AAVSO Photoelectric Photometry (PEP) Manual

Version 3.0
5 April 2025

Tom Calderwood

High quality photometry of bright, astrophysically interesting stars.

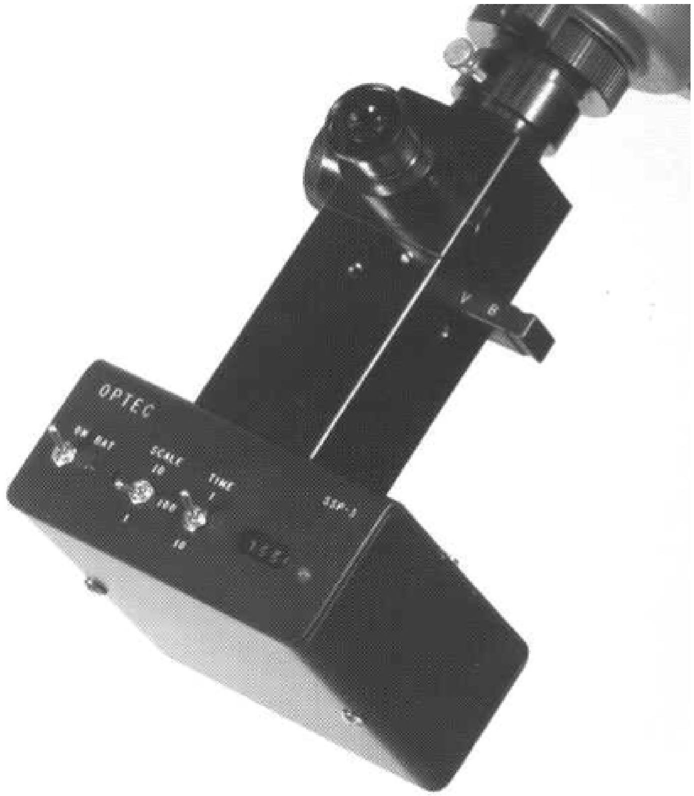
The AAVSO photoelectric section was founded in the late 1970s. We use old-school technology, but we can get superior results on bright stars. Compared to imaging systems, our equipment is fundamentally simpler to calibrate and operate, and data reduction is straightforward. What we lack in sensitivity, we make up for in quality. With properly chosen targets and careful technique, we remain a viable research group. Do not be intimidated by the length of this work. Fully half of it is appendixes that can perused as needed. Chapter One is the most important reading.

If you are a newcomer to PEP, think of your first year as an apprenticeship. Expect to work only in V band as you get "the hang" of operating the photometer and reducing the raw data.

This document lacks the polish of other AAVSO manuals, but I think you will find it entertaining reading. The content lies somewhere in between a cookbook and a reference book. I will try to provide a wide, but not too deep overview of the equipment and practice of photometry with single-channel photometers. I will fudge on the details, occasionally, in the service of clarity.

An important development is that Optec Corporation no longer manufactures the photometers that almost all of us use. Jerry Persha, the inventor of the Optec devices, can still supply us with some hardware support.

Tool of the Trade



Optec SSP3 Photometer © Optec Corporation

Revision history

17 January 2017	First general release
28 May 2020	Release 2.0. Some equations fixed, some material simplified, less-important stuff removed. Another revision expected in 2021.
5 April 2025	Release 3.0. Major revision. Some old material removed, expanded appendix section.

Acknowledgments

Thanks to Erwin van Ballegoij for reviewing a draft.

Table of Contents

Chapter 1 — Observing

1.1 Scopes and mounts

At base level, a PEP observer needs a telescope with a tracking mount, a photometer, and at least one photometric filter. The optical tube almost needs to be cassegrain, though a compact refractor is usable. The Optec photometers weigh about 2.5 pounds—you don't want them at the front end of a Newtonian. Likewise, the photometer will turn a long refractor into an amazing pendulum that is no fun to balance. In either case, the photometer, which has a right-angle eyepiece you must look through, would swing all over creation as the tube is moved around the sky.

Your mount, surprisingly, need not be equatorial. We will aim the photometer so that the target star is dead-center in the field of view. This means we don't care about the field rotation that affects altitude-azimuth mounts. The mount *does* need to track the sky automatically and well. A GOTO mount is not strictly necessary, but it makes a big difference in the ease of operation. If you plan to operate a GOTO mount strictly with a hand-control, be sure that the controller supports "user defined objects." Life without this is an incredible headache because the star catalogs in the controllers will not have all the stars you need, or will not have them easily accessible for the back-and-forth pointing we use.

If you plan to operate your mount via a computer, there should be no difficulty configuring user objects, and it will be easier to command slewing than with a hand controller. But you'll still use the handpad for fine positioning and you need slow speed adjustments that really work. I once bought a mount that advertised a wonderful range of slew speeds, but it turned out that the two slowest were unusable, causing endless problems.

If your mount is not computerized, it is essential that the manual slow motion controls work very smoothly, without backlash. A further consideration involves fork (or half-fork) mounts. The photometer sticks out a long way from the back end of the optical tube, and it will hit the base of the mount if you try to swing it between the tines of the fork. If you operate the mount in alt-az mode, a considerable portion of the sky near the zenith will likely be inaccessible. On an old Meade LX-200, you can only get to about 65° altitude. Of course, you can wait for objects to sink to a lower elevation, but sometimes you really want to shoot straight up. If you operate in equatorial mode, some parts of the sky will still be out of reach, but you can aim overhead.

German equatorial mounts are fine, unless they are tripod-mounted and the photometer case hits the legs. Finally, your GOTO mount need not have perfect slewing. However, it is important for slew errors to be reasonably *predictable*. When you first slew to your program or comparison star, you want to know where it is likely to be relative to the center of the field. If your desired star is not remarkable in color or brightness, you may find yourself having to choose among possibilities, and end up taking data on the wrong one.

Above all, act carefully if buying a new telescope or mount. Ask an experienced observer if it is really suited to your needs.[1]

1 A final word on scopes: you don't need expensive, super-corrected optics. We work right on the optical axis, where aberrations are at a minimum. Increased aperture does more good than a tighter point-spread function.

1.2 Photometers and filters

Generation 1 SSP3 Generation 2 SSP5

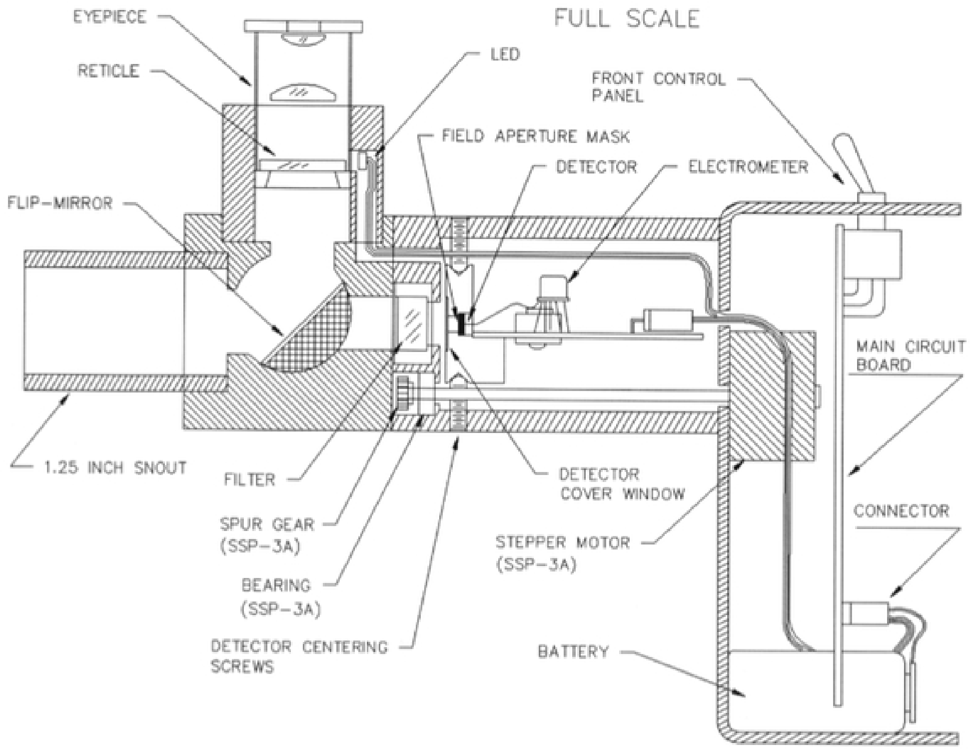
Your choice for a photometer will likely be an SSP3 or SSP5 by Optec.. They are available used as “Generation 1” or “Generation 2,” and with or without motorized filter switching. Only Generation 2 models are computerized and thy have some definite advantages. You may have to be patient for one to appear on eBay or AstroMart, but the devices are out there and AAVSO has some "loaners" available. If you are new to handling scientific instruments, you probably want to start with an SSP3. These photometers are almost indestructible. I've dropped them four feet onto concrete and had them survive. The SSP5 is more delicate, and if it suffered the same treatment you might be out hundreds of dollars for a new photomultiplier tube. The trade-off is that the SSP5 can see much fainter stars. Of course, it costs more to buy in the first place. Another attraction of the model 3 is that it can be operated off of an internal 9V battery.[2]

The Gen. 1 photometers have a four-digit LED display that shows the counts, and the operator would manually record the numbers. Gen. 2 can either display those counts, or send them to a computer for automatic logging. An advantage here is that the computerized log will handle counts as high as 65535, whereas the on-board display can only go to 9999 (but see Appendix I). If you are observing a var/comp pair where one star is way brighter than the other, the bright star might overflow the display. Either generation might be fitted with a motor for switching among filters, making it an “A” model (SSP3A/SSP5A). Nonmotorized models have a sliding metal bar with mounting holes for two filters. You push-in/pull-out the “slider” to effect a filter change. Motorized models have space for at least six filters in the slider. Jerry Persha has released a control/acquisition/reduction program, SSPDataQ for use with the Generation 2 photometers, though it is no longer supported. It runs only on Windows,

2 A rechargeable 200mAh battery can be used, though I have had trouble with degradation in nickel-hydride cells. Alkaline batteries may fail in cold weather.

and communicates over an RS-232 link. The Gen. 2 data protocol is not complicated. You could write your own software to control it.

The physical packages for the SSP3 and SSP5 are almost identical. They have a 1.25” snout that slides into a conventional focuser. The forward rectangular box contains the optical bench, and the box at the rear has the processing electronics. A Gen. 1 SSP3 is illustrated below.



Generation 1 SSP3, © Optec corp.

The sensitivity of the SSP3 is such that you will want at least an eight-inch telescope to have a reasonable selection of targets (ten-inch if you want to do B band). Stars in the PEP program are usually bright, but there's no sense artificially limiting your choices. The SSP5 can go about five magnitudes fainter than the SSP3 (or seven magnitudes if you buy the extended-sensitivity photomultiplier), hence its attraction for experienced observers.

When outfitting your photometer, you will want at least a V band filter, and your next choice should be B band.[3] If you buy a photometer that has been sitting around for years, plan to buy new filters (over time, the cement that holds the layers of colored glass in the old-style filters together deteriorates and becomes cloudy). We now buy "interference" filters from Chroma Technology. The Chroma filters are thinner and transmit more light. Photometric filters are not cheap, so buy what you actually need.[4] If you use the manual filter sliders, have the

3 If you have enough aperture.

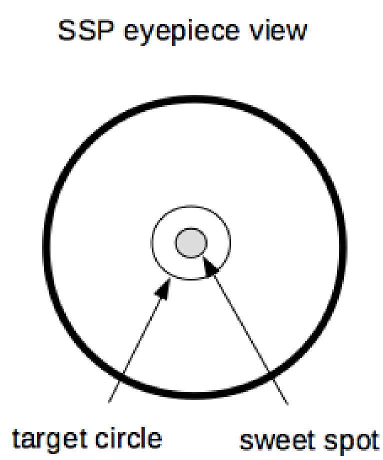
4 The standard SSP5 has zero response in R and I bands, but the extended-sensitivity version will work in R. There are caveats regarding the R and U filters (see section 3.5 on Transformation).

filters mounted in the color pairs you will need. As of this writing, filter sliders are still available from Optec and I think you can send them filters for installation. If you are going to do both BV and VI photometry projects, get two sliders and put BV in one and VI on the other. Old sliders will have B in the right position, V in the left, so that when the slider is pushed all the way in, the B filter is in the optical path. Optec has since reversed this convention. If you prefer to have your short-wavelength filter on the right, as I do, you need to specify that when ordering.

Don't clean your filters (or telescope objective) unless you need to. Your system may need re-calibration after such maintenance. There are four screws that adjust the X-Y position of the sensor, and one or two screws that lock the eyepiece in place. Don't fool with those, or your target reticle may not be centered properly.

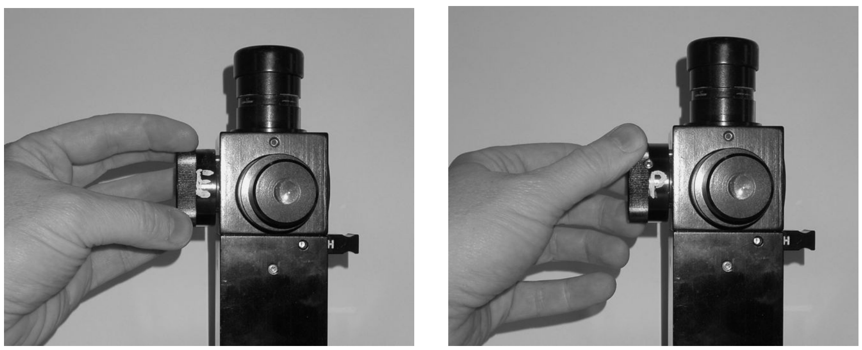
1.3 Basic operation

The SSP devices have a flip mirror. The mirror directs light to the eyepiece, or flips out of the way so light falls on the sensor. You start with the eyepiece, slewing the scope to put your star in the center of the target circle. The eyepiece has a one inch focal length, so with an 8" F/10 SCT you are working at a magnification of 80x. At this power, the field of view is about 24 arcmin in diameter, and the target circle perhaps 1.5 arcmin. The detector sweet spot is the central 35% of the target (by radius). At the rim of the circle, sensitivity may fall to 0. As you begin to experiment with centering a star, you will notice that your left/right/up/down eye position makes a difference, shifting the apparent location of the star in the circle. You'll get used to this, and gradually learn to keep proper eye alignment (keep the whole field stop of the eyepiece in view, if possible). Though I am nearsighted, I don't wear my glasses when making observations. I usually focus the photometer with my glasses on, and then remove them, improving the eye relief.



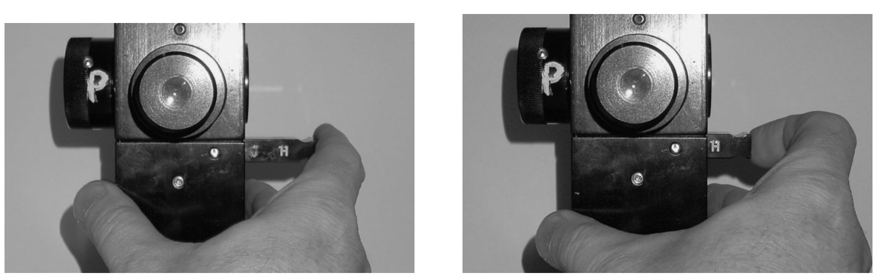
Once the star is well-centered, the flip mirror knob is turned (I have put white letters "E" and "P" ["eyepiece," "photometer"] on the knob so it is easy to tell the position in the dark). A series of three integrations can now be taken, but there is a catch: the integration timer is not synchronized to the mirror flip. Integration is actually happening all the time. Every ten seconds the counter is reset, regardless of the position of the mirror. Therefore, the first count you get after flipping the mirror is almost certainly not a full integration, and must be discarded (while an integration is taking place, the display will show the results of the prior integration). Generation 1 photometers have an LED to the right of the display. Except in early models, when the display updates, the light flashes, which is handy if the current and prior counts are the same.

