

IR_Calibration -- Overview



OBJECTIVES

After performing this lab exercise, learner will be able to:

- Program Arduino board to get the distance between two points
- Sense dark and light objects
- Differentiate between different colors

EQUIPMENT

To carry out this experiment, you will need:

- TBS1KB - Digital Oscilloscope from Tektronix
- Arduino Duemilanve or Uno board
- Voltage probe (provided with oscilloscope) / BNC cables
- Breadboard and connecting wires

THEORY

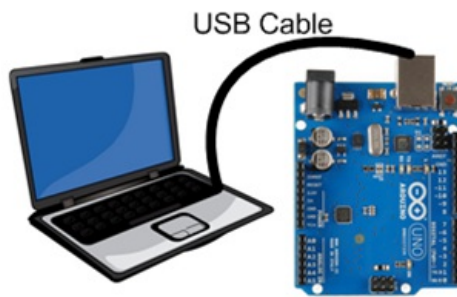
- Infrared rays emitted from the IR LED, when strikes an object reflects back some part of the radiation depending upon the properties of the object.
- If the object is light, the reflected radiations will be more when compared to a darker object.
- Different colors behave differently with IR radiations on them.

IR_Calibration -- Procedures

Step 1

DUT / SOURCE SETUP

- Ensure you have Arduino IDE (software to program the Arduino boards) installed on your computer.
- Connect the Arduino board to PC using USB cable.



- Program it with relevant code .
- Take the output from mentioned probing point(s) .

Step 2

MEASUREMENT / SCOPE SETUP

- Power ON the oscilloscope .
- Connect the Channel 1 probe of the oscilloscope to **Vout-1** .
- Acquire the signal(s) from **circuit** on oscilloscope .

Step 3

- Do the Autoset on the scope to efficiently capture and view the signal .
- If AUTOSET feature is not enabled, then manually set the horizontal and vertical scale, and trigger condition to view 3-4 cycles of waveform without any clipping.

Step 4

- From the measurement menu, configure **MAXIMUM** and **MEAN** measurement on acquired channel.

Step 5

- Take the measurement (**MAXIMUM** and **MEAN**) of the signal and distance between the IR sensor from the object.

Step 6

- Now vary the distance between the object and IR sensor, measure signal amplitude and the distance.
- Take measurements for different set of distances.

Step 7

- Plot a graph of amplitude (in volts) Vs distance between sensor and object. Verify that amplitude decreases sharply with increase in distance.
- The plot would look somewhat like:

