

Phase-sensitive detection in the undergraduate lab using a low-cost microcontroller

K. D. Schultz

Citation: *American Journal of Physics* **84**, 557 (2016); doi: 10.1119/1.4953341

View online: <http://dx.doi.org/10.1119/1.4953341>

View Table of Contents: <http://scitation.aip.org/content/aapt/journal/ajp/84/7?ver=pdfcov>

Published by the [American Association of Physics Teachers](#)

Articles you may be interested in

[A low-cost mirror mount control system for optics setups](#)

Am. J. Phys. **83**, 186 (2015); 10.1119/1.4895343

[Low-cost student experiments in optics](#)

Phys. Teach. **52**, 442 (2014); 10.1119/1.4895370

[A low-cost spatial light modulator for use in undergraduate and graduate optics labs](#)

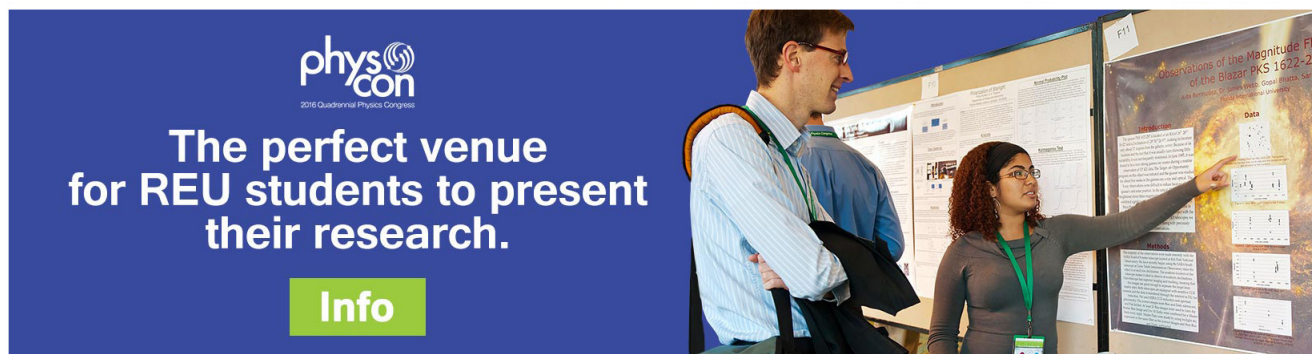
Am. J. Phys. **80**, 211 (2012); 10.1119/1.3666834


[Measuring Boltzmann's constant with a low-cost atomic force microscope: An undergraduate experiment](#)

Am. J. Phys. **74**, 873 (2006); 10.1119/1.2335475

[CageLab: A low-cost apparatus to demonstrate rotational motion for advanced undergraduates](#)

Am. J. Phys. **73**, 566 (2005); 10.1119/1.1791273




 2016 Quadrennial Physics Congress

The perfect venue
for REU students to present
their research.

[Info](#)

APPARATUS AND DEMONSTRATION NOTES

The downloaded PDF for any Note in this section contains all the Notes in this section.

Frank L. H. Wolfs, *Editor*

Department of Physics and Astronomy, University of Rochester, Rochester, New York 14627

This department welcomes brief communications reporting new demonstrations, laboratory equipment, techniques, or materials of interest to teachers of physics. Notes on new applications of older apparatus, measurements supplementing data supplied by manufacturers, information which, while not new, is not generally known, procurement information, and news about apparatus under development may be suitable for publication in this section. Neither the *American Journal of Physics* nor the Editors assume responsibility for the correctness of the information presented.

Manuscripts should be submitted using the web-based system that can be accessed via the *American Journal of Physics* home page, <http://ajp.dickinson.edu> and will be forwarded to the ADN editor for consideration.

Phase-sensitive detection in the undergraduate lab using a low-cost microcontroller

K. D. Schultz^{a)}

Hartwick College, Oneonta, New York 13820

(Received 25 January 2015; accepted 23 May 2016)

Phase-sensitive detection is an important experimental technique that allows signals to be extracted from noisy data. Commercial lock-in amplifiers, often used for phase-sensitive detection, are expensive and host a bewildering array of controls that may intimidate a novice user. Low-cost microcontrollers such as the Arduino family of devices might seem like a good match for learning about such devices, but making a self-contained device that includes a reference signal, a voltage input, a signal mixer, a filter, and a display is difficult. Here, we present the construction of a phase-sensitive detector (PSD) using an Arduino. © 2016 American Association of Physics Teachers.

[<http://dx.doi.org/10.1119/1.4953341>]

I. INTRODUCTION

Lock-in amplification and phase-sensitive detection are important techniques in experimental physics, and there have been many papers describing the pedagogical uses of these techniques.^{1–3} Commercial devices are expensive and can be intimidating for new users. Building a home-made instrument can be instructive,⁴ however, doing so requires advanced electronics skills that a student may not already have, putting the emphasis on the electronics and not the method. Conversely, it is possible to perform the mixing and filtering on a computer using a computer's sound card or other low-cost data acquisition devices to handle the input and output.⁵

In this paper, we describe a phase-sensitive detector (PSD) that uses the popular Arduino microcontroller and the Processing programming environment.⁶ The major design goal was to make the device as self-contained as possible, a task made difficult by the memory and hardware constraints of a typical microcontroller. While these devices are fantastic for controlling robots and basic data-logging, turning them into scientific instruments requires techniques that go beyond what is normally found in the literature. An added benefit is that these techniques can be used to build other types of instrumentation such as arbitrary function generators and fast digital multimeters (DMMs) out of a low-cost microcontroller that typically costs just a few tens of dollars.

II. ARDUINO

Arduino^{7,8} is a catch-all term for a family of open-source hardware based on Atmel micro-controllers with a pre-loaded bootloader, allowing the user to program the Arduino with the simpler Arduino language. Arduino programs, which are called *sketches* in the Arduino environment, are based on C and C++, but with a simpler instruction set. If one wishes, the more expansive AVR instruction set can be used.^{7,8} Programming and communication can be done via USB or through a set of on-board communication pins. There have been sixteen different Arduino-labeled boards produced at the time of writing, each of which has its own unique hardware and memory specifications. In this paper, the Arduino Uno R3 is used, simply because it was what was on hand and is one of the cheaper and more basic Arduinos. The heart of the Uno R3 is the Atmel ATmega328 microcontroller.⁹ The Uno R3 has 14 digital input/output pins, six of which provide Pulse Width Modulation (PWM) output. This unit also has six analog inputs and an on-board 16-MHz oscillator. The ATmega 328 has 32 kB of Flash memory to store programs and 2 kB of SRAM for variable storage. The program given in this paper is easily stored in the flash memory, but the limited SRAM places severe restrictions on the amount of data that can be taken and manipulated on-board the Arduino.

III. PHASE SENSITIVE DETECTION

A block diagram for phase-sensitive detection is shown in Fig. 1. The input stage of the PSD consists of two signals,

the input of interest and the reference signal. After the signals are acquired, the input signal may be amplified and filtered. After the input stage comes the mixing stage where the input and reference signals are combined. There are a number of ways of doing this, but in this implementation the signals will simply be multiplied. Finally, the output of the mixing stage is heavily filtered by a low-pass filter. Once again this can be done using analog circuitry, but here it will be done mathematically.

A. Generic phase sensitive detection

Let us assume that the input signal V_i and the reference signal V_r are given by

$$\begin{aligned} V_i &= V_1 + V_1 \sin(\omega_i t - \phi_i), \\ V_r &= V_2 \sin(\omega_r t - \phi_r). \end{aligned} \quad (1)$$

Here, V_i has a dc offset because the Arduino, like most microcontrollers, operates only with positive voltages. During mixing the offset for V_r is removed and therefore is not included in the following derivation. Upon multiplication of the two signals in Eq. (1), and making use of a trigonometric identity, the output of the mixing stage is

$$\begin{aligned} V_{\text{mix}} &= V_1 V_2 \sin(\omega_r t - \phi_r) \\ &+ \frac{V_1 V_2}{2} \left\{ \cos[(\omega_r - \omega_i)t - (\phi_r - \phi_i)] \right. \\ &\left. - \cos[(\omega_r + \omega_i)t - (\phi_r + \phi_i)] \right\}. \end{aligned} \quad (2)$$

At this stage Eq. (2) is nothing more than what we find in heterodyne detection in radio and optical engineering.¹⁰ The power of coherent detection is that small signals get amplified by a larger local oscillator signal, as is evident in Eq. (2). Upon mixing of the two signals, there are contributions to the signal at the original frequencies and at the sum and difference frequencies. This is the heart of PSD in that the output filter is set to reject all frequencies other than the difference frequency of the inputs. Typically in lock-in detection V_i and V_r are made to oscillate at the same frequency before entering the PSD. With this constraint the output V_o

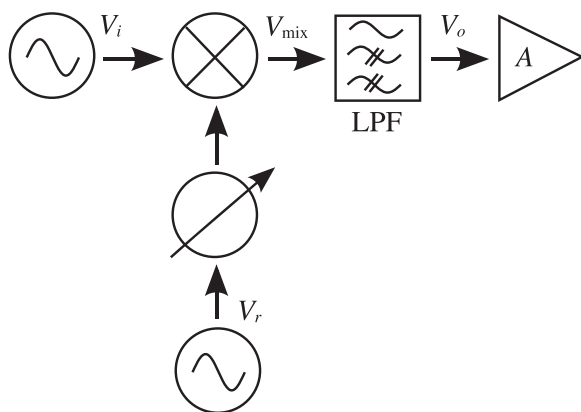


Fig. 1. Block diagram of a traditional phase sensitive detector; V_i and V_r are the input and reference voltages, respectively, LPF is a low-pass filter, and A is an amplifier. A phase-shifter is shown after V_r , but this component and the amplifier are not used in this project. See text for more details.

of the PSD simply depends on the phase difference between the two signals

$$V_o = \frac{V_1 V_2}{2} \cos(\phi_r - \phi_i). \quad (3)$$

The final effect of this filtering is to move our output from $\omega_{r,i}$ to dc (the aforementioned radio and optical engineers would call this homodyne detection). The stronger the filtering, the more noise is rejected so the signal-to-noise increases. However, in effect, the PSD is performing signal averaging, so with each factor of two increase in the signal-to-noise the collection time increases by a factor of four. The other disadvantage of a PSD is that by making the measurements at dc we are placing our signal where $1/f$ -noise dominates.¹⁰ Even though we have added no active amplification, the presence of a strong local oscillator can boost a weak experimental signal.

IV. IMPLEMENTATION

A self-contained phase-sensitive detector requiring only a computer for display, the Arduino, and as few passive circuit components as possible, places severe constraints and therefore requires some software and hardware exploits that are not commonly presented in introductions to microcontrollers. The structure of this section is to discuss the implementation of each of the sub-systems shown in Fig. 1. Briefly, and following Fig. 2, the Arduino synthesizes a sine wave output from a wavetable. As the output is updated from the wavetable, the input signal is quickly read. Finally, after the Arduino has cycled through the complete wavetable, mixing is performed mathematically onboard the Arduino, and the resulting waveforms are sent to the host computer running Processing for filtering and display.

A. Creating a reference signal

Creating the reference signal is in many ways the most significant constraint on the project. The Arduino does not have a true analog output. The only way to approximate an analog signal is to use the PWM pins of the Arduino. The internal clocks of the Arduino need to be manipulated such that the duty-cycle of the PWM is varied in such a way that with an external RC low-pass filter, a sine wave is created. The exact manipulations are discussed in what follows.

To generate a reference signal under these limitations a technique sometimes called “bit-banging”¹¹ is employed. Bit-banging takes advantage of the timers on the Arduino and PWM. In PWM, the duty-cycle (ratio of “on” to “off”

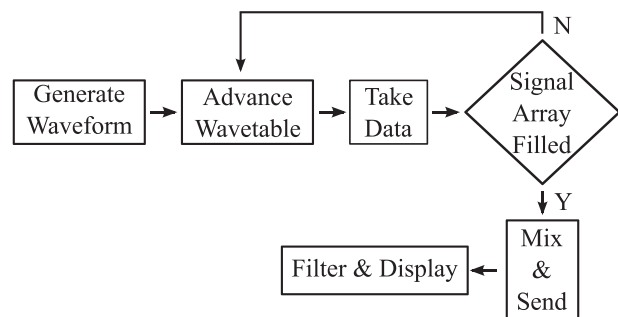


Fig. 2. All the steps through the mixing of the signals are performed on the Arduino; the filtering and display are accomplished in a Processing sketch.

times) of a square wave is modulated. The greater the duty-cycle, the longer the PWM pin on the Arduino is held HIGH. The value HIGH on an Arduino can take on different meanings depending on whether a pin is being used as an input or an output. Here HIGH or LOW refers to a digital write command to an Arduino pin, resulting in 0 V for LOW and 5 V or 3.3 V for HIGH, depending on the Arduino model. Passing that square-wave through a low-pass filter, effectively averaging the signal, produces a dc voltage that increases with the duty-cycle. The final step is to rapidly change this dc voltage so that the desired waveform, in this case a sine-wave, is synthesized.

This paper will only briefly sketch the relevant idea. For details, readers should refer to the comments in the source code for this project, the “bit-banging” paper,¹¹ or the ATmega328 datasheet.⁹ The technique relies on two counters on the Arduino. Timer 1, TCNT1, is a 16-bit timer/counter that is configured as an 8-bit counter. TCNT1 runs at the full Arduino clock frequency (16 MHz) and produces the PWM output signal. Timer 2, TCNT2, runs slower than TCNT1 by a factor of eight and is used to step through the pre-generated wavetable representing the reference waveform. In what follows, the convention with “n” at the end of a name represents a generic counter, and if it is replaced by a number that will designate a specific counter.

Figure 3 shows the basic idea behind “bit-banging.” Figure 3(a) shows the slow counter (TCNT2) while Fig. 3(b) shows the fast counter (TCNT1) before and after its Output Compare Registers (OCRnA) are changed. Finally, Fig. 3(c) shows the PWM output of the Arduino. When the value of the timer TCNTn matches the register OCRnA a flag is set. The outcome of the OCRnA flag can exhibit different behavior depending on how the Arduino is programmed. For the Arduino PSD, when the OCR1AL flag is set, the PWM pin is made to go LOW [Fig. 3(c)], but TCNT1 keeps incrementing until it overflows and TCNT1 starts counting again from zero [Fig. 3(b)]. At this point the PWM pin goes HIGH again, and the OCR1LA flag is reset. By changing the value of OCR1AL the duty-cycle of the PWM signal is changed, as illustrated for later times in Fig. 3, and consequently the dc voltage that is used to construct V_r is also changed. The pre-programmed wavetable for V_r is used to update OCR1AL. The Arduino is programmed to call an interrupt when the TCNT2 timer, running eight times slower than TCNT1, reaches the value in the register OCR2A. This interrupt does two things: updates OCR1AL to the next value of the wavetable, and gets an input voltage for the ADC. OCR2A is pre-set in the Arduino code such that the frequency of V_r is given by

$$f_r = \frac{\text{rate of TCNT2}}{\text{OCR2A} \times \text{wavetable length}}. \quad (4)$$

The last step in generating V_r is to place a simple low-pass RC filter on the PWM pin to get a smooth sine-wave.

B. Getting the signal in

The Arduino ADC is a 10-bit successive approximation circuit connected to an 8-channel multiplexer, although only six channels are used on the Arduino UNO. The measurements are single-ended and normally referenced to a 1.1 V on-board reference voltage, although it is possible to use an external reference voltage. Because interrupts are used to

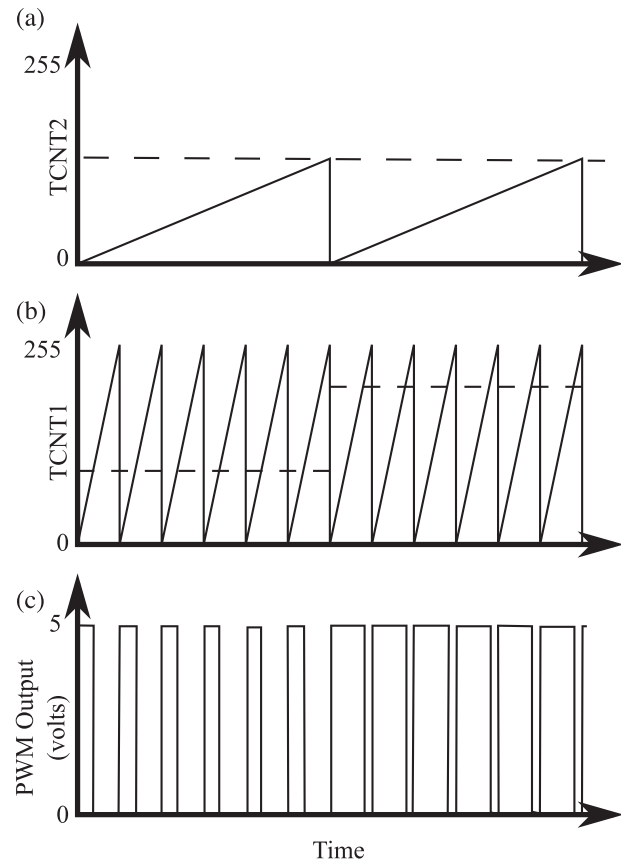


Fig. 3. The “bit-banging” technique requires a slow counter (a), a fast counter (b), and a pulse-width modulated signal (c). The dashed line in (a) represents OCR2A, the counter value that calls the necessary interrupt to update OCR1A and read in the input voltage for mixing. The dashed line in (b) represents the changing OCR1A, which sets the duty-cycle for the PWM. The reference signal that modulates the experiment is the output of a low-pass RC circuit with (c) as the input.

create an analog output for the reference signal, it is important to ensure that anything that happens during the interrupt is quick, which is why in Fig. 2 the data is sent for filtering after the wavetable has been completely cycled through. Serial calls are slow.

According to ATMEL, the ultimate sampling rate for single-ended measurements is limited by the ADC clock speed. The ADC clock speed is derived from the main system clock, and for maximum resolution should be between 50 and 200 kHz. However, for purposes of this project satisfactory resolution and accuracy are obtained by setting the ADC clock to a speed of 1 MHz. The successive approximation circuit requires 13 clock cycles, giving a sampling rate of 77 kHz,¹² which is much faster than this project needs.

C. Phase shifts and mixing

Because the values of the reference signal are pre-loaded into the Arduino memory, changing the phase while data is being taken is difficult. Often, however, the ability to shift the relative phase between V_i and V_r is needed to maximize the signal. Additionally, sometimes what is needed in a measurement is the phase shift between the reference and input signals. To meet these needs, a cue is taken from dual-phase lock-in amplifiers.¹³ These amplifiers use a reference

waveform and a so-called quadrature reference signal, which is just the reference waveform shifted by 90° . Both the in-phase $V_r \cos(\omega_r t)$ and quadrature $V_r \sin(\omega_r t)$ reference waveforms are separately mixed with the signal input as discussed in Sec. III. The result is two mixed signals, the in-phase (I) and quadrature (Q) waveforms. It is the I and Q signals that are sent from the Arduino to the host computer for filtering and display. The phase ambiguity is removed by calculating the magnitude of the mixed signal $R = \sqrt{I^2 + Q^2}$. The phase difference between the input and reference signals is found $\phi = Q/I$.

D. Filtering and display

Once the Arduino finishes running through the waveform, it takes the collected data and sends it via USB to a computer running the Processing IDE.^{14,15} Processing is a programming language and development environment that was designed to make it easier for the arts community to become software literate. Like the Arduino, it has a vibrant community that has produced numerous tutorials, examples, and books that make learning Processing relatively easy. Processing is also a direct forebear of the Arduino development system and therefore makes it a natural fit with the Arduino side of this project.

In this project, the Processing sketch (Processing-talk for program) initiates serial communications with the Arduino, and once communications are established the Arduino sends the I and Q data to the Processing sketch. The Processing sketch applies a recursive, single-pole, low-pass filter¹⁶ to the array that is to be displayed (I, Q, R, ϕ). A single-pole recursive filter can be written as

$$y[n] = (1 - x_{\text{decay}})x[n] + x_{\text{decay}}y[n - 1], \quad (5)$$

where $x_{\text{decay}} = e^{-2\pi f_c}$, y is the output of the filter, x is the input data, and f_c is the time constant of the filter. Mathematically, this filter is identical to a single-pole RC filter in electronics. The stronger the filtering, the better it is

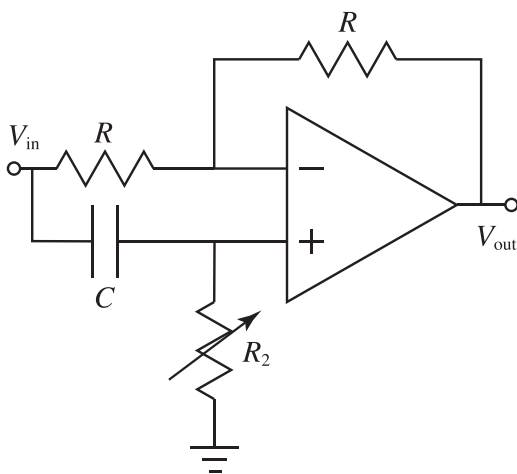


Fig. 4. Phase shifting circuit for testing the Arduino PSD. The input of this circuit is connected to the Arduino output corresponding to the reference signal. For testing, the output of this circuit was fed into the Arduino input used to capture V_s . The op-amp used was a standard 741. Changing R_2 shifts the phase through 180 degrees. The midpoint of the R_2 potentiometer and C are chosen such that $1/R_2 C \sim \omega_r$. Sample component values are $R = 10 \text{ k}\Omega$, $R_2 = 10 \text{ k}\Omega$, and $C = 100 \text{ nF}$.

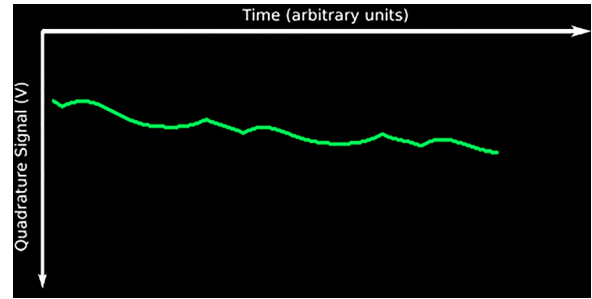


Fig. 5. A screenshot of the Processing output, the displayed axes were overlaid onto the Processing screenshot to aid understanding of this figure. The display can be set to show the phase, magnitude, I , or Q signal as a function of time. The display shows a numerical value of the quantity displayed as well as a graphical display. Here, the in-phase signal of the Arduino PSD is displayed as R_2 of the phase-shift test-circuit is varied. Future plans for this device are to make a more useful display.

for lock-in detection, since inadequate filtering causes the output to oscillate at the reference frequency. Algorithms for calculating coefficients for higher-order filters with faster roll-off are also available.¹⁶ These may be of interest because single-pole filters are limited to attenuations of 6 dB/decade.

V. TESTING

To test this project, an all-pass phase-shifter was built.¹⁷ Figure 4 shows the circuit used to test the Arduino. The voltage gain for this circuit is 1 V/V, so the amplitude of the output is unchanged with respect to the input voltage. The high-pass RC filter at the non-inverting terminal of the op-amp controls the amount of phase-shift at the output. At $\omega_0 = 1/RC$ the phase-shift is 90° and changes by $90^\circ/\text{decade}$. By varying R_2 , the 90° point shifts and moves our operating frequency along the phase plot. To test the Arduino PSD, the reference signal generated by the Arduino is sent to the input of the phase-shifting circuit. The output of the phase-shifting circuit becomes the input signal to the PSD. As mentioned in Sec. IV, the final output of the PSD is a graphical display of either I, Q, R , or ϕ as a function of time. At the time of writing this paper, the display capabilities of this project are very rudimentary. Nevertheless, Fig. 5 shows the I signal as displayed in the Processing sketch as R_2 is varied, demonstrating that the PSD is operating correctly.¹⁸

VI. CONCLUSIONS

A phase-sensitive detector using an Arduino microcontroller has been described. The main design goal of the project was to make the PSD as self-contained as possible, namely, that an external reference signal was not needed and that any phase corrections were unnecessary. Aside from the Arduino and a computer the only other hardware needed are a resistor and a capacitor. Furthermore, the techniques used for this project can be used in other projects involving using the Arduino microcontroller as a standalone, low-cost, scientific instrument. The “bit-banging” technique is not limited to sine-waves, but can be used to output any synthesized waveform that may be required. More complicated waveforms can be created outside of the Arduino environment and loaded into the EEPROM allowing faster execution and freeing up regular memory.¹¹

ACKNOWLEDGMENTS

The author would like to thank Professor Lawrence Nienart of Hartwick College for his advice and encouragement throughout this project. The author would also like to thank the reviewers for helpful suggestions.

^{a)}Electronic mail: schultz@harwick.edu. URL: www.HartwickChaosLab.github.io

¹W. Yang, "Teaching phase-sensitive demodulation for signal conditioning to undergraduate students," *Am. J. Phys.* **78**, 909–915 (2010).

²K. G. Libbrecht, E. D. Black, and C. M. Hirata, "A basic lock-in amplifier experiment for the undergraduate laboratory," *Am. J. Phys.* **71**, 1208–1213 (2003).

³R. Wolfson, "The lock-in amplifier: A student experiment," *Am. J. Phys.* **59**, 569–572 (1991).

⁴P. Horowitz and W. Hill, *The Art of Electronics*, 2nd ed. (Cambridge U.P., Boston, 1989).

⁵M. González, G. Santiago, V. Slezak, and A. Peuriot, "Simple synchronic detection at audio frequencies through a PC sound card," *Rev. Sci. Instrum.* **78**, 055108-1–4 (2007).

⁶Up-to-date Arduino and Processing code can be found on my github page, which can be forked and modified. <github.com/HartwickChaosLab/Arduino-Phase-Sensitive-Detector>.

⁷Arduino Project Web Site; <<http://www.arduino.cc>>.

⁸M. Margolis, *Arduino Cookbook*, 2nd ed. (O'Reilly Media, Sebastopol, 2011).

⁹ATmega Data Sheet <<http://www.atmel.com/devices/atmega328.aspx>> (accessed October 1, 2015).

¹⁰P. C. D. Hobbs, *Building Electro-Optical Systems: Making It All Work* (John Wiley & Sons, New York, 2000).

¹¹J. Thompson, "Advanced Arduino sound synthesis," *Make* **35**, 80–88 (2014); available at <http://makezine.com/projects/make-35/advanced-arduino-sound-synthesis/>.

¹²Atmel white paper:AVR20, "Characterization and Calibration of the ADC on an AVR," available at <<http://www.atmel.com/images/doc2559.pdf>> (accessed October 1, 2015).

¹³M. Meade, "Advances in lock-in amplifiers," *J. Phys. E.* **15**, 395–405 (1982).

¹⁴Processing Project Web Site; <<http://www.processing.org>>.

¹⁵C. Reas and B. Fry, *Processing: A Programming Handbook for Visual Designers and Artists*, 2nd ed. (MIT Press, Cambridge, 2014).

¹⁶S. W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing* (California Tech. Pub., San Diego, 1997).

¹⁷S. Franco, *Design with Operational Amplifiers and Analog Integrated Circuits*, 3rd ed. (McGraw Hill, Boston, 2002).

¹⁸See supplementary material at <http://dx.doi.org/10.1119/1.4953341> for the Processing and Arduino code as described in this article can be found here, as well as a short tutorial on how to use the code. Up-to-date and modifiable code can be found by going to Ref. 6.

Complex artificial halos for the classroom

Markus Selmke^{a)} and Sarah Selmke

Universität Leipzig, 04103 Leipzig, Germany

(Received 16 November 2015; accepted 23 May 2016)

Halos represent a common and imposing atmospheric optics phenomenon whose displays are caused by tiny air-borne ice crystals. Their variety stems from a certain set of orientation classes to which these crystals belong. We present a robust and inexpensive device, made of modular components, that allows for the replication of most of these orientation classes in the laboratory. Under the illumination of light, the corresponding artificial halo counterparts emerge. The mechanical realization of this device allows a thorough understanding and demonstration of these beautiful atmospheric optics phenomena. © 2016 American Association of Physics Teachers.

[<http://dx.doi.org/10.1119/1.4953342>]

I. INTRODUCTION

In nature, halos can occur all year around due to innumerable refracting and reflecting ice crystals, present in either cold air or in high (and therefore cold) cirrus clouds. Depending on the specific weather conditions, the crystals causing these phenomena typically are either hexagonal columns, hexagonal plates, or both. While many additional shapes and forms can occur, these two suffice to account for most halos. A rich variety of halos exists, comprising arcs, spots, and rings, many of which are vividly colored. The origin of this variety is a set of different classes of stable ice crystal orientations. Still, there is a further source of the wide range of halos as each class of orientations allows different paths for light to travel through these crystals as they take on all orientations allowed in such an orientation class. The mechanism of most halo types can be explained by fake caustics and may be understood with similar

concepts as the rainbow. A recent article¹ in this journal gave an introduction to the topic, while details on the nomenclature and classification schemes can be found on Cowley's webpage² and in several well-illustrated books^{3–5} or articles.^{6,7} So far, direct demonstrations of the phenomena's varied displays have remained difficult due to the aforementioned complexity.

II. A MODULAR DEVICE

Using only a single crystal, several experimental works since Bravais⁸ have demonstrated the feasibility of constructing dedicated electromechanical devices that can replicate the orientation classes associated with all known regular crystal halos,^{1,9–12} except for the Lowitz orientations.¹³ While impressive chemical approaches also exist,¹⁴ specifically for the circular halos,^{9,15} macroscopic demonstrations have the clear advantage of a better visualization of the

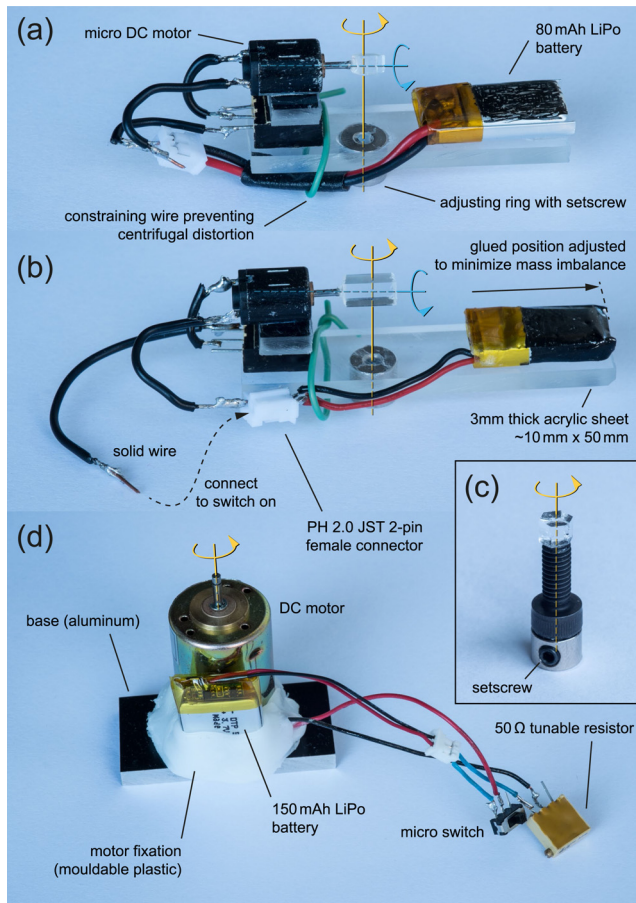


Fig. 1. The modular halo machine turns the crystals about the indicated axes: (a) module generating classical Lowitz orientations when mounted; (b) module generating the singly oriented columns orientations when turned on and the Parry orientations when kept off and mounted; (c) module generating plate orientations when mounted; (d) tunable dc motor to rotate the different modules.

involved and fake caustic mechanisms. Keeping the general construction idea, our modular approach simplifies past attempts for the non-trivial two-axis orientation classes. It allows a spectacular recreation of the previously reported halo demonstrations, except for the circular one.^{1,11} However, as compared preciously published works, our approach includes artificial Lowitz halos as well. Using a single commercially available small hexagonal glass rod¹⁶ and

parts obtainable from any well-resourced electronics store, the setup shown in Fig. 1 is straightforward to construct, requiring only limited soldering skills. In total, two thin (~ 3 mm) and one longer (~ 8 mm) slice cut from the purchased glass rod are needed. We note that the use of glass (or any $n > \sqrt{2}$ material) prevents a limited number of halo types (including, unfortunately, the common circumhorizon and the circumzenith halos. Luckily, they can be demonstrated straight away with a cylindrical glass filled with water.) by total internal reflection restrictions on some light paths.⁹

The device consists of four modules as described here. A single base motor [Fig. 1(d)] that swivels either of three other modules and these modules themselves. The base should be sturdy as it needs to withstand any remaining imbalance of the mounted modules when they are rotated. We mounted a 3.7 V LiPo battery-powered¹⁷ dc motor¹⁸ (2-mm shaft) with moldable plastic.¹⁹ While not necessary, a rotation speed control was added using a fine-tuneable resistor to the motor circuit.

Two of the three mountable modules were constructed on acrylic plastic sheets, matched in width to the small LiPo battery packs [see Figs. 1(a) and 1(b)]. These acrylic pieces were sawed off of a larger 3-mm-thick sheet. Both modules consist of a micro-motor glued on top of a smaller cut of the original acrylic sheet, which itself is glued on the same type of tuneable resistor used in the base circuit. An adjusting ring/shaft collar,¹⁸ as found in radio-controlled car equipment, of the same bore size as the base motor's shaft is glued right under the center of the crystal. The wiring was done using thin solid wires and PH 2.0 JST 2-pin connectors for the battery; this allows for convenient recharging and also serves as a switch (plugging/unplugging one wire) turning the module on/off for experiments. For the former task, any LiPo charging circuit¹⁷ with the corresponding male JST connector can be used. As the long wires may significantly distort during the spinning motion, constraining them was found to be necessary. For the same reason, cables were kept as short as possible.

The crystals were carefully superglued directly to the micro-motor's shaft. Apart from a steady hand, a level support during glueing turned out to be crucial for successful docking; small acrylic sheet pieces were used for this purpose. For the column-like crystal, this procedure also ensures that the crystal may adopt Parry-orientations, where one of the column's side faces is horizontal as depicted in the inset

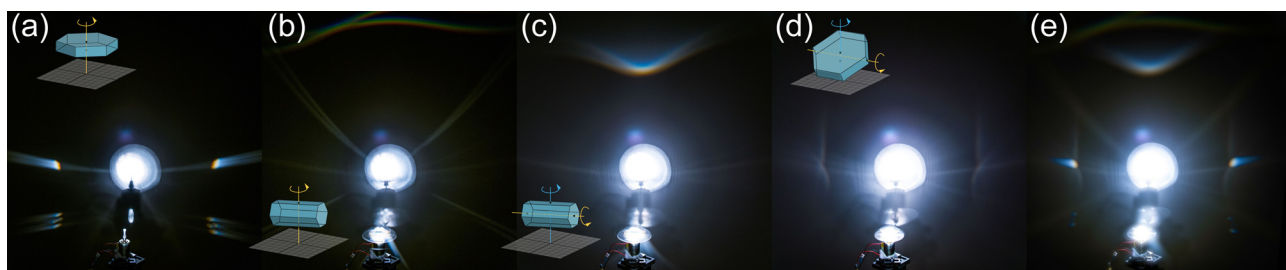


Fig. 2. The different halo machines/modules illuminated by white parallel light (a focusable Mini Maglite flashlight) produce various artificial halos: (a) plate orientations [cf. Fig. 1(c)] give rise to the parhelic circle, parhelia, and subparhelia (their splitting being due to crystal imperfections); (b) Parry orientations [cf. Fig. 1(b), micro-motor off] lead to colored Parry and white heliac arcs; (c) horizontal columns [cf. Fig. 1(b), micro-motor on] lead to tangent arcs (or the circumscribed halo for high inclinations); (d) Lowitz orientations [cf. Fig. 1(a)] lead to Lowitz arcs (Refs. 2–5 and 13). When combined, a complex artificial halo (e) is generated. The precise angular coordinates of the features differ (Refs. 1 and 9–12) from ice halos since the refractive indices are different ($n_{BK7\text{ glass}} \approx 1.52 \neq n_{\text{ice}} \approx 1.31$). However, because the dispersion $n(\lambda)$ is similar ($n_{\text{blue}} > n_{\text{red}}$), the general coloring is the same, i.e., red color towards the direct transmission spot (the sun in natural displays), indicating larger minimum deflection angles for blue light (Refs. 1–5) The supplementary material (Ref. 22) contains a similar series of images but from an angled view and using the laser diode of Fig. 3, showing the lower arcs and the subsun as well.

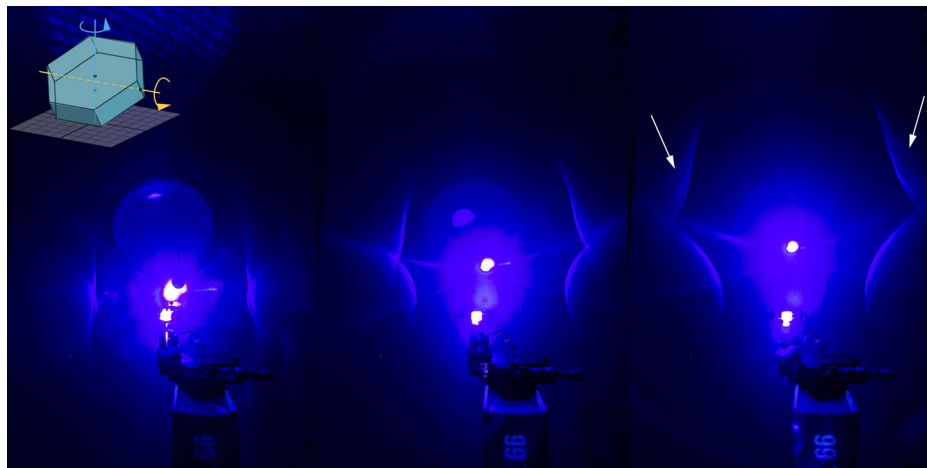


Fig. 3. Artificial Lowitz arcs (Ref. 13) generated by the mounted and spinning module in Fig. 1(a) with increasing light source inclinations (0° – 30°). A low-power, focusable, blue laser diode ($\lambda = 405$ nm, $P < 1$ mW) was used for illumination. The arrows indicate a faint feature/arcs appearing distinctly only at large elevations (Ref. 13).

of Fig. 2(b).^{2–7} The magnetic feature of the micro-motor gives a sufficient restoring force against rotation when left off such that the crystal remains with its side face flat the way it was glue-attached even when the base motor rotates the module of Fig. 1(b). By contrast, turning on the micro-motor corresponds to horizontal column orientations [singly oriented columns, inset Fig. 2(c)].

Alternatively, using a second long cut from the glass rod, one may construct a dedicated Parry orientation module similar to the plate orientations' module of Fig. 1(c), as both orientation classes require a single rotation axis only. The intended spinning of the two larger modules requires a counterbalance on the opposite sides of the sheets, which the small LiPo batteries powering the micro-motors provide. A test for sufficient balancing turned out to be the ability of the construction to rest level on the adjusting ring. The last module consists of a top-polished plate-like crystal piece glued on a screw and on top of an adjustment ring [see Fig. 1(c)]. The screw ensures that the plate crystal will be level with the crystals on the other two modules so that later composition of each module's imaged display is possible, provided the illumination is kept fixed. The polished top surface causes subparhelia when illuminated at an angle from below, as seen in Fig. 2(a).

III. SIMPLE AND COMPLEX HALOS

Each module may now be mounted on the base motor's shaft and fixed using the setscrew [cf. Fig. 1(c)]. In operation

and under illumination, each will then produce a projection of the corresponding artificial halo visible to the unaided eye. Since parallel light is necessary, one may either use a focusable flashlight or laser diode. The former has the advantage of producing colorful halos (Fig. 2), while the latter allows for better defined and brighter halos (Fig. 3). Having modules for the different types of halos mounted on a single base, one may superpose the images of each to produce an artificial counterpart to nature's stunning but somewhat rarer complex halo displays,^{2,4,5} as shown in Fig. 2(e).

Capturing the images may require long exposures depending on the light source brightness. For example, the images in Figs. 2–4 were recorded using approximately 20-s exposure time at the lowest ISO settings. As the light source inclination is changed, (artificial) halos change significantly.^{2–5} An example of such a dependence is shown in Fig. 3, where Lowitz arcs change in shape as the inclination increases from about 0° to $\sim 30^\circ$. Eventually, a feature only distinctly visible for high elevations is revealed (see arrows).¹³ Similarly, a study of the singly oriented columns machine shows a gradual transition from tangent arcs to the circumscribed halo, as shown in Fig. 4. Using the freely available computer simulation program HaloSim,²⁰ one may further study the generated halos and infer the responsible ray paths through the crystals for each feature of a given display. While obstruction of the light directed at the crystal is a concern, especially at larger incident light inclinations from below, the results show a good qualitative agreement with the expected displays.^{2–5}

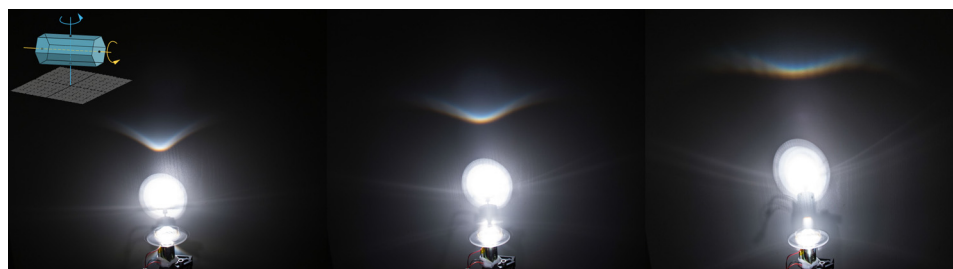


Fig. 4. Tangential arcs transitioning to the circumscribed halo for increasing light source inclinations (0° – 30°), both produced by the mounted and turned-on module in Fig. 1(b). The lower tangential arc is covered by the machine in these pictures.