Unless conditions are exceptional, the values you get for the three integrations will vary somewhat. I like to see the highest reading no more than about 1% above the lowest (assuming modest "dark" counts[5]). If the numbers are dancing around, you have bad sky conditions or bad tracking, the latter indicated by values going down, down, down. It should be obvious that you don't want to manhandle the flip knob: be gentle. Same with the filter slider:



Flipping the mirror: use the fingers, not the wrist

Move slider out: pull with thumb and forefinger, push with index finger



Move slider in: squeeze between thumb and forefinger

If your manipulations vibrate the scope a little, that's ok (let it settle), but a jolt will put you off target. As you

5 Dark counts are what you get with the mirror in the "eyepiece" position. They are set with an adjustment described below.

take your second or third integrations on a star, note if the counts are close to those of the first integration. A mismatch indicates a centering error or changing sky conditions. It cannot be overemphasized to pay attention to the counts and not simply log them. They are your key diagnostic for problems. If your counts are 10,000 or more, the aforementioned LED will come on and stay on.[6] On a Gen. 2 unit, the alphanumeric display will say "OVER."

Having got your star sample, flip the mirror and pan nearby for a sky sample. At a minimum, get the star outside of the target circle by one-half circle diameter. This is not always sufficient. If your star is very bright, scattered light in the optics may contaminate the field of view beyond the circle. If you suspect this, keep moving the scope further off the star until your counts minimize. Obviously, keep the circle away from any other stars, and don't forget to flip the mirror again afterward! You want counts from the sky, not the inside of the photometer case. If you can't seem to get consistent sky counts, there might be a dim star lurking below visibility in your background area—try checking a star chart.

Taking samples will seem awkward at first, but you will eventually develop a rhythm for the process. This is why I like my short-wavelength filter in the same position for all of my sliders—it's part of the pattern.

1.4 Setting the gain and integration time

The output of the SSP sensor is fed into an amplifier, which has three gain settings. If I am working with very bright stars, I use a gain of 1x, otherwise I use 10x. My feeling is that 100x is just magnifying the noise along with the star signal, and I don't use it. I see the gain setting as just a way to prevent count overflows, not a means to extract more information on dim stars. At 100x, my own SSP3 at room temperature has dark counts that change by fifty or even more in a ten-second integration. If you are only getting about 1000 net counts at that gain, internal noise is giving you a 5% variation in star readings before considering any sky effects. In principle, you could use different gain settings for the variable and the comparison to deal with a wide brightness range between the two stars. Be *very* careful about this. You cannot assume that the ratio between gain settings is exactly 10:1.

One second integrations are not useful on account of scintillation, an atmospheric effect that causes short-term fluctuations in brightness. Stick to ten seconds. If you have a Gen. 2 photometer, you can use five second integrations to prevent overflow, but take twice as many integrations so you still get a total of 30 seconds for a sample.

In preparation for taking data, there is an adjustment that must be made while the mirror is set for the eyepiece. The photometer has an "offset," which sets a floor for integration counts. The internal noise of the device generates some counts even with no light, and this "dark" count rate will vary with ambient temperature. I recommend a dark rate of about 400 per ten seconds for a Generation 1 photometer. There is a little hole to the right of the on/off switch with a screw. Turn the screw clockwise to reduce the dark counts, or counterclockwise to increase. In a Generation 2 photometer the offset is adjusted via the control buttons and the adjustment range is limited.[7]

6 See Appendix I for dealing with overflow counts.
7 See additional notes in Appendix FIXME

1.5 The standard sequence

We have already touched on the usual order for taking samples, with program the star bracketed by the comparison. The complete sequence, omitting sky samples, is as follows:

1. Comparison sample #1
2. Variable sample #1
3. Comparison #2
4. Variable #2
5. Comparison #3
6. Variable #3
7. Comparison #4
8. Check star
9. Comparison #5

The sampling data could be recorded in the following format:

Target	**Time**	**Integration 1**	**Integration 2**	**Integration 3**	**mean**
Comparison #1					
Comp Sky					
Variable #1					
Var Sky					
Comparison #2					
Comp Sky					
Variable #2					
Var Sky					
Comparison #3					
Comp Sky					
Variable #3					
Var Sky					
Comparison #4					
Comp Sky					
Check Star					
Check Sky					
Comparison #5					
Comp Sky					

The "check star" is a safety precaution. We only sample it once, so the measurement is not hugely reliable. However, if the observed check magnitude is seriously out of whack, we need to look for problems. The check is also useful for detecting variation in the assumed-constant comp star. Since each star sample is accompanied by a sky sample, a total of eighteen samples are taken. In a single color, this takes about twenty minutes. The integration means would be calculated the next day, not during data acquisition, but if you are keeping a paper record of the counts, it makes sense to put these means on the same sheet as the integrations. The sample start time need only be recorded to the minute, and it is not actually necessary to record the time of sky samples.

This sequence has proven effective, but it is not carved in stone. If you have two variables close together that use the same comp/check, you could do the following:

1. Comparison #1
2. Variable A #1
3. Variable B #1
4. Comparison #2
5. Variable A #2
6. Variable B #2
7. etc.

But don't push it too far. As the star samples get further apart in space and time, the reliability of the results can suffer. It should also be said that the three-integration pattern could be expanded to four or even five, but I wouldn't use two. Multiple integrations help smooth out variations caused by the atmosphere.

When you are doing two-color photometry, the star and sky must each be sampled with both filters. Rather than sampling star & sky in filter 1, then star & sky in filter 2, there is a more efficient pattern (here in B and V):

1. B band star
2. shift filter
3. V band star
4. move to sky
5. V band sky
6. B band sky
7. shift filter (for next star)

This involves only one star-to-sky motion per target.

1.6 Skies: the good, the bad, and the ugly

Our enemies in the pursuit of good photometry are wind, turbulence, heat, light pollution, aerosols, and water vapor. Wind shakes the equipment, turbulence and heat convection shift and distort the light, light pollution gives us photons we don't want, and aerosols and water absorb the photons we do want. It's a tough life, not even considering clouds (imaging observers can actually tolerate a bit of thin cloud cover, but we can't). If what you see in the eyepiece looks bad, wait and hope for improvement.

For the good results on dim stars, you need transparent skies. During the day, note how low-down the sky stays really blue. Look for jet traffic: if the contrails stretch from horizon to horizon, there's lots of water vapor in the sky. If you observe near a nighttime flight corridor, remember that those same contrails can float right in front of your stars.

All-in-all, watch the counts.

1.7 Monitoring

For a two-color observations, you nominally center a target star, flip the mirror, take a sample, flip the mirror again to check that the star is still centered, shift to the second filter, flip the mirror, take a sample, flip the mirror yet again to verify centering, shift to a sky position, flip the mirror, take the first sky sample, flip the mirror and check that you have not drifted onto a star, yadda, yadda, yadda. If your mount has excellent tracking there is no need to keep confirming the pointing, but *do* keep watching the counts. If your integration triples are changing significantly, that is a sign that pointing has gone awry, or clouds are moving through, or a satellite or meteor has lit up your sensor.

1.8 Extra notes on the SSP5

The SSP5 has a slightly different optical configuration. In front the PMT,[8] there is a “Fabry” lens that spreads the incoming light beam. As a consequence, the SSP5 does not have such a restricted sweet spot—it has full sensitivity over a wide area of the target circle.

Get in the habit of flipping the mirror to “eyepiece” before you move the telescope from one star to another: you might accidentally sweep across a bright star in between. You also might mistakenly command a slew that points at the moon or a terrestrial light. Yes, there is a safety circuit, but you don't want to power-cycle the photometer to reset it. Turn the photometer off right away when you finish observing, lest you turn on a bright light nearby as you close up shop for the night. And on the control panel of my SSP5, I put a big letter “E” to remind me to set the mirror to eyepiece before turning the unit on.

Optec used to sell two different V filters. The V filter for the SSP3 has always been a bandpass filter, allowing a window of transmission. There was once a second type of V filter used in SSP5s having the standard PMT. This PMT cannot detect anything redwards of V band, so the filter did not block the spectrum in that region. It was a lowpass filter that only cut off bluewards of the V band. You cannot use this filter in an SSP3! SSP5s with an extended-sensitivity PMT in my SSP5 must also use a bandpass V filter.

1.9 Our Mascot: Count von Count of Sesame Street

“The Count loves counting; he will count anything and everything, regardless of size, amount, or how much annoyance he is causing the other characters. For instance, he once prevented Ernie from answering a telephone

8 Photo-Multipiler vacuum Tube.

because he wanted to continue counting the number of rings...” - Wikipedia

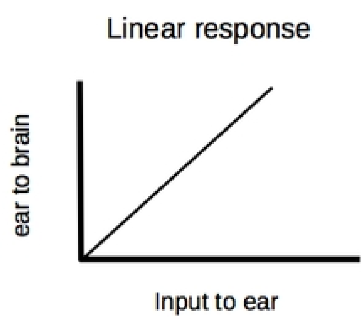
image: wikipedia.org

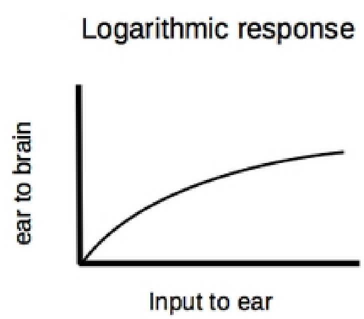
One...Two...Three...Ahahahaha!!

Chapter 2 — Technical details

2.1 The Magnitude System

We owe magnitude measures to the astronomers in ancient Greece. With their naked eyes, they divided stars into six numerical ranks of brightness, with magnitude one being brightest and six dimmest. This was the start of the trouble: as the stars got dimmer their numerical magnitudes got larger instead of smaller. The other problem was that the Greeks didn't understand how the eye responds to light of different intensities. They assumed the eye was *linear*, that when it told the brain that light A was twice as bright as light B, that meant that A really was twice the physical brightness. Not so. Human senses tend to be *logarithmic* or nearly so. Hearing works this way, and that's what allows us to distinguish such a huge range of sound intensity. As a sound level rises, the ear compresses the signal before feeding it to the brain. Below are diagrams illustrating the difference between linear response and logarithmic response.





The signal from a “linear” ear that could detect a cricket would blow the brain to bits if it heard an air horn. Logarithmic response lets the ear and brain get along over that wide range, and the eye/brain link works the same way. The Greeks thought of their magnitudes as six levels of linearly increasing brightness. If the brightness of a magnitude six star was *b*, then magnitude five brightness was *2b,* magnitude four was *3b,* and so on, the brightness increasing by the addition of *b* with each step. This meant stars of the top magnitude were six times brighter than those at the bottom (see table, below). But, in fact, magnitude one was 100 times brighter than magnitude six, and each step increased the brightness by a multiplication of about 2.512 ($2.512^5 = 100$). A big difference.

Magnitude —>	**6**	**5**	**4**	**3**	**2**	**1**
Linear-magnitude brightness	b	2·b	3·b	4·b	5·b	6·b
Logarithmic-magnitude brightness	b	2.51·b	6.31·b	15.85·b	39.82·b	100·b

Magnitude to brightness conversion

Actually, the Greek magnitude system did not exactly fit a 100x scale—what we use today is a modern refinement that also includes stars of zero and negative magnitudes. It uses fractional magnitudes and goes far fainter than human vision. The point to remember is that our photometers measure brightness, but we convert brightness to magnitude to do our data analysis.

Photometrists must work with three kinds of magnitude. The *instrumental* magnitude, “m”, is what we measure from the ground. This value is affected by absorption of light in the air. The *extra-atmospheric* or *extinction-corrected* magnitude, “m0”, is m with an adjustment for the estimated extinction (m0 is always brighter than m). Finally, there is *standard* magnitude, “M”, that is an adjusted extra-atmospheric magnitude. This refinement accounts for the non-uniform color sensitivity of different instruments. It can either dim or brighten the m0 value. If two observers properly color-calibrate their instruments, and properly estimate extinction during data collection, their standard magnitudes should be apples-to-apples comparable. [The magnitudes in star catalogs are standard. Unfortunately AAVSO CCD photometry defines standard magnitudes differently.]

Note: you will see the word “millimags” bandied about as a unit of measurement. A milli-magnitude is 0.001 magnitudes, a convenient unit for small values.

2.2 Time and Date

The civil calendar is not a very convenient time base for recording long-term astronomical data. It is divided into irregular months (with leap years thrown in) and there was a discontinuity when we switched from the Julian to Gregorian calendar. Instead, astronomers use *Julian Date* (JD) to mark time. JD 0 is the first of January, 4713 BCE, and the Julian days are numbered consecutively from then. As of this writing, seven decimal digits are needed to express a Julian Date (eg: 2457477). For convenience, we sometimes use *Reduced Julian Date* (RJD), that omits the first two digits of JD. There are no “Julian hours” or minutes: fractions of a JD are expressed as a decimal.

The Julian Day begins at the International Date Line, but in planning and recording our nightly observations, we use Universal Time (UT or UTC), which is referenced to the Greenwich meridian. UT is twelve hours behind “Julian time,” meaning that the Julian day advances at noon UT. There is a handy AAVSO utility for converting back and forth between Julian Day and civil day/time. For observers in the western hemisphere, JD typically stays the same during one night.[9]

2.3 Star Identifiers

Generally, the twenty-four brightest stars in a constellation are identified by Greek letters, with α the brightest and ω the dimmest. After that, they are designated by lower-case (a, b, c, ...z), then upper-case (A, B, C,…Q) Latin letters. This is the “Bayer” system. Variable stars within this range are known by the Bayer identifier. Post-Bayer variable star designations, which are not in order of brightness, begin with R, going to Z, then go to double-character designations, like SU. The two-letter identifications proceed in a strange pattern, which we need not go into here. Suffice it to say that all the letter combinations amount to 334 designations. Beyond this, the variables in a constellation are known as V335, V336, V337, etc. There are also “NSV” designations. NSV

9 There is also Heliocentric Julian Date (HJD) that is referenced to the position of the sun rather than the earth.

stands for New Catalog of Variable and Suspected Variable Stars, a list not ordered by constellation.[10]

Various star catalogs made over the years assigned only numbers to stars, and the prefix of the catalog is given with the number. The numbering system generally proceeds in order of Right Ascension. Below are catalogs you may encounter.

Catalog	Prefix
Henry Draper	HD
Hipparcos	HIP
Bright Star (Yale)	HR (or BS)
Smithsonian Astrophysical Observatory	SAO
Bonner Durchmusterung	BD

2.4 Photometric Bands

Your stereo system may have tone controls to boost or cut the treble, midrange, or bass portions of the audio. These controls are *filters*. Imagine what you could do with very extreme filters. If you were listening to an orchestra concert and cut the treble and mid-range deeply, you could, in principle, listen to just the notes from the double-bass players and exclude the other instruments. The filters allow you to select specific information from a broad spectrum of input. We use filters in photometry for just this reason: different bands of color provide unique data about what is going on in a star. No one has devised an optical filter that can boost a desired range of color—our filters only cut out what we prefer to ignore. Filters come in groups known as *systems*. The most common is the *Johnson* system, developed by Johnson and Morgan in the 1950s. The primary filters are U, B, and V. The U filter rejects visible light, permitting near-ultraviolet light to pass through. The B filter transmits blue light, and the V filter roughly passes the human "visual" color response in green. Johnson also defined an R filter for red light, and an I filter for red beyond human vision. There are many other color systems in use, but for AAVSO PEP we chiefly use Johnson in B and V, and the R and I filters defined in the *Cousins* system.[11] The color range that a filter lets through is known as its *passband*.

Instrumental, extra-atmospheric, and standard magnitudes in a particular filter band are denoted using the letter of the band. E.g.: v, v0, and V for the V filter; b, b0, and B for the B filter.

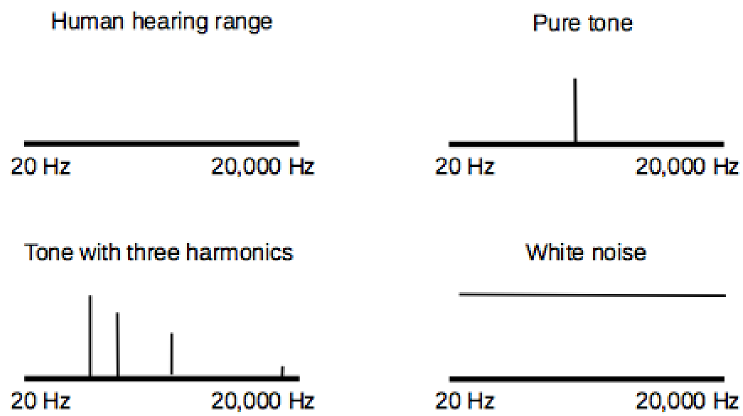
2.5 Response Curves

When I was growing up in the 1970s, quality stereo equipment was becoming readily available to the masses. A good amplifier might have a distortion specification of +/-3 decibels from 20 to 20,000 Hertz (Hz), which roughly

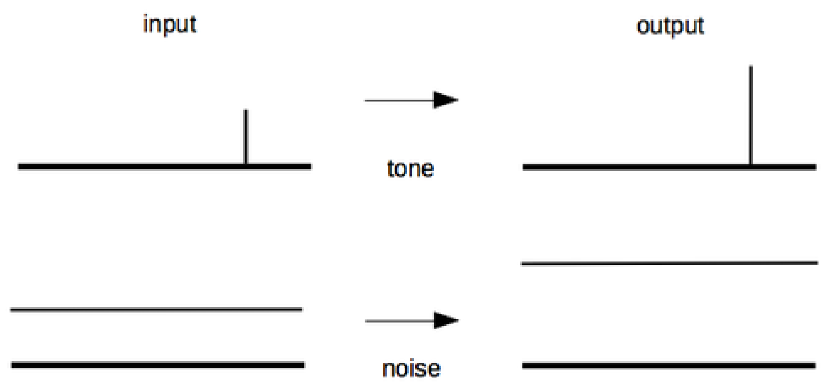
10 Beware of an ambiguity in certain star designations where the spelled-out Greek character matches a two letter variable prefix. The notorious example is μ Cep. If one queries online programs for "mu Cep," one gets results for MU Cep (both are variables). The same problem exists for η (nu). The usual disambiguation is to append a period after mu or nu ("mu." and "nu.") to indicate the Greek character. In the database of PEP stars, an alternate solution was used: insertion of 'i' in the prefix ("miu" and "niu").

11 The SSP3 cannot realistically make use of U band.

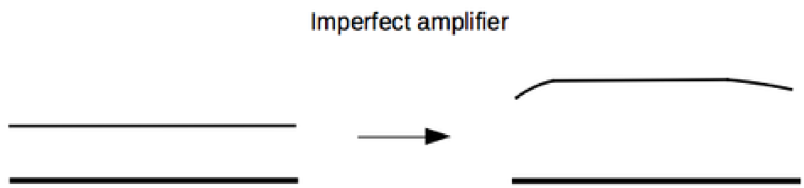
covers the range of human hearing. To visualize audio, we use spectrum diagrams, which show the intensity of sound at each frequency (below). A pure electronic tone consists of a single spike. A musical instrument like the flute produces a fundamental tone plus harmonics, or overtones, of decreasing intensity. An orchestra in action would have a vast forest of tones. At the extreme, white noise consists of sound at every frequency.



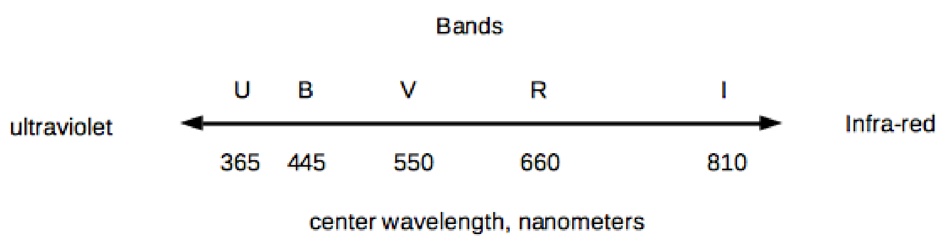
An ideal amplifier will take the input sounds and do nothing but magnify them uniformly from 20 through 20,000 Hz. In other words, the output spectrum is identical to the input spectrum, only at a higher intensity.



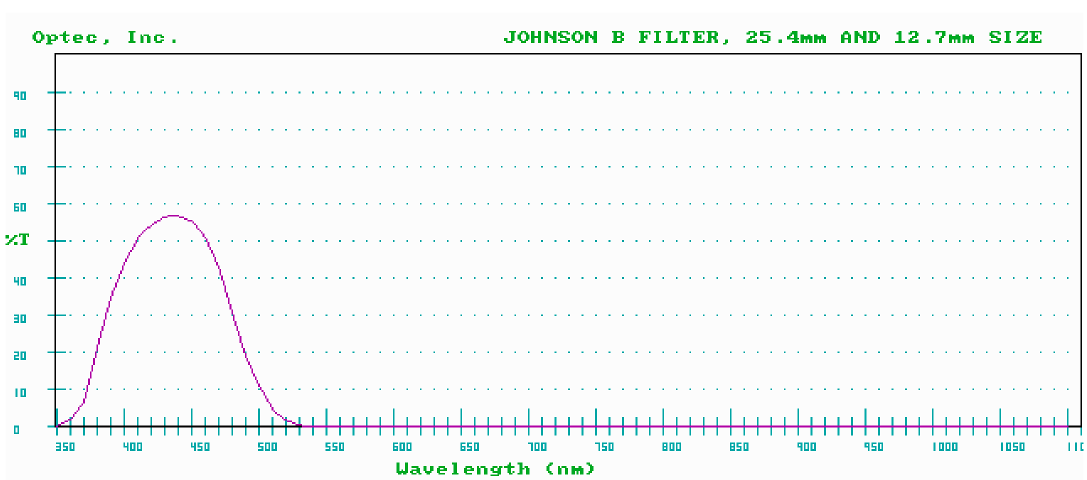
The ideal amplifier has a *flat response* over its operating range, as you can see by comparing the white noise inputs and outputs. Of course, no amplifier is perfect. It may work very well in its frequency mid-range, but suffer at the extremes. Below, our bass and treble lose a bit.



In optical systems, we have similar concerns about response, though we deal in attenuation, not amplification. Any spectrum can be characterized in terms of frequency, as above, or in wavelength. Optical frequencies are huge, inconvenient to describe in Hertz units. Instead, we use wavelengths, usually measured in nanometers (nm).

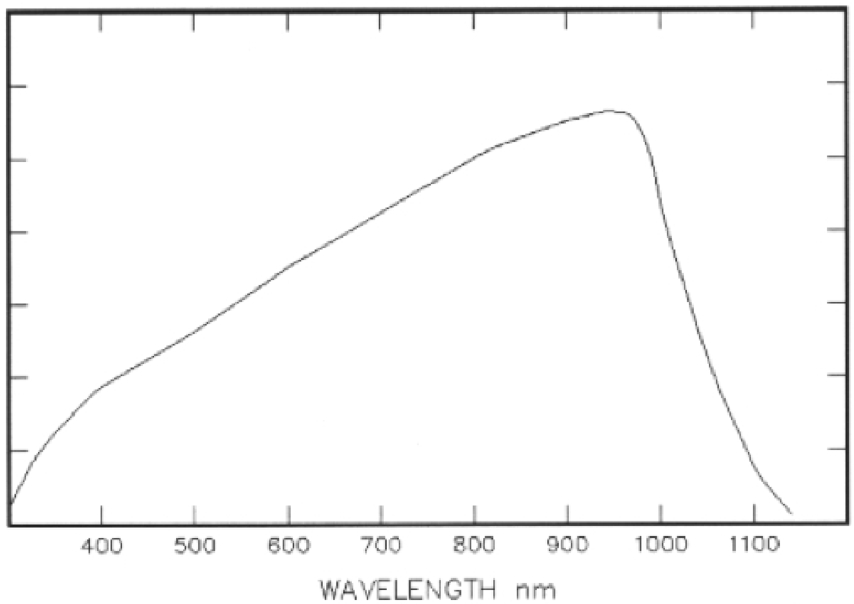


One might hope that the B, V, R, and I filters would exhibit flat response within their passbands, but they do not. Below is the response curve for the "old" B filter from Optec. Not only does the filter fail to sharply chop off at each side, its peak efficiency is only about 55%. This is not a criticism—Johnson's filters were not perfect either.



B filter transmission © Optec corp.

Filtering is a very tricky business, even in audio, and flat response curves are not to be had. But the situation is worse than it looks. Our telescope, filter, and sensor form a chain, and each component has its own response curve. The telescope curve is very nearly flat, but not perfectly so. It will transmit different wavelengths of light with slightly different efficiencies. The sensor in our photometer also lacks a flat response, being more sensitive at some wavelengths than others. If we think of these pieces as filters, that each transmit different fractions of the incoming light, we get a total response of scope% · filter% · sensor% at every wavelength.



SSP3 sensor response curve © Optec corp.

We needn't dwell on the details of all this, but remember that the sensitivity of our measurements, in any band, is affected by the combined efficiency of the whole photometric chain.

2.6 Single Channel Photometry

A photometer is nothing but a scientific-grade light meter. At the moment, the PEP group is using Optec SSP photometers almost exclusively. These are what are known as Direct Current, or DC photometers. The other category of photometer is the pulse-counting, or photon-counting type. The difference is this: when a photon arrives at a "counting" style photometer, the sensor puts out an electronic pulse. This pulse goes to a counter, which is allowed to accumulate over a period known as the *integration time*. The final count reflects the number of photons. In a DC photometer, the arriving photons are not registered individually. They produce a continuous current that is proportional to the number of photons. In the olden days, this current was fed into a chart recorder. The chart pen would deflect according to the strength of the current, and a measurement was called a *deflection*. The SSP photometers use additional electronics to turn the current into counts, but these are not photon counts. The term "deflection" still gets used by some writers, but it becomes confusing. I will banish it from this note and introduce *sample* in its place. A sample is a set of consecutive ten-second integrations (usually three), averaged to produce one value.

While a camera has millions of pixels, a single channel photometer has only one honking-big pixel. This pixel is much larger than the size of a star image. As a consequence, when we aim the photometer at a star, it also sees a bit of sky around the star. This presents a complication, because the sky is not perfectly dark. Earth's atmosphere glows, even on the darkest night. Furthermore, stars that are too dim to see can contribute light within the field of view of the sensor. We call all this light *sky background*, and it contaminates the measurement we make of the target star. To correct for it, every sample on the star is accompanied by a second sample on the sky near the star. When we report counts for the star, we subtract these *sky counts* from the *star+sky counts* to get a *net count*. This procedure is not perfect, but, with care, it works well.

Like most AAVSO observers, we in PEP practice the art of *differential photometry*. That is, we establish the magnitude of a variable star by comparing it against a (hopefully) constant star having a (hopefully) reliably published magnitude. Variable stars in the PEP program each have an assigned “comparison” star. Use of the same comparison improves the internal consistency among multiple observers. It should be noted, here, that photometry can be a squirrelly business—studying starlight through the atmosphere poses inevitable problems. Every measurement is subject to perturbations, and truly good reference magnitudes are established by expert observers averaging many observations.

The typical PEP observing sequence interleaves samples of the variable (or “program”) star with samples of the comparison: comp...var...comp...var...comp...var...comp, the telescope being moved back and forth. Each variable sample is referenced against the average of the two comparison samples that bracket it. Having three samples of the variable not only gives us a more reliable result than a single sample, it allows us to compute a statistical error, or uncertainty, for the observation sequence as a whole.

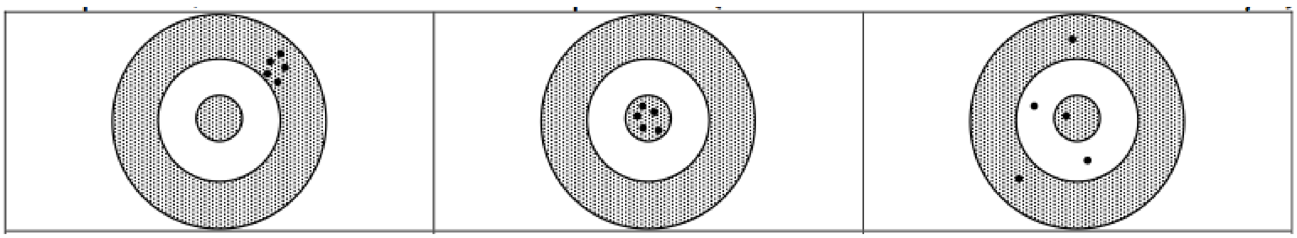
As noted above, our DC photometers produce counts that are proportional to the number of photons received during an integration. If p photons arrive, we will have a count of $k{\cdot}p$, where k is a constant for our photometer. The complete reduction of counts to magnitude will be left to another section, but suffice it to say that we will take the logarithm of the counts, so that the magnitude will be of the form $\log(k{\cdot}p)$. If we get p_v photons from the variable and p_c photons from the comparison, then the magnitude difference, ΔM, will be of the form $\log(k{\cdot}p_v) - \log(k{\cdot}p_c)$. If we have a reliable magnitude M_c for the comparison, then the magnitude of the variable will be $M_c + \Delta M$. That is fine for me and my photometer, but what about you? Your photometer will have a slightly different value of k. Doesn't that mean our instruments operate on different “scales,” like Fahrenheit and Celsius? How can we reconcile our results? The mathematics of logarithms comes to the rescue: $\log(k{\cdot}p)$ can be re-written as $\log(k) + \log(p)$. This means that the differential magnitude formula can be simplified as follows:

$$\begin{aligned}\Delta M &= \log(k{\cdot}p_v) - \log(k{\cdot}p_c) \quad \text{becomes}\\ &= (\log(k) + \log(p_v)) - (\log(k) + \log(p_c)) \quad \text{[expanding the logarithms]}\\ &= (\log(k) - \log(k)) + (\log(p_v) - \log(p_c)) \quad \text{[re-arranging terms]}\\ &= \log(p_v) - \log(p_c) \quad \text{[canceling terms]}\end{aligned}$$

My differential magnitude is independent of k, and so is yours. We can compare them directly. This independence applies to all multiplicative factors that affect our respective counts. If your scope aperture is bigger than mine, your photon counts will be higher by a factor. If your filter has a 10% higher transmission efficiency, your counts will be higher. Likewise if your sensor has 5% greater sensitivity, or your integration timer runs 3% slower. All these considerations drop away when we compare stars differentially on a logarithmic scale.

2.7 The Sinkhole

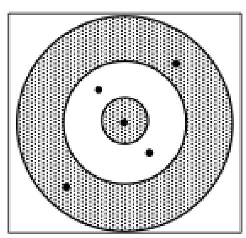
Much photometric ink has been spilled on the distinction between *accuracy* and *precision,* the main source of trouble being that the former term has an ingrained colloquial meaning that is different from its technical usage. There is also the question as to whether you are discussing a single measurement, or a group of measurements. The latter situation is usually illustrated with the infamous “target diagrams:”



precise, not accurate — precise and accurate — imprecise and inaccurate

images: haystack.mit.edu

On the left, we have an archer who is precise, but not very accurate. His arrows landed in a tight group, but they missed the center by a wide margin. Moving right, we have an archer who is both precise and accurate, with a tight group on the bullseye. Rightmost, we have an archer who is neither precise nor accurate. Below, we have an archer who might be described as accurate but not precise:



the average is accurate

He certainly is not precise, but one can argue that his *average* has high accuracy, in that the mean (x,y) location of his five arrows is right on the money. If his objective was to *locate* the center of the target by the average of his shots, he did very well. However, he still would get a low score in the competition.

