

# Frequency Meter

## Guided Project Instructions

**nPoints**  
BY NTS PRESS



**Related Core Concepts:**  
**Lights Out!**  
**Lumens!**

**“During this project, you will explore how both hardware timing and software timing are use to control the generation of a waveform”**

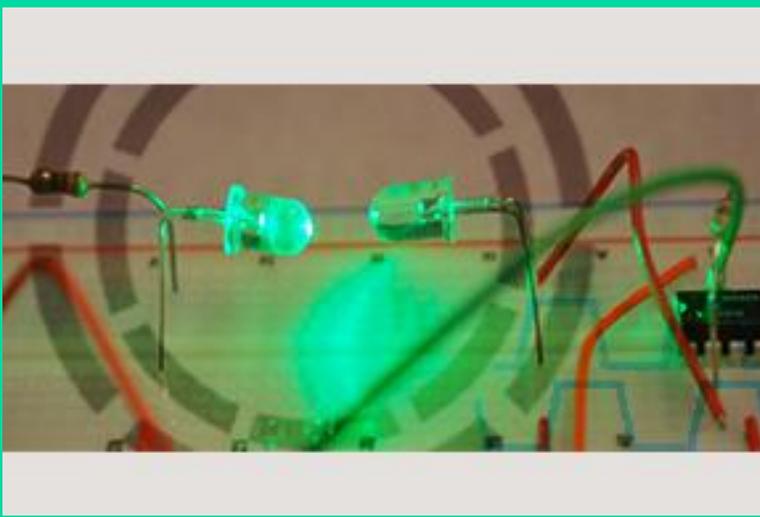


Figure 1 Example Light Frequency Meter

## Learn It!

Detecting the frequency content of any signal is extremely useful in applications involving vibrations or impact. Frequency detection is also useful as a type of feedback in motor speed systems. The pulse counting experimental technique employed in this project is widely used in diverse applications including time keeping, voltage digitizing, and rotational characterization. The experimental set-up consists of two parts. First, a driver circuit controlled by a data acquisition (DAQ) device blinks an LED at a

constant frequency. Then, a second sensing circuit, which includes its own LED used as a photo sensor, detects the blinking light and converts this optical signal into a digital pulse train. The digital pulse train is input to the DAQ device's counter input, where counting the number of digital pulses for a given time interval yields the blinking frequency. During this project, you will explore how both hardware timing and software timing are use to control the generation of a waveform (in this case, the digital pulse train) by a DAQ device.

## Build It!

This experiment consists of a light source and detector circuit

**Task 1:** Since the current-producing capabilities of a DAQ device's output channel is limited (e.g., the Digital Output channel of a myDAQ can only source up to 4 mA), instead of simply powering the LED directly from a DAQ device output channel, construct the transistor-based circuit shown in Figure 1 to toggle the LED. This will be your light transmission circuit.

**Task 2:** Construct an amplifier circuit using an LM741 and a resistance “R” that will give you a gain of about 1000 to be used as a light detector. You do not need to provide a resistance in the inverting or non inverting leg we can assume it will be close to one ohm which will make it so that the gain for the inverting configuration will directly be related to the resistance in the feedback leg.

Note: Make sure to test your detector circuit before connecting it to your DAQ device to confirm you are not causing the output to go beyond 3.3V since this may damage inputs on your DAQ. Also the transmission circuit should have an LED with a wide energy band gap such as green one while the detection circuit should use a smaller energy band gap LED like red.

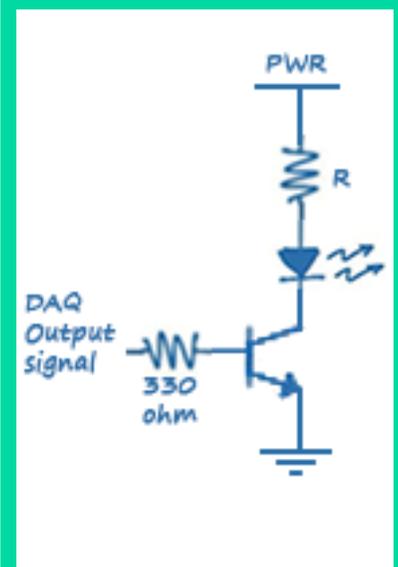


Figure 2 Transistor driven LED circuit

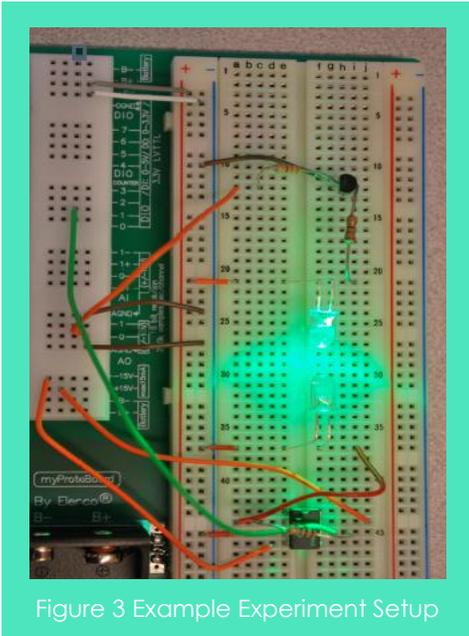


Figure 3 Example Experiment Setup

**Task 3:** Create code that will generate a pulse train and output it using hardware timing on your DAQ device's AO line. Also have it read from a counter input and calculate frequency based on that reading. Wire your AO line to the "DAQ Output Signal" line from the circuit built from Figure 1. Wire the OpAmp output to your DAQ device's counter input.

### Guiding Questions:

- What is gain bandwidth product and what is it for your OpAmp?
- Why was an LED with a large energy band gap chosen for the transmitter? What would happen if two LEDs of similar band gap sizes were used instead?
- What is the limiting factor in the frequency that can be produced and read in this setup?

Note: The pulse train can be made using the "Simulate Signal" Express VI found on the Express>>Input palette. You can change the output and offset to make it fit the 0-5V requirement. The DAQ Assistant VI can be used for the Analog output and Counter input. Be sure to set your timing to either N Samples or Continuous.

**Task 4:** Run your setup, vary the frequency control and note the output frequency value. Calculate the error between the frequency you set on your control and the frequency you read back.

### Software Implementation

**Task 4:** Connect the input of your light transmitter circuit to a Digital Input/Output (DIO) channel on your DAQ device. Then, write a LabVIEW program that configures this channel as a Digital Output (DO) and toggles this DO between its LO and HI digital level at frequency  $f$ . Because no hardware timing is available to control the DO channel, you will have to toggle between these states under the control of your LabVIEW program.

**Task 5:** Repeat task 4 with the new software implementation and take note of where in the frequency range you begin to see an error deviation.

### Expand it!

- The pulses read from detecting light are the basis for optical encoder designs. Create an optical encoder by cutting some holes in a disk of paper and spin this between the LEDs. Keep the transmitting LED constantly on for this portion. Try and see if you can calculate RPM from spinning the disk between the LEDs. Also try to find ways to optimize this design.

## Research It!

### Pulse Wave

[http://en.wikipedia.org/wiki/Pulse\\_wave](http://en.wikipedia.org/wiki/Pulse_wave)

### Optical Encoders and Accessories

<http://www.encoder.com/literature/optical-encoder-guide.pdf>

### Rotary Encoder

<https://www.youtube.com/watch?v=CpwGXZx-5Ug>