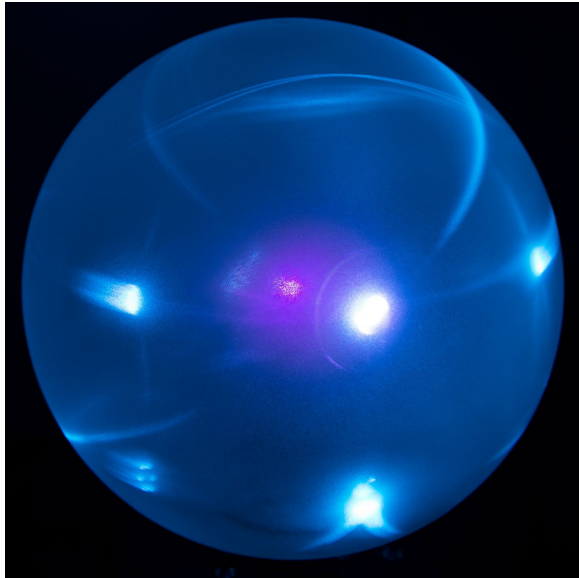


Fig. 5. Similar experiment as in Fig. 2, but using the laser diode of Fig. 3 and a spherical projection screen. The image is a superposition of 4 individual ones (Ref. 22).

On a closer look, distortion affects the artificial displays. For instance, while the natural parhelic circle spans across the sky at constant elevation (i.e., parallel to the horizon),^{2–5} the artificial counterpart is seen to bend upwards [cf. Fig. 2(a)]. In fact, here it's a hyperbola. This is because the deflected rays of constant inclination e against the floor¹ (its normal being the z -axis) lie on the surface of the cone $[\tan^2(\pi/2 - e)]z^2 = x^2 + y^2$, which projects on a vertical screen or the wall at a distance $x = l$, yielding a hyperbola as the conic section. To circumvent such distortions and to better approximate the natural phenomena, a spherical projection screen may be used. We used a hollow and clear acrylic sphere of diameter $\varnothing = 12''$ with a $\varnothing = 3.5''$ opening. Having the blue laser diode in mind, we coated it with a single layer of white UV fluorescent spray paint. A spherical lamp shade should work as well. Small holes drilled at different heights into the sphere allow light to be directed at the machine mounted at the sphere's center.²² Only after switching on a module is the screen carefully put in place and the dc motor turned on. Now, the artificial parhelic circle spans the screen at constant height just as the natural phenomenon, as shown in Fig. 5. Also, a heliac arc from Parry orientations now properly surround the zenith (top of the sphere). A further advantage of such a spherical enclosing screen is the enhanced brightness of the display that can be achieved, owing to the closer distance and the smaller area upon which the deflected light projects.

IV. CONCLUSION

The inexpensive and easy-to-construct modular device presented here may be used in classroom demonstration experiments and explanations of halo phenomena in all their diversity. A majority of the most common ice halos (e.g., excluding the circular 22° , the circumhorizon halo, and the circumzenithal halo) may be artificially reproduced using the presented setup. Similar to rainbow experiments,²¹ ice halos serve as a good alternative example for the physics of refraction and caustics.¹ These demonstrations may serve to demystify and bring attention to the sometimes complex but always stunning ice halo displays that are visible all year and all around the world.

^aElectronic mail: markus.selmke@gmx.de. URL: <http://photonicsdesign.jimdo.com>

¹M. Selmke, "Artificial Halos," *Am. J. Phys.* **83**(9), 751–760 (2015).

²L. Cowley, "Atmospheric optics," <<http://www.atoptics.co.uk/halosim.htm>>.

³R. Greenler, *Rainbows, Halos and Glories* (Cambridge U.P., England, 1980).

⁴W. Tape, *Atmospheric Halos*, Antarctic Research Series (American Geophysical Union, Washington, D.C., 1994).

⁵W. Tape and J. Moilanen, *Atmospheric Halos and the Search for Angle χ* (American Geophysical Union, Washington, D.C., 2006).

⁶W. Tape, "Analytic foundations of halo theory," *J. Opt. Soc. Am.* **70**(10), 1175–1192 (1980).

⁷G. P. Können, "Polarization and intensity distributions of refraction halos," *J. Opt. Soc. Am.* **73**(12), 1629–1640 (1983).

⁸A. Bravais, "Mémoire sur les halos et les phénomènes optiques qui les accompagnent," *J. École R. Polytech.* **31**(18), 1–270 (1847).

⁹M. Vollmer and R. Tammer, "Laboratory experiments in atmospheric optics," *Opt. Express* **37**(9), 1557–1568 (1998).

¹⁰M. Vollmer and R. Greenler, "Halo and mirage demonstrations in atmospheric optics," *Appl. Opt.* **42**(3), 394–398 (2003).

¹¹M. Großmann, K.-P. Moellmann, and M. Vollmer, "Artificially generated halos: Rotating sample crystals around various axes," *Appl. Opt.* **54**(4), B97–B106 (2015).

¹²S. Borchardt and M. Selmke, "Intensity distribution of the parhelic circle and embedded parhelia at zero solar elevation: Theory and experiments," *Appl. Opt.* **54**(22), 6608–6615 (2015).

¹³M. Riikonen, L. Cowley, M. Schroeder, M. Pekkola, T. Öhman, and C. Hinz, "The Lowitz Arcs," *Weather* **62**(9), 252–256 (2007).

¹⁴Z. Ulanowski, "Ice analog halos," *Appl. Opt.* **44**(27), 5754–5758 (2005).

¹⁵A. Cornu, "Sur la reproduction artificielle des halos et des cercles parhéliques," *Comtes Rendus Ac.* **108**, 429–433 (1889).

¹⁶Edmund Optics®, 4-mm Aperture, 25-mm L High NA Hexagonal Light Pipe, \$75.

¹⁷Most electronic stores will have these parts, for instance: <<https://www.adafruit.com>>.

¹⁸Most radio-controlled vehicle supply stores will have these parts, for instance: <<http://www.conrad.com>>.

¹⁹Gorilla plastic: <<http://www.gorilla-plastic.de>>.

²⁰L. Cowley and M. Schroeder, HaloSim simulation program; <<http://www.atoptics.co.uk/halo/halfeat.htm>>.

²¹G. Casini and A. Covello, "The 'rainbow' in the drop," *Am. J. Phys.* **80**(11), 1027–1034 (2012).

²²See supplementary material at <http://dx.doi.org/10.1119/1.4953342> for additional images, including the spherical projection screen setup.