Another perspective on accuracy and precision was offered by Arne Henden:

> *It is pretty safe to say that the average CCD observer has very good precision, but pretty poor accuracy. What this means is that the uncertainty from point-to-point in, say, a time series, is excellent. That is why so many observers are able to detect an exoplanet transit (millimagnitude depths) … where the peak-to-peak amplitude may only be a few hundredths of a magnitude. Compare one observer to another observer for the same object and the same night, and you might see far larger separation between the mean levels of the two time series—the "accuracy" part.*

There is a useful distinction in this description: precision defined in terms of the *uncertainty* of the measurements. Every measurement, even a digital one, has some level of uncertainty. So does an averaged group of measurements. Our fourth archer, from a measurement perspective, had high accuracy but also high uncertainty. The second archer had high accuracy and low uncertainty.

In some quarters, there is an effort to sidestep problems with the word "accuracy" by substituting a new word, *trueness*. Trueness is the metric for how close the measured value is to the True value. In the PEP vision I proposed, I deliberately stayed away from both accuracy and precision so as not to drag readers into the sinkhole too early. The vision may now be rephrased as *Highly-true, minimally-uncertain photometry of bright, astrophysically interesting stars.* This is what we mean by "high quality." A practical upshot is that if you and I take photometry of the same target at the same time, our results ought to agree, within our mutual uncertainties, and not because our uncertainties are large.

Chapter 3 — Data Reduction

3.1 Software

I am aware of four different approaches for reducing AAVSO PEP data:

1. AAVSO PEPObs (V band only)
2. SSPDataQ from Jerry Persha
3. PEP spreadsheets
4. Homebrew programs

Most people seem to prefer spreadsheets, and we now have a version for BVRI reductions. I hate spreadsheets and wanted elaborate reduction capabilities, so I wrote my own program. PEPObs has the advantage that it is available as an Internet web page, and so requires no special software on the user's computer (Appendix A). SSPDataQ is a Windows program for Generation 2 photometers that communicates directly with the SSP, but you must use at least two filter bands. Choose the reduction package that is most convenient for you. Just be aware that different tools will give slightly different answers.[12] If I must make exacting comparisons among the results from different observers, I obtain the raw data and re-reduce them all through a single tool. In section 3.5, I go into the details of reduction, which are good to know, even if you don't write your own tool.

3.2 Data management

If you have a productive career as a photometrist, you will end up collecting a lot of observations. Early on, it is important to develop a strategy to keep your data organized. If you store data on a computer, don't fall into the trap of giving all your files cryptic names and dumping them into a single directory. There are various ways to structure your personal archive. At the top level, you might break it up by calendar year, or perhaps by variable type (eclipsers, pulsators, etc.), or maybe reverse those two levels of stratification. I presently have top-level directories for each star. My filenames for individual observations are of the form <star_id>_<telescope>_<RJD>_<bands>, where RJD has an integer and fractional part, eg: p_cyg_TC9_57424.810_BV. I observe with more than one telescope, hence, its inclusion in the file name.

3.3 Observational honesty

We want the data we report to be free of "opinions"—judgment calls by the observer. Human expectations are not always in line with reality, and our magnitudes are supposed to represent reality. Human factors *do* creep in, however, and we must manage them responsibly. For instance, what if I complete a standard sequence on a star, only to find that I forgot a sky sample along the way? Do I just throw away my data, or try to estimate the

---

12 You may hear statements like, "This tool uses the method of chapter 4 in Henden and Kaitchuck." That characterization is fine so far as it goes, but keep in mind that chapter 4 of H&K is not a software functional specification—it does not define an algorithm. Various programmers can read chapter 4 and come up with different implementations. Also, authors may use different approximations in their calculations.

background counts? If my background has been consistent over the other samples, I have no problem using an average background in place of the missing one, but I would put a note with the observation that there was an estimate involved. What if a hot integration slipped through when I was not looking, only to be found during reduction? If integrations have been stable, I would drop the offender and make a note in the observation record.

A consideration to make when making a call about a reduction: will the magnitude need to stand on its own, or is it to be evaluated as part of a larger group? For instance, if we are fitting a line to several magnitude readings, we need not be too worried about a small bias in one of them.

The questions around ratty data can become a philosophical conundrum. If the sky looks bad and I choose not to observe, there is nothing to report. But if I go ahead and try to collect data, am I right to exclude them from some larger analysis because they are of poor quality? In principle, it seems I should report everything, but crummy data may only muddy the story. One needs to be careful about excluding poor results just because they don't agree with expectations.

3.4 Avoiding embarrassment

When you submit data to AAVSO, or any other organization, look at your numbers. *Do they make sense?* Some hilariously bad results get reported when people don't perform simple quality control. Just because the computer tells you a value doesn't make it right. If you find problems with your reported observations, delete them and upload fixed versions.

3.5 Gory details of reduction

3.5.1 Instrumental magnitudes

Having acquired counts from our photometer, what is the conversion to magnitude? We already know that it involves taking a logarithm, and since magnitudes go in the opposite direction of brightness, we must introduce a negative factor somewhere. Further, we want the magnitude to decrease by 5 for a 100x increase in brightness. The formula, then, is

$$m = -2.5 \cdot \log_{10}(\text{counts})$$

This is equivalent to $-5/2 \cdot \log_{10}(\text{counts})$, or $\log_{10}(\text{counts}^{-5/2})$. Let's verify the formula: call the instrumental magnitude of some star $m_1 = \log(c^{-5/2})$. Consider what happens when c increases by a factor of 100. The new magnitude, m_2 =

$$\log((100c)^{-5/2}) = \log(100^{-5/2} \cdot c^{-5/2}) = \log((10^2)^{-5/2} \cdot c^{-5/2}) = \log(10^{-5} \cdot c^{-5/2}) = \log(10^{-5}) + \log(c^{-5/2}) = -5 + \log(c^{-5/2}) = -5 + m_1$$

Our star is five magnitudes brighter. Q.E.D. Remember that the 2.5 factor is actually 2.5, *not* 2.512 rounded down.

Because of the properties of logarithms, the differential magnitude between the variable and comparison can be expressed two ways:

$$\Delta m = m_{var} - m_{comp} = -2.5 \cdot \log(counts_{var}/counts_{comp})$$

3.5.2 First-order extinction

First-order extinction is the simplest correction applied in the reduction process. The Earth's atmosphere attenuates starlight, and the more atmosphere through which the light passes, the more attenuation. When collecting data, the variable and comparison stars, though typically close together, are at slightly different altitudes. We compensate for the extinction difference. The quantity of atmosphere is measured in "airmass" units, and the symbol for the value is "X." Straight overhead is an airmass of one. At thirty degrees elevation, the airmass is two, and it quickly rises as you go lower. To compute the correction, you need two numbers: the differential airmass between variable and comparison (ΔX = var airmass - comp airmass), and the extinction coefficient, kappa (k')[13], in units of magnitudes per unit airmass. Note that ΔX can be positive or negative. The differential extinction is $\Delta X \cdot k'$, and this value is subtracted from the instrumental magnitude.

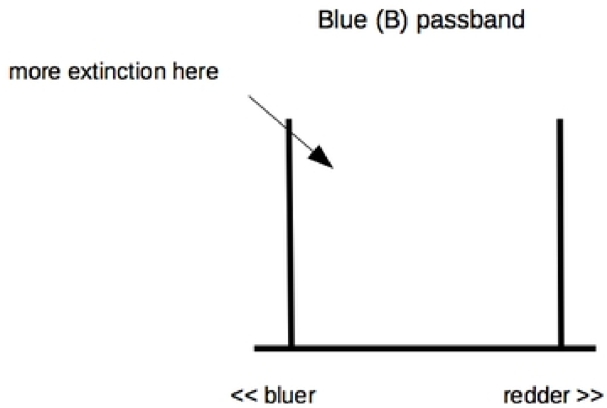
3.5.3 Color contrast

Before proceeding further, we must introduce the concept of color contrast, based on *color index*. A star's color index is the difference in standard magnitudes in two passbands. The most common index is B-V, the B magnitude minus the V magnitude. A reddish star will have a bright V magnitude (relative to B), and B-V will be positive. A bluish star will have a bright B magnitude, and B-V will be negative. The difference in indexes, $\Delta(B-V) = (B-V)_{var} - (B-V)_{comp}$, gives the color contrast between variable and comparison. Values of Δ(B-V) near zero indicate stars with very similar color. A positive Δ(B-V) indicates the variable is reddish relative to the comparison. Color contrast can be expressed both in terms of standard magnitudes, B&V, or instrumental magnitudes, b&v. These two contrast values are usually very close, but they are not the same, and they have different uses.

3.5.4 Second-order extinction in B band

In the blue part of the spectrum, extinction increases rapidly as the wavelength shortens. In B, we see an effect caused by measurably different levels of extinction within the passband. At the short-wavelength end, light will experience more attenuation than at the long-wavelength end.

13 Well, kappa *prime*. Extinction, k, is formally divided into a color-insensitive part, k', and a color-sensitive part, k'', with the total extinction $k = k' + k''$. For first-order extinction, k'' is zero.



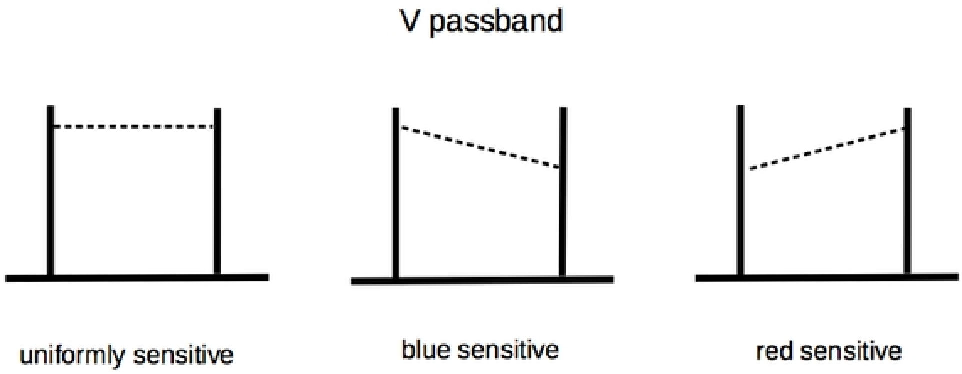
This means that at a given altitude, a star with an excess of blue light will suffer more extinction than a star with an excess of red light. If our variable and comparison have different color indexes, this will affect our results. Second-order extinction is quantified as $k''_B \cdot X \cdot (b-v)$. The differential correction can be approximated as $k''_B \cdot X_{mean} \cdot \Delta(b-v)$,[14] where the first term is the second-order extinction coefficient, and the second is the average of the variable and comp airmasses. This extinction is subtracted from the instrumental magnitude. Here, we use the instrumental color contrast, rather than the standard contrast. This is because the process for measuring *k*" takes place in instrumental magnitudes.

Common k''_B values are in the range -0.02 to -0.04, according to the books, but I have them as high as seen -0.1 in actual practice. Second order extinction can be substantial. For example, if $k''_B = -0.03$, $X_{mean} = 1.5$, and $\Delta(b-v) = 0.500$, the correction will be +0.022. Second order extinction in V, R, and I is negligible, and U band is its own strange world.

3.5.6 Transformation

No two combinations of scope/filter/sensor are identical. In particular, every system has different sensitivity to color across the spectrum of a given passband. Hence, your instrumental results for a star will differ from those of everyone else. As an example, consider the (exaggerated) spectral sensitivities illustrated below. At left is a system with uniform spectral sensitivity; at center, a system with more sensitivity at the blue end of the V passband; at right, a system with more sensitivity in the red. In general, measurements by these systems will not agree. Transformation adjusts your instrument to the "standard system," whose data points are comparable for all observers.

14 Assumes the two stars are fairly close together.



To effect transformation in V and B bands, we need coefficients ε_v and ε_b, known as "epsilon-V" and "epsilon-B".[15] The transformation adjustment is epsilon · Δ(B-V), a value added to the instrumental magnitude. For an SSP3, ε_V tends to be around -0.05 (though I have seen much lower). Thus, for Δ(B-V)= 0.500, the transformation adjustment is -0.025.

To think about this, imagine that I have the blue-sensitive system. First, consider that my variable and comp have the same B-V. This means that they are equally affected by my non-uniform color response, so a differential comparison of the two will be unaffected (the V transformation is $\varepsilon_v \cdot 0$). Now consider my variable to be bluer than the comparison, so that Δ(B-V) < 0. My comp has a comparative excess of red, and that red light will suffer in my response curve, making the comp appear dim relative to the variable. If my comp is dim, that makes the variable appear brighter than it really is. With the ε_V and color contrast values both less than zero, the transformation value added to the instrumental magnitude will be positive, making the standard magnitude of the variable dimmer.

HERE Note that the "old" Optec R filter does not transform well to the standard system, and the U filter will not transform at all.

3.5.7 Complete magnitude reduction formulae

We can now state the formula for converting differential instrumental magnitudes to differential standard magnitudes.

Δ standard magnitude = Δ instrumental magnitude - extinction(s) + transformation

$$\Delta V = \Delta v - k'_V \cdot \Delta X + \varepsilon_V \cdot \Delta(B\text{-}V)$$

$$\Delta B = \Delta b - k'_B \cdot \Delta X - k''_B \cdot X_{mean} \cdot \Delta(b\text{-}v) + \varepsilon_B \cdot \Delta(B\text{-}V)$$

Adding the standard magnitude of the comparison star to the Δ standard magnitude gives us the final standard magnitude of the variable.

15 I am going to skip the "mu" coefficient (μ) used to transform the B-V color index.

3.5.8 Airmass computation

To a first approximation, the airmass of a star at altitude *a* above the horizon is 1/sin(*a*) or cosecant(*a*). A gory algorithm for computing airmass from RA/Dec and your location and time is given in Appendix G.[16]

3.5.9 Time of observation

When we report a time for the whole observation, which took place over many minutes, we typically choose the time of the second variable sample. This is the "middle" of the sequence (not counting the check star). Alternately, one could use the average times of the first and fourth comparison samples. Given our low time-resolution, the exact value is not a big deal. However, you should record your individual sample times at consistent points: e.g.: at the beginning of the first integration or end of the last integration.

3.5.10 Metadata (data about the data)

How do metadata affect the reduction of your results? They don't, but they are important when someone comes along later and tries to evaluate your data. Metadata also help observers detect problems in their own reductions (*Oops, wrong extinction coefficient…*). Observation records in the AAVSO archive contain fields for only a limited amount of metadata, so we must encode any additional information in the "notes" section. The recommended format is <keyword>=<value>, with such pairs separated by the Unix pipe character, '|'. We avoid using apostrophes and quotation marks, which make for complications when parsing the comments with shellscripts. The challenge with metadata is to include enough without going overboard. Appendix C has the PEP metadata definitions.

3.5.11 Reference magnitudes

If you observe the PEP program stars, the comp and check stars are specified in a file on our web site. Reference V magnitudes for the comparison and check stars are given in the file, and B-V are given for both the variable and comp. If you expand to other filter bands, or other stars, you need a reliable source for magnitudes. In the case of B band, we can compute the comp B magnitude from the information in the database. We are given V and B-V, hence, the B magnitude is (B-V) + V.

