

Controlling Vibrations

The screen of a TV or computer monitor has many thousands of pixels and millions of possible colors and these need to be updated many times per second—far faster than any human could do the calculations or make the adjustments. The technology relies on its electronics to carry out the calculations, but the people who design and maintain that technology still need to understand the math behind it. Demanding users—people who strive to achieve the best possible results from the equipment they buy—also benefit from understanding how color can be programmed mathematically.



“Oscillations of Units”
© Agsandrew | Dreamstime.com

This activity has a second purpose as well, since the mathematics used in this activity applies generally to all “harmonic oscillations.”

Light itself consists of electrical and magnetic fields in harmonic oscillation, but other examples include the vibrations caused by earthquakes, the musical notes produced by a violin, and the resonances that can destroy a bridge. Even if you never again program colors quite this way, the exercise will help you understand and control the mathematics which describes all oscillations.

The general equation to describe simple harmonic motion can be written as shown below. The equation is written in radians, which is the form required by LabVIEW’s function interpreter, by Excel and by most other computer applications.

$$Y = C + A \sin(2\pi t / T + \phi)$$

“C” is the central point for the oscillation. The center point is often zero so oscillations are alternately positive and negative. Particularly in electronic systems, however, the center point must often be set to a value above zero to keep the oscillation from becoming negative. Color output can never be negative (unless the emitter becomes a “light sucker”), so the functions in this activity will always have a center point of +50.

“A” is the “amplitude.” It is a positive number which specifies how far the oscillation can vary from the center point. “Since the maximum brightness available with our system is 100 and the minimum is 0, the largest possible amplitude with our system is 50.

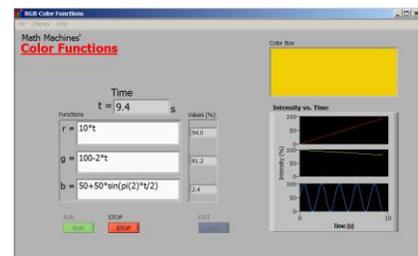
“T” is the “period.” It specifies the time for one complete oscillation.

“φ” (Greek letter “phi”) is the “phase angle.” It specifies the point in the cycle at which the vibration begins. If φ = 0, the oscillation starts at its center point and is increasing. If φ = π/2, the oscillation has a quarter-cycle “head start,” and has already reached its maximum when t = 0. “φ” is most important when you need to consider how two or more different oscillations work together.

“t” is the independent variable, elapsed time, measured in seconds.

“Y” is the quantity that is oscillating. In this activity Y specifies the instantaneous brightness of an individual color within our pixel.

Task: Your task in this activity is to create mathematical functions which cause the color of the RGB Color Mixer to oscillate in controlled ways.



Additional Materials: RGB Color Mixer

Math Machines Program:

Color Functions

Activity File: Sine001



1. Load the activity file Sine001. In this activity file, one example of the period function above has been entered to control the intensity of red, green and blue pixels.

General form: $Y = C + A \sin(2\pi t / T + \phi)$

This example: Intensity = $50 + 50\sin(2\pi t/10)$

Using LabVIEW syntax: r, g, b all = $50 + 50 * \sin(\pi(2)*t/10)$

- a. Run the program as loaded and describe in words how the light from the color box changes.

- b. Identify the value of each constants in these functions:

i. Center point, $C =$ _____

ii. Amplitude, $A =$ _____

iii. Period, $T =$ _____

iv. Phase angle, $\phi =$ _____

- c. Identify the variables in these functions:

i. Independent variable(s) _____

ii. Dependent variable(s) _____

2. Modify one or more of the functions for r, g and b to make the RGB Color Mixer oscillate in yellow light, beginning at a medium (50%) level, increasing to maximum brightness after 2.5 seconds, emitting no light when $t = 7.5$ seconds, and returning to medium when $t = 10$ seconds. It should also be at maximum brightness when $t = 12.5$ s, 22.5 s, 32.5 s, etc. Recall that “yellow” is a combination of equal parts red and green, with no blue. Record the functions you use below.

Red, r = _____

Green, g = _____

Blue, b = _____

3. Modify as many of the function as necessary to make the RGB Color Mixer oscillate with the same timing as in question 2, but showing only purple light. ("Purple" is a combination of equal parts red and blue.) Record the functions you use below.

Red, $r =$ _____
 Green, $g =$ _____
 Blue, $b =$ _____

4. Modify as many of the function as necessary to make the RGB Color Mixer oscillate twice as fast as in the previous question while showing only white light.

Red, $r =$ _____
 Green, $g =$ _____
 Blue, $b =$ _____

5. Use phase angles to modify as many of the function as necessary to make the RGB Color Mixer oscillate so it begins with a maximum intensity of white light and oscillates with the same timing as in question 4. (Note that multiples of π must always be entered in the form "pi(n)" even if $n=1$. "n" can be any positive or negative real number.)

Red, $r =$ _____
 Green, $g =$ _____
 Blue, $b =$ _____

6. Modify as many of the function as necessary to make the RGB Color Mixer oscillate so it begins with a maximum intensity of white light, decreases to zero after 25 seconds and returns to maximum intensity once every 50 seconds.

Red, $r =$ _____
 Green, $g =$ _____
 Blue, $b =$ _____

7. Modify as many of the function as necessary to make the RGB Color Mixer oscillate so it (a) begins with a maximum intensity of white light, (b) becomes pure blue after 25 seconds, and (c) returns to maximum white intensity after 50 seconds.

Red, $r =$ _____
 Green, $g =$ _____
 Blue, $b =$ _____