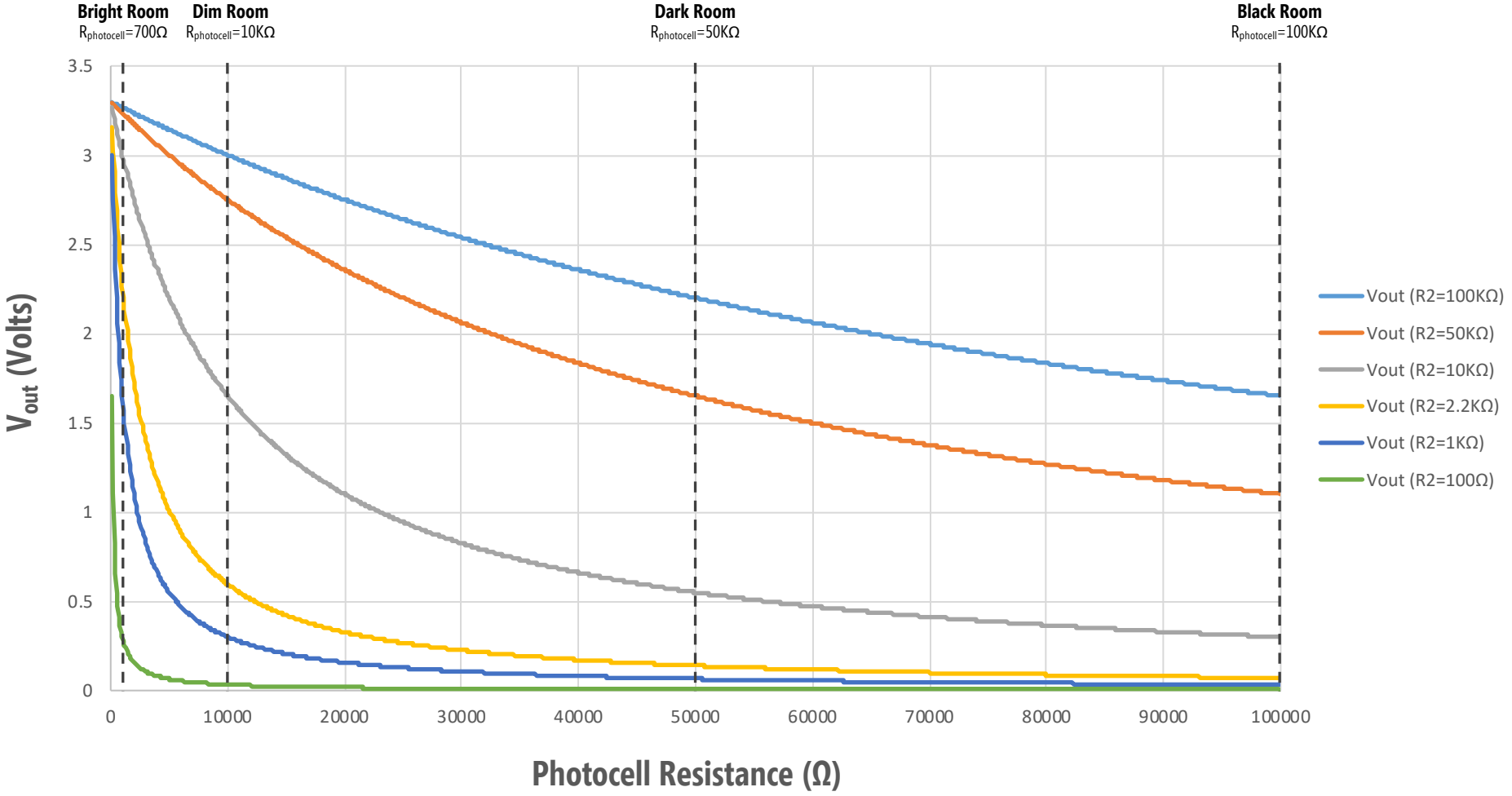


USING A VARIABLE RESISTIVE SENSOR

GRAPHING VOUT FOR PHOTOCELL

$$V_{out} = Vin * \frac{R_2}{R_1 + R_2}$$

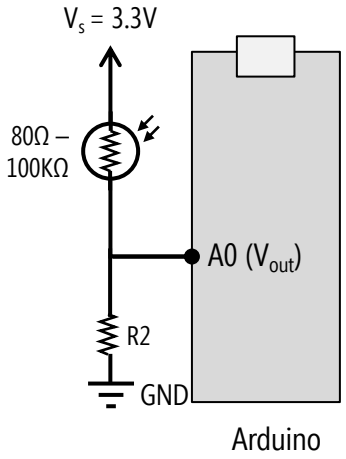




USING A VARIABLE RESISTIVE SENSOR

AXEL BENZ FORMULATION FOR R2

$$V_{out} = Vin * \frac{R_2}{R_1 + R_2}$$



First, measure the R_{min} and R_{max} values for your specific environment using a multimeter

Then use these measurements to find the value for R_2

$$R_2 = \sqrt{R_{1_{min}} * R_{1_{max}}}$$

$$R_2 = \sqrt{80\Omega * 10{,}000\Omega}$$

$$R_2 = \sim 900\Omega$$

EXPERIMENTATION

You'll also experiment with setting thresholds in the mapping function based on empirical observation

RedBearDuoReadPhotocellFadeLED

```
SYSTEM_MODE(MANUAL);

const int LED_OUTPUT_PIN = D0;
const int PHOTOCELL_INPUT_PIN = A0;

// Set the min and max photocell values (this will be based on
// the brightness of your environment and the size of the voltage-divider
// resistor that you selected).
const int MIN_PHOTOCELL_VAL = 1200; // Photocell reading in dark
const int MAX_PHOTOCELL_VAL = 3700; // Photocell reading in ambient light (tested in my office)

void setup() {
  pinMode(LED_OUTPUT_PIN, OUTPUT);
  pinMode(PHOTOCELL_INPUT_PIN, INPUT);
  Serial.begin(9600);
}

void loop() {

  // Read the photo-sensitive resistor value
  int photocellVal = analogRead(PHOTOCELL_INPUT_PIN);

  // Remap the value for output.
  int ledVal = map(photocellVal, MIN_PHOTOCELL_VAL, MAX_PHOTOCELL_VAL, 0, 255);

  // The map function does not constrain output outside of the provided range
  // so, we need to make sure that things are within range for the led
  ledVal = constrain(ledVal, 0, 255);

  // We want to invert the LED (it should be brighter when environment is darker)
  ledVal = 255 - ledVal;

  // Print the raw photocell value and the converted led value (e,g., for Serial Plotter)
  Serial.print(photocellVal);
  Serial.print(",");
  Serial.println(ledVal);
```

Use Serial Plotter to figure out good min and max values for map