Beyond B band, we must look elsewhere for reference data. It should be pointed out that there is no documentation for where the database magnitudes actually came from. When we bring more star magnitudes into the PEP ecosystem, we should be careful about their origins. A convenient source of magnitudes is the SIMBAD website, but I only use it for casual inquiries. A better choice is the General Catalog of Photometric Data, from which most PEP program magnitudes seem to have been drawn. The GCPD contents have been submitted to a vetting process that seems reasonably thorough and consistent. A drawback is that the index only works with HD star designations, and not HR or SAO numbers, which are common in the PEP database. I think we should avoid introducing any new HR/SAO identifiers into the mix.

16 If you ever run across the term *zenith angle*, that is the star's angular distance from the zenith, not its altitude.

As a general rule, we want to draw magnitudes in all bands from the same source for a given comparison, and we want the color index, whether B-V or V-I, etc., of the comparison to be close to that of the variable, so as to minimize transformation problems. If you use a comparison magnitude that is not traceable to a chart ID, you *must* include that magnitude the metadata.

As regards star coordinates, I would like to propose that we limit precision of right ascension to the tenth of a second, and declination to the arcsecond (eg: 13h 42m 20.5s, 33d 15m 42s). Extra digits, though rife in the PEP database, are of no practical value.

3.5.12 AAVSO extended file format

If you are reducing your own data, you will need to produce an AAVSO-standard text file that can be uploaded via WebObs. The format definition is available on the AAVSO website. Read the file definition carefully. The comparison and check star magnitudes are *instrumental*, not standard. If you are using reference magnitudes from the PEP database, the chart will be “PEP.” The definition for TRANS is out-of-date (we don't use Landolt fields). Put “YES” in the transform field if you transform.

Chapter 4 — A Quick Digression on Statistics

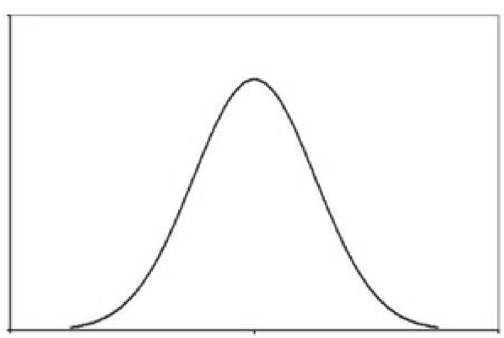
4.1 Precision

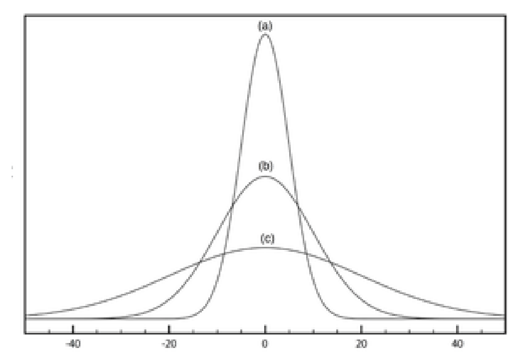
No measurement is exact. The joke about astronomy is that observations that agree within a factor of two of theory are doing well. In photometry, we can do better than that, but we must work at it. Measurements are affected by problems both random and systematic. We call these problems *errors*. A systematic error is introduced when:

- We have dew on the optics
- Our tracking slips
- Counts are recorded incorrectly
- We use the wrong comparison star
- Time or date are noted wrong

and a dozen other things. We strive to eliminate systematic errors by good habits and operating the equipment alertly. Random errors cannot be eliminated, but statistical techniques let us manage them.

When we thrice measure the magnitude of a star, we have *sampled* its magnitude three times. What is the character of this sample? We model our measurements as a normal, or Gaussian distribution around the "true" magnitude of the star. The normal distribution is illustrated below:





images: cpp.edu chemwiki.ucdvis.edu

If the true measurement would be at the center, we can expect actual measurements to be distributed around it in proportion to the height of the curve (left). Normal distributions can have different levels of scatter (right). A normal distribution is characterized by its *standard of deviation,* σ ("sigma"). In the right-hand diagram, the tall curve has a small σ, whereas the squat curve has a large σ. If we were making a distribution curve for the heights of a collection of 100 men—a distribution expected to be Gaussian—we could measure all the individuals and compute the σ of that group as:

THIS SECTION IS UNDER REVIEW

$$\sigma = \sqrt{\frac{\sum_n (h_i - h_{mean})^2}{n}}$$

Where h_{mean} is the average height, and the h_i values are heights of individuals

We could also *estimate* σ by measuring only some of the men. In this case, the formula becomes:

$$\sigma_{est} = \sqrt{\frac{\sum_k (h_i - h_{mean})^2}{k-1}}$$

Where $k<n$, and h_{mean} is the mean of only k measurements

We divide by k-1 because the estimate based on k is likely too small. This is known as the *unbiased* sample standard deviation.[17] The precision of our subset is given by dividing the estimated σ by the square root of the number of samples. It is the *standard deviation of the (sample) mean* (SDOM).

$$\sigma_{mean} = \sqrt{\frac{\sum_n (h_i - h_{mean})^2}{n \cdot (n-1)}}$$

This formula is closely related to that for *standard error*, and as n gets large the two converge.[18] This is the error, or uncertainty, that we report with our observations. We expect a 68% chance that the true magnitude is within +/- SDOM of our measured value. This "one-σ" estimate is, thus, not very good. If we double the SDOM, we have a two σ-estimate that has a 95% chance of success. In the interest of full disclosure, the normal distribution is just a model for our measurements. It only truly applies if they are fully independent, and ours are not. Why? Each differential magnitude is based upon two comparison star samples. The "after" comparison sample for the first variable sample is reused as the "before" sample to compute the second variable sample. Also, any statistic computed on just three points cannot be tremendously robust.[19]

17 The biased standard deviation uses k as the divisor.
18 The standard error formula uses n^2 as a divisor. For our purposes, it significantly underestimates the error.
19 Though AAVSO PEP has long used SDOM there is work afoot to see if it is really justified. This could be changed.

Chapter 5 — Calibration

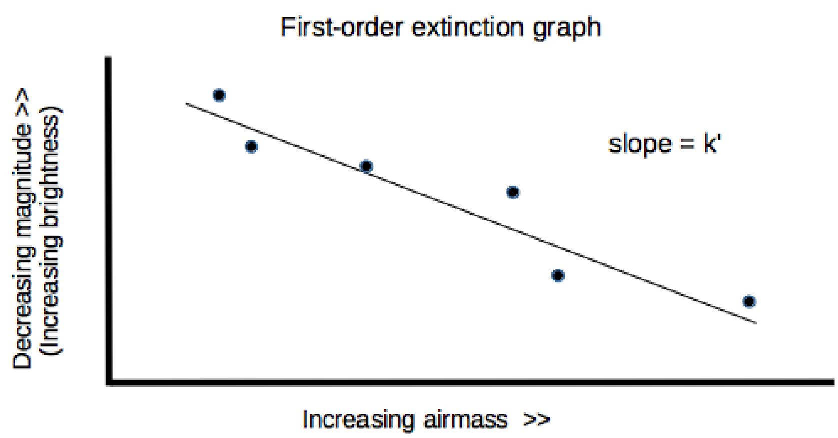
Having introduced the factors needed to reduce our data, we can go into detail about calibration observations. These are performed to establish coefficients for first-order extinction, second-order extinction, and transformation.

5.1 First-order extinction

There are three sources for k':

1. Follow the instrumental magnitude of one star over a range of airmass during the night.
2. For several standard stars at different airmasses, compare the instrumental magnitudes against their standard magnitudes ("Hardie method"). This is done in a brief period.
3. Assume a fixed or seasonal value.[20]

For methods 1 and 2, you are creating a graph of the sort below.



If we will be out for an extended observing run, the single-star/large-range method may be convenient. First thing in the evening, you would take samples on a star that is either high in the sky, or low in the east. Over the course of the night, sample the star as it changes altitude, and once more before you finish. If you will be observing only part of the night, be sure the star you choose for extinction will cover a reasonable range of altitude. Ideally, you want samples—I like to have at least five—over an even distribution of airmasses. This means that you sample your extinction star more often when it is lower. A simple approximation for airmass is 1/sin(altitude). Henden & Kaitchuck (see Appendix B) provides lists of standard extinction stars for northern and southern hemisphere observers. These are fairly bright stars with a B-V color index close to 0. To reduce the extinction measurements, you plot the instrumental magnitudes against airmass, and fit a line to the points, the

20 A common value for sea level extinction is 0.25. B band tends to be about 0.13 higher.

slope of the line being *k*'. Because magnitudes *decrease* as brightness increases, the slope of the above line, in magnitudes per unit airmass, is actually positive.

The disadvantage of the above method is that one needs to be taking data for an extended period, and for the extinction to stay fairly constant during that time. For some observing sites, the latter constraint is a significant problem. Most observers who face this difficulty compute a coefficient separately for every star they observe, based upon the change in extinction of the comp star during the standard sequence. I don't like this method. The twenty or thirty minute duration of the sequence will not place the comp at a significant range of airmasses, leading to noisy results. One can argue that when the star is high in the sky, the differential extinction will be quite small, so the noise is unimportant, and when the star is lower, the airmass range will improve and the noise will go down. I'm still not sure that this approach is any better than just randomly picking an extinction correction between 0 and 4 millimags (in V band), or, with more sophistication, selecting a value from 0 to 4 based upon the amount of differential airmass. In any event, the comparison star almost certainly has a nonzero B-V, introducing the possibility of a skewed estimate of the first-order B extinction.

I typically use the “Hardie” method for measuring extinction, which takes fifteen minutes or less. It depends on having reliable magnitudes in both of your filter bands for a selection of stars that are at a variety of altitudes. I use the H&K first-order extinction stars. The Hardie method is described in Astronomical Techniques, chapter 8. I have pre-selected sets of stars for each month of the year. If my observing run takes place early in the night, I can use the set for the preceding month, and if up very late, use the set for the following month. If I am running for a very long time, I might use sets from two months at different times, just to check for consistency during the night. With the Hardie method, one cannot just plot the instrumental magnitudes, for each extinction star is of different brightness. Instead, one plots the *difference* between the standard and instrumental magnitudes (V-v) against airmass. The fitted line, again, gives the extinction coefficient.

Regardless of the method you use, it is dangerous to perform the fitting calculation without generating a plot and actually looking at it. An aberrant data point can skew the results, and it may be necessary to drop one or more values. A crummy collection of points may indicate unstable extinction that night. Even a crude diagram will suffice for this safety check.

5.2 Transformation (the easy way)

There are three methods to determine the epsilons:

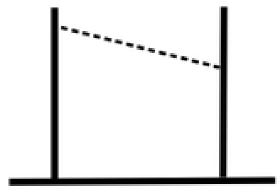
1. Blue/red star pair
2. All-sky (multiple stars, range of colors)
3. Cluster (ditto)

Epsilons are usually established only once a year but preferably based on multiple observing runs.

What, exactly, is the purpose of transformation? In differential photometry, we measure the difference in instrumental magnitudes between two stars, Δv. That difference is unlikely to match the difference established by a “standard” photometer. In the Johnson photometric system, the results from his photometer *are* the standard.

His response curve, like ours, is not flat, but we use his results as the anchor for own our work. Transformation, then, is an adjustment to our Δv so that it matches the ΔV of Johnson; or, so that ΔV - (Δv + transform) = 0.

Let's return to our response curve of the blue-sensitive photometer:



blue sensitive

To establish the V transformation, we will measure the instrumental magnitude difference between a bluish star and a reddish star. ΔV-Δv is the shortfall (or excess) of measured difference, where $\Delta V= V_{blue} - V_{red}$, and $\Delta v= v_{blue} - v_{red}$. Looking at the response curve, we see that a star rich in blue light will fare well, but a red-heavy star will lose, in comparison, a significant fraction of its brightness. Since v_{red} will be more positive (dimmer) than it should be, Δv will be too negative, and ΔV - Δv will be greater than zero. The transformation, which is added to Δv during reduction, must, therefore, be negative.

For any given pair of stars, the amount of transformation will depend on the color contrast: less contrast = less transformation. Therefore, we normalize our measured shortfall/excess by dividing it by the color contrast, to give us a coefficient of transformation, e_V:

$$\varepsilon_V= (\Delta V - \Delta v)/\Delta(B\text{-}V) \quad (*)$$

When we apply transformation to a variable/comparison combination, the correction will be $\varepsilon_V \cdot \Delta(B\text{-}V)$.

So this seems simple, in practice: measure a blue/red pair. Well, not so fast—I was loose with terminology. We actually need to measure $\Delta v0$, the extinction-corrected instrumental magnitude, not Δv. Anytime we throw extinction into the mix we are adding a complication that is best avoided. The solution has been to find blue/red pairs that are very close together. When such a pair is near transit, the extinction for the two stars is very nearly the same. Since the extinction corrected magnitudes are $v_{blue} - k'_V \cdot X$ and $v_{red} - k'_V \cdot X$, the corrected differential magnitude is:

$$(v_{blue} - k' \cdot X) - (v_{red} - k' \cdot X) = (v_{blue} - v_{red}) - (k' \cdot X - k' \cdot X) = v_{blue} - v_{red}.$$

The extinction drops out. Unfortunately, bright blue/red pairs are hard to come by. The AAVSO PEP webpages list a total of 12 in both hemispheres, but the Aquarius and Pegasus pairs have been deprecated as unreliable, and Andromeda is questionable.

The transformation observation of a pair is an extension of the standard sequence, but with no check star. The blue star is treated as the variable, and the red as the comparison. Instead of three variable star samples, we take seven, bracketed by eight comparison samples. The mean Δv so obtained is used in formula (*). We take seven samples because we want this measurement to be very reliable, and it is customary to compute the standard deviation of the differential magnitudes to quantify their consistency. Clearly, we want good skies for this measurement, but we do not need ideal transparency. What we need is *consistent* transparency during the sequence. The AAVSO procedure calls for conducting the sequence within one hour of transit for the pair. This minimizes differential extinction. For pairs that transit at low airmass you can push the time envelope.

The formula for ε_B is very similar to that for ε_V, needing only a correction for second-order extinction:

$$e_B = (\Delta B - \Delta b + X \cdot k''_B \cdot \Delta(b-v))/\Delta(B-V)$$

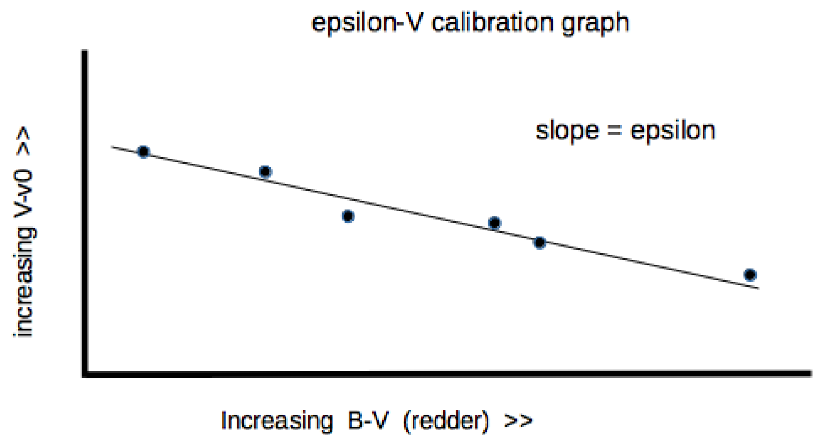
Which means that you must measure second-order extinction before reducing an e_B sequence (you can still collect the e_B data beforehand). Some people use a fixed estimate of k''_B to get around this.

5.3 Transformation (the hard way)

As noted, above, blue/red pairs are hard to find, and good calibration is dependent on high-quality reference magnitudes. Further, we want Δ(B-V) to be large, and we also want Δb and Δv to be large. Satisfying all these conditions is not easy. Our alternative is an *all-sky* calibration. Here, we sample multiple standard stars of varying B-V, which usually requires us to make the measurements over a large portion of the sky. These measurements must be corrected for first-order extinction, and therein lies the rub. If your skies are not uniformly transparent in space and time, proper extinction correction may not be possible.

An all-sky or cluster calibration for ε_V is illustrated below.[21] For standard stars of varying color, the difference between the standard and extinction-corrected instrumental magnitudes is plotted versus their standard color index. This measures the gap between standard and instrumental magnitudes as a function of color index, which should be linear. Epsilon is the slope of the fitted line. A cluster calibration, which uses stars in a single open cluster, has the advantage of not needing the first-order extinction correction,[22] making it more reliable (you should do this near transit, like the red/blue pairs). The problem is that calibration stars in the "standard" clusters are too dim for the SSP3, unless you have a monster telescope.

21 For ε_B, the vertical axis becomes B-b0.
22 Second-order correction still applies.

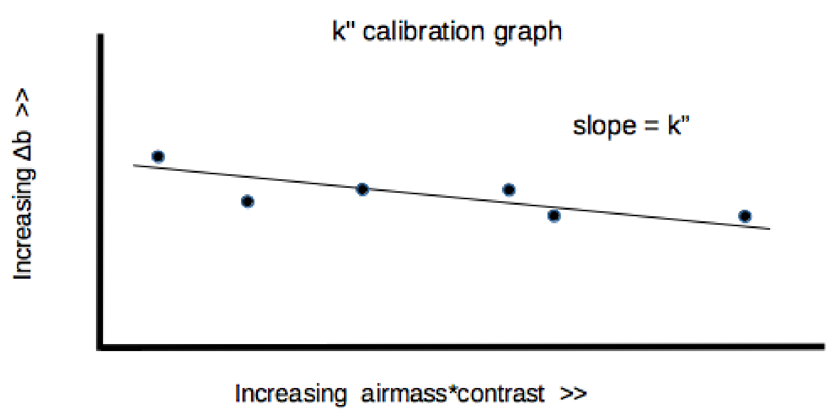


In the case of a star pair calibration, you are effectively doing the above fitting operation with just two points. Hence, it is important that the standard magnitudes be very reliable and the stars be as different in B-V as possible. For B and V bands, we really don't need the all-sky technique–the good red/blue pairs work fine.[23]

5.4 Second-order extinction

For calibrating second-order extinction, we return to the red/blue pairs. Instead of observing near transit, we follow a pair from high in the sky to low (or vice versa). Like first-order extinction, we want the observations reasonably evenly-spaced in airmass, and to cover as wide a range in airmass as is feasible. If your pair transits near an airmass of 1.0, you could sample them at X=1.2, 1.4, 1.6, 1.8, 2.0, and 2.2, which will come at decreasing time intervals (if the stars are setting). Your latitude, the declination of the pair, and your horizon will determine the range over which you can sample.

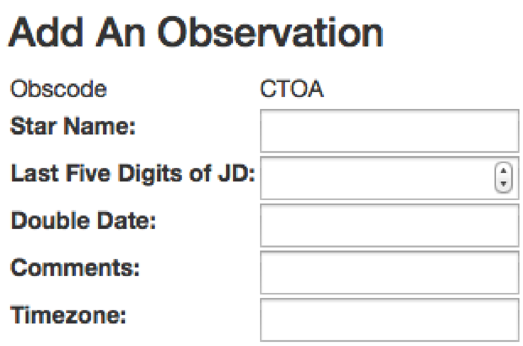
The coefficient is determined yearly, by plotting the instrumental B difference, Δb, versus the product of airmass and contrast, X·Δ(b-v). As the airmass increases, the difference will decrease (the blue star will get redder, and the red star will not change much). k" is the slope of the line fitted to the data points.



23 ΔI values are being established for I band calibration.

Appendix A: PEPObs

Raw V band observations can be processed on the AAVSO website go through a reduction tool accessed through the WebObs page. You are presented with a form in which to enter star identification, date and time, and samples. The “Double Date” is the pair of evening/morning civil dates for the night in question. This field is not parsed for format, it is just a note that may later be used by AAVSO staff if there is a later question about the correct date. The “Comment” field will be included in the data record stored for your observation. Your observer code is automatically populated (you must be logged-in on the AAVSO site to submit data). What you enter for Timezone will have no effect on the calculations. This tool only works for stars in the PEP database. It is presently accessed from the AAVSO home web page under "Submit and Access Data," thence under "Upload Photometry," and thence under "Unreduced PEP observations (PEPObs)." The direct link is currently https://apps.aavso.org/pepobs/.


Add An Observation

Obscode CTOA

Star Name:

Last Five Digits of JD:

Double Date:

Comments:

Timezone:

The form then continues with a series of cryptically-numbered lines, each having a time, count, and gain.

Time	Target	Deflection	Gain
4:25	3	1000	10
0:00	4	20	10
4:27	1	800	10
0:00	4	23	10
4:28	3	1004	10
0:00	4	21	10
4:29	1	799	10
0:00	4	18	10
4:30	3	1010	10
0:0	4	22	10
4:32	1	812	10
0:00	4	24	10
4:33	3	1007	10
0:00	4	22	10
4:35	2	750	10
4:36	4	26	10
4:37	3	1002	10
0:00	4	25	10

Add this observation to the report

The numbers refer to the following pattern: 3=comparison, 4=sky, 1=variable, 2=check. The sky samples are implicitly associated with the immediately-preceding star samples, and the time associated with a sky sample is not used. The time format is hh:mm. Sample counts are entered as integers, so you will round your average of three integrations. When you "Add" the observation, some consistency checks are performed, and, if passed, the observation is added to a list at page top:

Current Report

Star Name	JD	Magnitude
ALF ORI	2457456.6870	4.289

Submit this report

You can then enter data for more stars, or submit what you have.

There are bugs in WebPEP. The most serious is that it calculates sidereal time incorrectly (fast by 8 minutes, as I recall). Another: if you succeed in collecting some *very* consistent samples, WebPEP can report an error of 0.

Appendix B: References

Publications:

Photoelectric Photometry of Variable Stars, 2nd edition, Hall & Genet; especially chapters 9-14. Readily available on the used market. Fairly approachable. Spills a lot of ink on pulse-counting.

Astronomical Photometry, 2nd edition, Henden & Kaitchuck; especially chapter 4 and appendices G, H. More technical and still in print. Like Hall & Genet, it spends a lot of time discussing pulse counting systems.

Software for Photometric Astronomy, Ghedini. Harder to find.

Astronomical Techniques, various authors; see chapter 8 on PEP reductions by Hardie. Long out of print, but available at the Internet Archive (http://archive.org).

A word of caution as you start exploring outside of this document: historically, photometrists have worked a lot with V, B-V, and U-B in place of V, B, and U. This has certain advantages, but requires different math (eg: the transformation coefficient for B-V is mu (μ), and for U-B is psi (ψ)).

SSP Manuals:

SSP-3 and SSP-5 manuals, both Generation 1 and Generation 2 are available on the AAVSO PEP web pages.

Websites:

https://gcpd.physics.muni.cz/cgi-bin/photoSysHtml.cgi?0 General Catalog of Photometric Data (GCPD)
One of the most reliable online sources of bright star magnitudes in a variety of photometric systems. You must know the HD number of the star. R&I magnitudes in UBVRI system are Johnson, not Cousins.

http://simbad.u-strasbg.fr/simbad/sim-fbasic SIMBAD catalog
Has numerical information about a wide variety of astronomical objects with a flexible search interface.

https://www.aavso.org AAVSO home page

http://ssqdataq.com SSP data reduction software packages

Appendix C: Metadata

PEP metadata are recorded in the “notes” of the AAVSO extended format photometry record. Different fields in the notes section are delimited by the ‘|’ character. Fields are of the form KEYWORD=<value> and no commas are allowed. Avoid using quotation marks and apostrophes. Aside from the COMMENT, values should not have spaces. Multiple fields within the COMMENT should be separated by a semicolon.

SCOPE	optical tube used
SENSOR	photometer used
LOC	latitude/longitude of observer (to nearest degree)
INDEX	color index of reduction (BV, VI, etc)
K_B	first order extinction coefficient (here, in B band)
K_B_EST	estimated extinction
KK_B	second order extinction coefficient for B band (formerly KK_BV)
KK_B_EST	estimated second order extinction
TB_BV	transformation coefficient for B band in the BV index
TB_BV_EST	B transform when second order extinction is estimated
CREFMAG	comparison star magnitude
PROG	reduction program
DELTA	standard color contrast between variable and comparison
DELTA_EST	“estimated” delta (i.e.: a catalog value rather than a measurement)
COMMENT	observer comments, if appropriate (spaces allowed, delimit multiple comments with ';'

An example metadata section would be:

```
|SCOPE=10IN_SCT|SENSOR=SSP3|LOC=44N/131W|INDEX=BV|K_B=0.35|KK_B=-0.03|TB_BV=0.01|
DELTA=0.317|CREFMAG=7.22|PROG=TJC_PEP_5.0|COMMENT=POOR SEEING;HUMID|
```

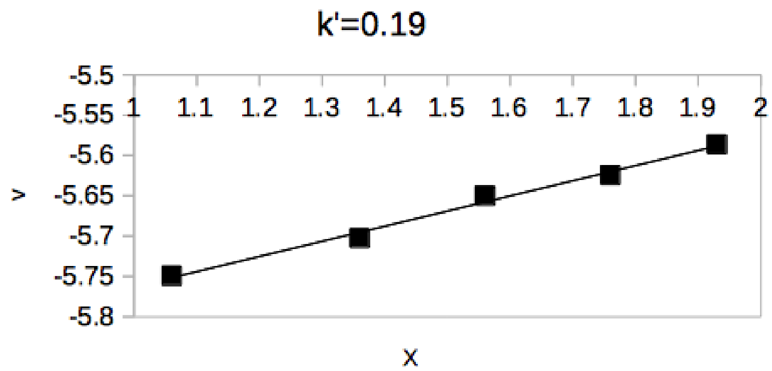
The PEP spreadsheet can automatically generate an AAVSO extended format data file that includes the above metadata.

Do not use double or single quotation marks in the metadata (this goes for fields like Telescope and Sensor in the spreadsheet).

Appendix D: Extinction examples

Below are example reductions of first and second-order extinction data, beginning with the "simple" determination of first-order extinction. For a single star, we record the instrumental magnitude, $-2.5\cdot\log_{10}$(net counts), at a range of airmasses during the night.[24] The extinction coefficient (k'_V in this case) is the slope of the line fitted to instrumental magnitude versus airmass. The graph is here presented with the magnitude brightening in the negative y direction, so that the positive slope of the line is seen directly.

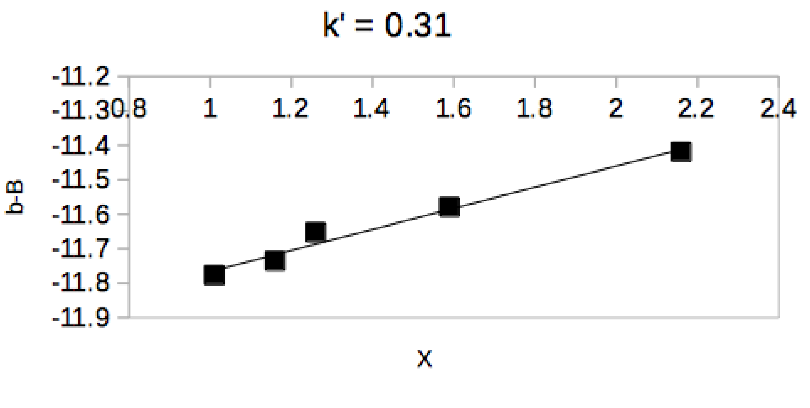
X	v
1.06	-5.7493
1.36	-5.7026
1.56	-5.6500
1.76	-5.6244
1.93	-5.5865



Next, here are data for determining first-order B extinction via the "Hardie" method. A selection of stars having reliable standard magnitudes are observed in quick succession. They are chosen so as to span an airmass range of about 1.0. The stars, below, were taken from the list of Appendix A in Henden & Kaitchuck. This particular set is a bit imbalanced, having three stars at X<1.3, but it still illustrates the procedure. The value of k'_B is the slope of the line fitted to the standard magnitude minus the instrumental magnitude (B-b) versus airmass.

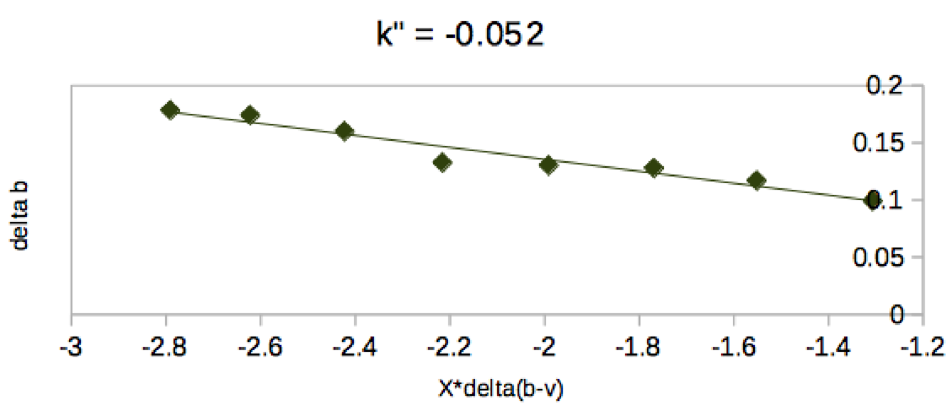
star	**X**	**B**	**b**	**b-B**
lambda Per	1.01	4.31	-7.4667	-11.7767
136 Tau	1.16	4.59	-7.1456	-11.7356
pi 2 Ori	1.26	4.36	-7.2930	-11.6530
phi Gem	1.59	5.08	-6.4993	-11.5793
gamma Cnc	2.16	4.68	-6.4993	-11.4191

24 For B band, we would be careful to choose a star with B-V near 0 to avoid second-order extinction.



Finally, a second-order extinction reduction. Here, we observe a red/blue pair over a range of airmasses during the night. Magnitude deltas are in terms of the blue star minus the red star, so $\Delta b = Blue_b - Red_b$.[25] Note how Δb increases as the airmass increases. Both stars are losing light as extinction grows, but the blue star has a lot of light at the blue end of B band, where second-order kicks in, so its b magnitude dims (increases), faster, and $Blue_b - Red_b$ becomes more positive.

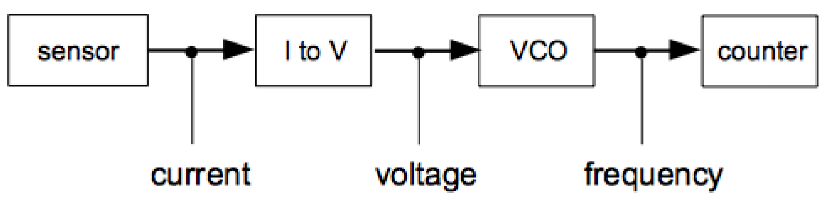
X	Δb	Δv	Δ(b-v)	X · Δ(b-v)
1.01	0.0991	1.3933	-1.2942	-1.3071
1.21	0.1169	1.3993	-1.2824	-1.5517
1.40	0.1278	1.3912	-1.2634	-1.7688
1.58	0.1302	1.3905	-1.2603	-1.9913
1.76	0.1326	1.3915	-1.2589	-2.2157
1.96	0.1599	1.3969	-1.2360	-2.4226
2.13	0.1740	1.4050	-1.2310	-2.6220
2.30	0.1783	1.3916	-1.2133	-2.7906



25 And Δ(b–v) = Δb–Δv.

Appendix E: SSP electronics

All the SSP sensors generate currents. If we could count the individual electrons coming out we could actually generate photon statistics, but such is not the case. The current from the sensor is fed into a circuit that produces a voltage proportional to the current (current-to-voltage converter). In the olden days, this voltage got fed into the strip-chart recorder. In the SSP, this voltage is fed into a voltage-to-frequency converter (VFC).[26] This device puts out a train of electronic pulses, proportional in frequency to the voltage. The pulse train is fed into a counter. The output of the counter is copied to a buffer at the end of the integration time, and the counter is reset to 0. The contents of this buffer is what you see on the display. The pulse train runs continuously–independent of the counter circuit–and that is what you see on the analog output port, or on the "Pulse" test pin on the circuit board of a Generation 2 photometer. This means is that if you sample the pulse train yourself, it doesn't matter if the display overflows. The voltage-to-frequency circuit is nominally rated to 10kHz, which translates to 100,000 counts in ten seconds. However, bench tests have shown excellent results up to 14kHz. Care must be exercised for count rates below 1kHz: linearity suffers when ten-second counts drop below one thousand.



26 Also known as a voltage-controller-oscillator (VCO).

Appendix F: Additional notes

a) When starting up in the evening, the photometer needs to stabilize, both thermally and electronically. Let the device come to ambient temperature before you put it to use. After you power it up, let it idle for ten minutes. Unless you are going to take a significant hiatus in data-collection during the night, leave the photometer on for the whole observing session.

b) I have seen a unit that had light leaks. I was using it under a dome that had red lights, and I noticed that my sky counts were elevated at times. It turned out that light was getting in through the eyepiece and reaching the sensor. The magnitude of the leak depended upon whether I was standing between the light and the photometer, casting a shadow. If you operate near a light source (eg: computer monitor) and you are seeing some inexplicable dark counts, try shielding the photometer case and see if the counts settle.

c) There is a phenomenon where the counts overshoot, then undershoot, then stabilize when switching from a bright star to a dim one. I have seen this on occasion, and so have others. What causes it, I do not know, but it is another reason to pay close attention to the counts. The workaround is to let a few integrations pass.

d) Airplane lights, satellites, and meteors can cause sudden increases in counts. Watch for them, and re-do suspect integrations. I once had to do photometry during a meteor shower, and got "hit" a few times. Elon Musk's space junk is now a significant problem.

e) The Generation 1 SSP3 is intended to run on an unregulated power supply of about 14 volts. In cold weather, a 12 volt supply may not suffice.

f) Although you get a 0-64K count range using SSPDataQ, it is still possible to overflow on a bright star at a too-high gain. If you get a data set that has impossibly low counts for a bright target, this is what happened. Do not despair: add 65,536 to the low counts.

g) If your Generation 1 SSP3 display is completely blank with no overflow light, the unit is saturated (you're probably playing with it in daylight).

h) When looking through the eyepiece, you exhale on the photometer case. In cold weather, you will fog or frost-over the count display. Learn to breathe out of the side of your mouth in winter.

i) The weak mechanical link in SSP hardware is the switches. The levers have no protection against impacts, and it is relatively easy to break them off. When shipping a photometer put some kind of protection around the levers. Replacement switches are hard to find, but we have a small stock of spares.

j) If you are working in I band be careful about ambient red light, especially under a dome. The SSP3 is *very* sensitive to near infra-red and such light bouncing its way into the optics can throw off your counts.

Appendix G: "Starparm" target list file

The list of recognized PEP targets is found in the "starparm" file on the PEP web pages. This rather cryptic file lists variable, comparison, and check star "triplets," one per line. The coordinates, magnitudes, and (for var and comp) B-V colors are provided along with some other information. The columns in the file (by group) are as follows:

For the variable star.....
usename, auid, desig, vrah, vram, vras, vded, vdem, vdes, vspec, vvmag, vbmv,

`variable name, database ID, old-style ID, RA hour, min, sec, Dec deg, min, sec, spectral type, V magnitude, B-V color`

For the comparison star...
cname, crah, cram, cras, cded, cdem, cdes, cvmag, cbmv,

`comparison name, RA hour, min, sec, Dec deg, min, sec, V magnitude, B-V color`

For the check star...
kname, krah, kram, kras, kded, kdem, kdes, kvmag,

`check name, RA hour, min, sec, Dec deg, min, sec, V magnitude`

And finally...
deltabmv

`(B-V) of variable minus (B-V) of comparison`

Generally speaking, observers only need to consult starparm to obtain the var/comp/check sky coordinates to program into their mount controllers. However, if you use SSPDataQ, get the comparison magnitude and color from this file.

Many old entries in the file have the RA and Dec seconds specified to an absurd number of decimal places. For your mount, round Dec seconds to the nearest second and RA seconds to the nearest tenth.

Appendix H: Cutting corners on sky samples

Our standard procedure calls for a sky sample to accompany every star sample. While this is the most reliable approach it can be overkill. If your sky conditions and ambient temperature are well-behaved it is feasible to skip some sky samples. This is most appropriate when performing long sequences such as two-filter transformation calibration or differential photometry using more than two filters. But since there are no plans to make our data reduction tools accommodate less-than-complete sky sampling you must interpolate for the missing data. As an illustration, consider the following single-filter transform sequence (of already-averaged integrations):

red star	1000	red star sky	40		
blue star	700			blue star sky	39
red star	1002				
blue star	699				
red star	1005				
blue star	702				
red star	1010				
blue star	705				
red star	1012				
blue star	706				
red star	1013				
blue star	705				
red star	1013				
blue star	707			blue star sky	43
red star	1014	red star sky	45		

Over the course of the sequence the red star sky counts rose by five and the blue star sky counts rose by four. This would typically be caused by a gradual change in temperature. It is reasonably safe to assume that the change was linear over the time span, so we could assign the following sky estimates (in italics):

red star	1000	red star sky	40		
blue star	700			blue star sky	39
red star	1002	*sky estimate*	*41*		
blue star	699			*sky estimate*	*39*
red star	1005	*sky estimate*	*41*		
blue star	702			*sky estimate*	*40*
red star	1010	*sky estimate*	*42*		
blue star	705			*sky estimate*	*41*
red star	1012	*sky estimate*	*43*		
blue star	706			*sky estimate*	*42*
red star	1013	*sky estimate*	*44*		
blue star	705			*sky estimate*	*43*
red star	1013	*sky estimate*	*44*		
blue star	707			blue star sky	43
red star	1014	red star sky	45		

Strictly speaking the interpolated values should be calculated as a function of elapsed time and rounded, but I just eyeball them. You might want to take sky samples halfway through the sequence and interpolate the two halves separately. Because incomplete sky sampling disrupts the normal filter change sequence (for two or more filters) you must stay *very* alert to what you are doing with the slider.

Appendix I: Built-in counter overflow

Optec photometers have a built-in four digit counter to display the results of integrations. Of necessity, the display can only show counts up to 9999, a major limitation. In Generation II photometers a count over 9999 will cause the display to read "OVER." About the only thing you can do about this (unless you are using the computerized interface) is switch to five second integrations and record six such integrations instead of the usual three. But happily, the Generation I photometers offer a workaround for the overflow situation.

Your car odometer displays some number of digits, say five. Five decimal digits will count up to 99,999. When you go the additional mile the odometer reads 00,000 - the hundred thousands digit is lost. But you know that the odometer "rolled over," so as the counter increases again you know that the true mileage is the displayed count plus 100,000. Same thing goes for the Generation I counter. The low-order four digits are correct and all we need to do is establish the missing higher-order digit in order to determine the correct count.

When the Gen. I counter overflows a red LED will light up to the right of the display. If the four displayed digits are 5392 the true count might be 15392 or 25392 or even 95392. The high-order digit is easy to find: just switch your integration time to one second. Approximately speaking, 15392 will become 1539, 25392 will become 2539, and 95392 will become 9539, revealing whether the high-order digit was 1, 2, or 9. I say approximately because the "timer" in Gen. I photometers is based a simple analog circuit, so the one and ten second integrations are not exactly one and ten seconds, nor is the one second integration exactly one-tenth of the ten second.

Once you know the missing digit, go back to ten second integrations and when recording the counts simply add that digit multiplied by 10,000. In some cases your star counts might be on the hairy edge between ten-thousands, bouncing between, say about 19990 and 20010. Just be watchful of this situation—identifying the right high-order digit for a particular integration will be straightforward. If you make a mistake in this regard it will be obvious when you reduce the data, for the computed magnitude will have a huge uncertainty.

If you are dealing with a *very very* bright star the ten second counts might exceed 100,000.[27] In this case, cutting the integration to one second will only reveal the ten-thousands digit, the hundred-thousands digit still being lost. Use good sense. That digit is either 0 or 1 and which one is correct should be obvious.

But to return to the Generation II photometers. Although the display goes only to 9999, the internal computer-accessible counter goes to 65535 then wraps-around back to 0 and starts over. If a very bright star is giving you low counts, just add 65536 to what the computer reports.

27 Think Betelgeuse at maximum light.

Appendix J: When I reduce data for you

From time to time some of you will have occasion to send me PEP data for reduction - particularly for calibration observations. I use a program to process these observations, and it helps me to have your data in the format my program naturally takes as input. To begin with, I need a plain ASCII text file - no pdfs, jpgs, pngs, or .docs. Second, the file *name* needs to include some indication of 1) who the observer is, 2) what the target is, 3) the filter bands, and 4) the Julian date (not the calendar date) of the observation. Somewhere in the file put your extinction and transformation coefficients. Data lines in the file have the following format:

Julian Date, UT hour, UT minute, UT second, mean star counts, mean sky counts, 0, 0

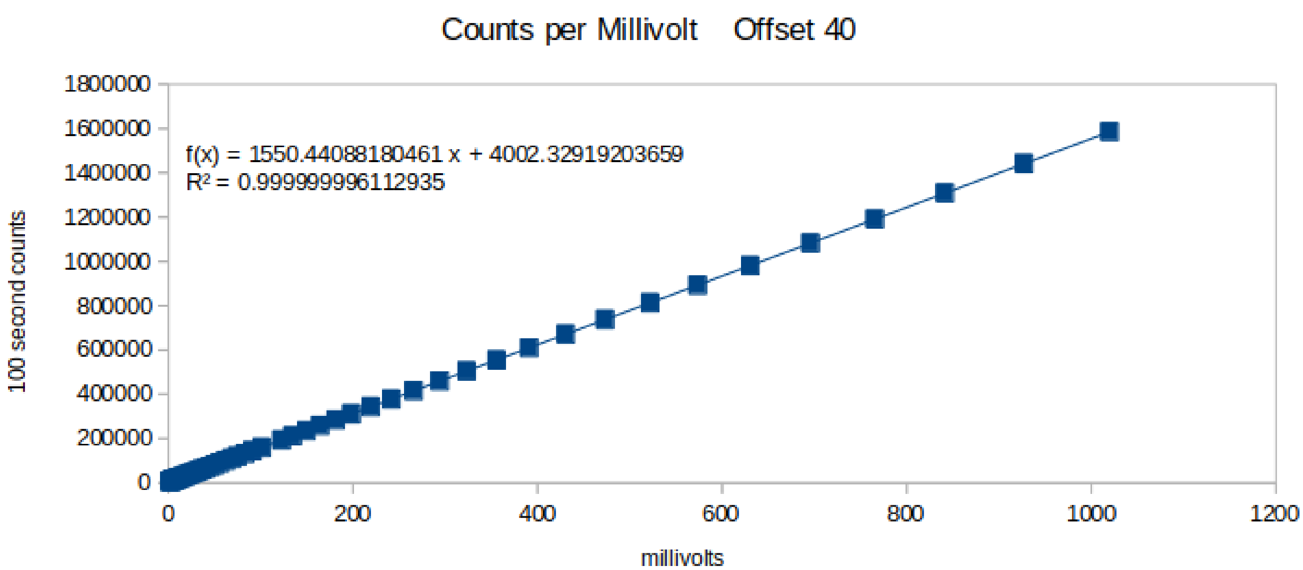
Put a space after each comma. If you don't record UT seconds, leave them as 0. Leave leading zeroes off of all numbers. If the observation is through two filters, alternate between the two bands with the shorter wavelength first. Below is an example of an actual input file for Betelgeuse.

```
2459304, 3, 42, 58, 2652.667, 1181.667, 221, 2507  #  B_1
2459304, 3, 44, 5, 4462.0, 1189.333, 340, 1261  #  V_1
2459304, 3, 46, 54, 15711.0, 1189.333, 245, 2523  #  B_2
2459304, 3, 47, 40, 66967.0, 1221.0, 468, 415  #  V_2
2459304, 3, 49, 50, 2630.667, 1181.333, 376, 2506  #  B_3
2459304, 3, 50, 38, 4491.667, 1191.333, 336, 580  #  V_3
2459304, 3, 53, 52, 15685.333, 1187.0, 567, 839  #  B_4
2459304, 3, 54, 34, 66838.333, 1218.333, 608, 977  #  V_4
2459304, 3, 56, 40, 2621.0, 1185.333, 403, 950  #  B_5
2459304, 3, 57, 26, 4413.333, 1197.333, 206, 704  #  V_5
2459304, 4, 0, 22, 15602.0, 1192.333, 1215, 1265  #  B_6
2459304, 4, 1, 8, 66422.0, 1226.333, 215, 597  #  V_6
2459304, 4, 3, 9, 2619.0, 1192.0, 220, 1460  #  B_7
2459304, 4, 3, 53, 4406.667, 1203.333, 124, 1276  #  V_7
2459304, 4, 6, 41, 1377.667, 1194.0, 108, 1462  #  B_8
2459304, 4, 7, 27, 1765.667, 1205.667, 195, 967  #  V_8
2459304, 4, 9, 29, 2602.333, 1197.333, 535, 2540  #  B_9
2459304, 4, 10, 53, 4432.0, 1208.333, 3948, 969  #  V_9
```

The B_ and V_ are notes that I put in the file. They are preceded by a '#' that tells the program to ignore them. You need not put them in. As you can see, the last two numbers of each row are not zero. I run my raw integrations through a pre-processing program that computes "empirical" signal-to-noise ratios for the star and sky data. I use them for diagnostic purposes, but you will leave them as zero. When working with color pairs, you need not take the short-wavelength data first. That is, if you are sampling a star in B&V, it is fine to sample V before B. But in files you send to me,put the lines of B data before the V data for each pairing. The mean counts in my example are expressed to three decimals. That was an arbitrary choice on my part. I wanted to preserve some fractional precision but didn't want fill my human-readable files with endless floating-point digits. You can use integers if you like, though in low-count situations it could slightly affect the results.

Appendix K: SSP linearity

Ideally, we want our photometers to yield counts that rise linearly with star brightness. That is, if one star give us *x* counts, a star twice as bright should give *2x* counts.[28] And we want this property to hold over a very wide range of brightnesses. I have performed some testing of SSP3 electronics and found that they behave very linearly, though with some degradation at low count levels. Appendix E illustrates the SSP "signal chain." The first two stages can be assumed to be highly linear. The potential weak link is the voltage-to-frequency converter (a.k.a. voltage controlled oscillator). In the SSP3 it is fairly easy to disconnect the VCF from the sensor electronics, directly apply test voltages, and see the resulting counts. The graph below shows a particular test run over a wide range of count rates.



The integrations were made with an external pulse counter and lasted one hundred seconds. The offset was forty counts per second, and for every millivolt (*v)* of input signal the counts equaled approximately 1550*v*+4000 with excellent linearity past 14000 counts/second, as indicated by the correlation coefficient (R^2). This is far beyond the stated range of the voltage-to-frequency circuit. But looking at the low end of the test samples shows less excellent performance. On the next page is part of a spreadsheet containing the actual test data.

28 After subtracting the sky counts, which include the "offset."

mV	Counts	model	Model-counts	% error	rate (Hz)
1.326	6098	6058	-40	0.657	61
1.77	6786	6747	-39	0.584	68
2.373	7721	7682	-39	0.514	77
2.771	8336	8299	-37	0.451	83
3.3	9156	9119	-37	0.408	92
3.844	9977	9962	-15	0.148	100
4.527	11037	11021	-16	0.144	110
5.445	12459	12444	-15	0.117	125
6.044	13386	13373	-13	0.096	134
7.026	14907	14896	-11	0.076	149
7.958	16355	16341	-14	0.087	164
9.053	18052	18038	-14	0.075	181
10.012	19540	19525	-15	0.075	195
11.387	21672	21657	-15	0.068	217
13.184	24464	24443	-21	0.085	245
14.552	26581	26564	-17	0.063	266
16.243	29194	29186	-8	0.027	292
19.059	33551	33552	1	-0.004	336
19.932	34913	34906	-7	0.021	349
22.238	38483	38481	-2	0.005	385
24.888	42589	42590	1	-0.002	426

The columns are as follows: *mV* is the millivolt input, *counts* is the 100 second counts, *Model* is the (rounded) count expected based upon the formula 1550v+4000 derived from the totality of test data. *Model-counts* is the difference between the two prior values. The *% error* is that difference divided by the model counts (times 100), and *rate* is the count rate in Hertz. At the bottom of this table excerpt you can see that the error is tiny, and in the full table it is almost entirely tiny as the count rate goes higher. But at a rate of 61 Hz, the error is over ½%. As the rate increases there is a sharp drop-off in error at about 100Hz, which corresponds to a ten-second count of 1000.

The implication is that it helps to have the counts at least as high as 1000[29] at offset 400, and at a lower offset of 120 this also holds true. But the question arises: for a very dim star, why not just crank the offset up very high to make up for the low star counts? This helps up to a point, but at an offset of about 700, for instance, the cross-over to lower errors comes at somewhat higher count rates, so turning up the offset is not a cure-all. If you were to use a bright comparison (at 24.888 mV) with a dim variable (1.326 mV) your measured magnitude (after subtracting away the offset) would be wrong by about 0.02.

That being said, if your comparison and variable stars are *both* giving low counts, then the consequences of nonlinearity are much reduced. Early photometrists who worked with significant linearity problems would strive to match the brightnesses of the variable and comparisonas best they could.

It is harder to perform this test with an SSP5 and I have not tried to.

29 Be aware that the analog timing in the Generation 1 SSP3 is a bit "slow," so that a ten second integration is actually over ten seconds. The true count rate is lower than what you see on the display.

Appendix L: War Stories

If you stick at the photometry business, you will eventually find yourself trying to untangle mysteries in the observational data of yourself and others. I want to at least touch on some factors that come into play in these investigations. “Debugging” differential photometry requires thinking about measurement problems in a new way.

For starters, every reduced magnitude is based upon measurements of two stars, not one. A magnitude problem could be caused by trouble with either measurement, or both. A latent problem could be hidden when errors cancel out, only to emerge with different targets. Let's consider variable *Var* and comparison *Comp,* where *Var* is brighter than *Comp*. The differential magnitude $\Delta M = M_V - M_C$ will be negative.

Magnitude:	4.0	3.0	2.0	1.0	0.0	-1.0	-2.0
			|		|		
			Comp		*Var*		

$$\Delta M = M_V - M_C = 0.0 - 2.0 = -2.0$$

If we make a “hot” measurement of *Var*, one that is too bright, M_V moves further to the right on our scale, and ΔM becomes more negative. This makes our reduced magnitude, $M_C + \Delta M$, more negative, brighter. But if our measurement of *Comp* is hot, the M_C moves closer to M_V, and ΔM becomes less negative. Our reduced *Var* magnitude will become dimmer, not brighter. Conversely, “cold” measurements will have the opposite effects. As an exercise, try swapping the relative positions of *Var* and *Comp* on the magnitude scale.

What might cause a hot measurement? An example: In 2014 I started taking B band data for the first time. One of my targets was CH Cygnus. This star was also being followed by Jerry Persha, inventor of the SSP devices. I was alarmed to see that my B band magnitudes were about 0.25 brighter than his—a very large amount—but my V magnitudes agreed well. I reduce my data with a homegrown program, so I first assumed that I had a software bug. But I couldn't find any problem with the B band code, and, furthermore, my check star magnitudes were reasonable. Jerry used the same Optec filters as I did, so filter problems did not seem to be an explanation, but he suggested that I might have a “red leak” in my B filter. A filter leak allows light from outside the intended passband to reach the sensor. This will make the star appear brighter. A quick check on the catalog magnitudes of CH Cyg in increasingly red bands (left-to-right) showed:

B	**V**	**I**	**J**	**H**
8.77	**7.08**	**5.345**	**0.76**	**-0.35**

(J and H are actually near-infrared). The clue here is the huge difference between B and J: J is eight magnitudes brighter. If the blue filter were letting even a little of that light through, we could have trouble. But why wouldn't Jerry have this same problem with his filter? The answer was that he had a leak, too, but his photometer couldn't *see* it. Jerry uses an SSP5, the photomultiplier-based photometer. The photomultiplier tube was insensitive to light redder than the R band. I had an SSP3, which uses a photodiode sensor that, in principle, might detect the J band light. But how to prove this? The solution was some inventive filter-swapping by Jim Kay between an SSP3 and a near-infrared SSP4. The two-step process worked as follows: first, the JH filters from the SSP4 were installed in the SSP3, and the SSP3 pointed at an IR-bright star. Jim confirmed that J band light was getting detected by the SSP3 sensor, even though such light was outside the nominal wavelength range of the device. Next, Jim put the BV filters on the SSP4, and confirmed that it could detect light through the B filter. The SSP4 photodiode is definitely not sensitive to B band, so near-IR must have been leaking through it. (Optec has now come out with a new B filter). Optec never saw this leak in quality-control testing, because the range of wavelengths to which the B filters were subjected did not extend to the near-IR. Observers had not noticed it because so few stars have such a huge IR excess. My check star did not have the excess, so it was unaffected.

This kind of detective story is not uncommon, and it illustrates the thought process needed to explain aberrant readings—particularly the need to think about your photometry rig as a *system* of interacting components. Other leak stories exist in the photometric history books. An interesting one took place during Nova Delphinus 2013, where certain observers where getting hot magnitudes in V band. In PEP-land, we use Optec filters almost exclusively, but the CCD observers buy their filters from a variety of sources. Each manufacturer's filters will have slightly different characteristics, compounding the difficulty of making everyone's measurements agree. In this case, the V filters from one vendor had a passband that extended too far into the red. We don't demand that all filters have identical cutoffs—that's one of the reasons for transformation. But just to the red side of V band is the location of the hydrogen-alpha emission line that was a strong radiator in the nova. Broad-band photometry does not cope well with emission lines. A proper V filter will not pass that line, but the suspect filters had such a long tail on the red side that some of the Hα was sneaking through.

Appendix M: Constant stars

AAVSO inadvertently provided us with a handy way to check our photometry. In the main database there are stars that were once thought to be variable but are now believed constant. It is still possible to upload data for these stars, allowing inter-observer comparisons. The larger the value of Δ(B-V) the more strictly the accuracy of your transformation will matter. Likewise, observations at higher airmasses are more sensitive to extinction calibration. You can observe in V band only, but take BV if you can. These stars are all in the northern hemisphere - southern targets are harder to come by.

star*	**V**	**B-V**	**comp**	**V**	**B-V**	**Δ(B-V)**	**check****
alf cas	2.225	1.168	HD 5395	4.629	0.956	0.212	HD 9927
RU Cas	5.558	-0.091	HD 6960	5.551	-0.066	-0.025	HD 5015
RR Ari	5.750	1.188	HD 13174	4.989	0.335	0.853	HD 13363
NSV 16300 (Ori)	5.695	-0.210	HD 34658	5.340	0.416	-0.626	HD 36134
niu Aur	3.960	1.133	HD 38656	4.510	0.951	0.182	HD 41357
1 Gem	4.158	0.827	HD 38751	4.877	1.019	-0.192	HD 43039
NSV 2877 (Aur)	4.338	1.019	HD 40111	4.823	-0.06	1.079	HD 42471
SY UMa	5.28	0.07	HD 82328	3.175	0.462	-0.392	HD 82621
alf Com	4.318	0.454	SAO 82650	5.990	0.39	0.064	SAO 82692
NSV 6687 (UMi)	4.275	1.434	HD 136726	5.013	1.369	0.065	HD 142105
VW Dra	6.319	1.094	SAO 17360	6.72	0.5	0.594	SAO 17312
NSV 24912 (Aql)	5.118	0.551	HD 185018	5.978	0.881	-0.330	HD 188310
51 Peg	5.467	0.655	HD 218235	6.157	0.441	0.214	HD 215510

* These are the "PEP" designations for submitting data. NSV 16300 is HD 35299, NSV 2877 is HD 43039, NSV 6877 is HD 127700, NSV 24912 is HD 187691, and niu Aur is the PEP identifier for nu. Aur (HD 39003).

** The check star is unnecessary for our purposes but depending upon your reduction tool the data for it may be required input. You could put in fake numbers if you like (but do comment "no check star"). Magnitudes are not given here.

Appendix N: Erratic counts

The SSP is vulnerable to moisture. At the "front end" of the electronics is a huge (giga-ohms) resistor. If moisture accumulates there then electrons can sneak through the water instead of the resistance path. The result is erratic counts. In a humid environment the photometer should be kept indoors when not in use, and storing it with a dessicant pack is not a bad idea. I live in an arid climate but when there is rain I take the photometer off of the telescope.

Should your SSP suffer from excess hydration, all is not lost. The moisture can be driven off by gentle heating. Once technique that has been successfully applied is to leave the photometer in a sunlit car for a few days with the windows rolled up (assuming that the air inside is dry). Another option is to place the unit under a heat lamp. It may help to flip the mirror up, extract the filter slider, and even remove the back cover of the SSP.

Another problem that can appear is misalignment of the photodiode with respect to the eyepiece reticle. You think you are centered on your target star but you're not and you have have slipped off of the "sweet spot" of reliable sensitivity. You can diagnose this condition by placing a star just inside the target circle at the 12, 3, 6, and 9 o'clock positions. The counts at 12 and 6 should be similar as should the counts at 3 and 9 (your tracking must be good).

Re-alignment is tricky and Optec basically won't do it anymore. The most reliable approach is to have Gerry Persha do it. If that is not an option then the adventurous observer can try to adjust the diode with the four allen screws, with the object of getting reasonably consistent counts just inside the periphery of the target circle. This will not be a perfect alignment but it's the best you can do without Gerry's optical bench. The diode is held in a carrier that can be pushed (not pulled) with the allen screws. Pushing is accomplished by tightening a screw. Before tightening, one must first loosen the opposite screw, and this will determine how far the carrier can be pushed.

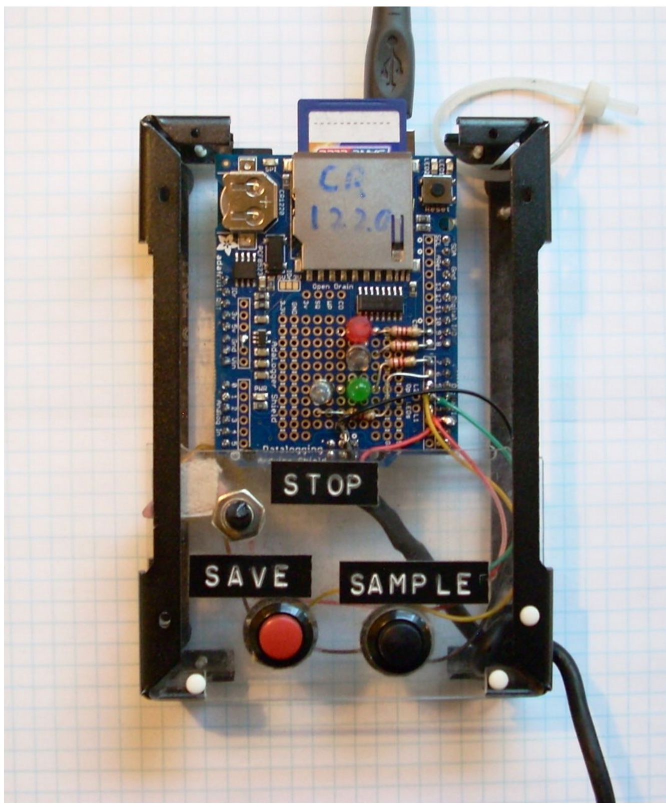
Let's say that the left-hand periphery of the reticle circle is giving higher counts than the right. In this case (if memory serves) we want to push the carrier towards the left - towards the excessively high counts.[30] On the other hand, if the counts at the top of the circle are too high, we push the carrier down - away from the high counts.

The allen wrench required is 3/32". It is best to have two of them inserted in pairs on the opposing screws. If you try to keep switching one wrench back and forth you will become endlessly confused about which direction to turn it. As for the screw on the underside of the photometer, if you insert the short end of a wrench in the socket the weight of the long end tilts the short end to one side and it will stay in the socket.

30 On account of the side-to-side reversal brought on by the mirror.

Appendix O: External data logger

Optec photometers have a built-in four digit counter to display the results of integrations. Of necessity, the display can only show counts up to 9999, a major limitation. Generation 1 photometers have a “pulse” output available through a phono jack.[31] This is the very same pulse used to drive the internal counter and it can be used by an external device that is not limited to four digits. Optec, in fact, used to sell a circuit board for IBM-compatible computers that could log the counts. For my own photometers, I have built an Arduino device to record the pulse output and write the timestamped results to an SD card. It consists of an Arduino Uno computer with an Adafruit datalogger “shield” having a real-time clock and an SD card slot. It is configured to make three back-to-back integrations of ten seconds every time it is activated. Pushbutton controls and LED indicators make up the user interface. If you enjoy electronic projects, this makes a work-saving addition to your photometry rig.



In the picture above you can see three buttons: *Sample*, *Save*, and *Stop*. There are four lights: a red one at top, blue in the middle (hard to see), and two greens at the bottom. All of these “peripherals” are connected to digital input/output pins of the Arduino. Upon pressing *Sample*, the Arduino will begin counting pulses and turn on the right-hand green LED. After counting for ten seconds, the left-hand LED is lit, instead. After ten seconds more, both LEDs are illuminated. When thirty total seconds have elapsed, both LEDs are turned off. Thus, they are counting up to three in binary, marking the progression through the integrations. The sample remains in RAM along with any subsequent samples until the *Save* button is pressed. At that time, all samples are written to the SD card (both green LEDs are turned on during this operation as a confirmation).

31 Generation 2 photometers do have a pulse signal inside if you know where to look for it.

The *Stop* button operates in two ways. It is used in conjunction with either the *Sample* or *Save* buttons. If *Stop* and *Save* are pressed together, all the samples in RAM are erased. If *Stop* and *Sample* are pressed together, the last sample is erased. If *Stop* and *Sample* are pressed while a sample is in progress, that sample is terminated. When the stop function is invoked, the red LED is lit as confirmation.[32]

The SD card driver is, unfortunately, a RAM hog. As a consequence, the Arduino has room for only sixty samples (and that only with some software tricks). When memory is full, the two green LEDs are made to flash as an indication that the existing samples must be saved to free up space.

And the blue LED? It's the pulse "heartbeat." It flashes once for every eight pulses that come in, giving the user not only proof that the photometer is connected, but also a qualitative indication of the count rate. The pulse goes to a digital input that calls an interrupt service routine that counts the pulses and also flashes the LED. This routine is operational all the time, not just during samples.

The logger writes output files with names of the form <filename>.PHO, where filename is five digits of Julian date, one digit of UT hour, and two digits of UT minute: DDDDDHMM.PHO. For hours past 9, the letters A-N are used. The file contains lines of the form:

```
JD   UT_hour   UT_minute   UT_second   integration1   integration2    integration3
(e.g.:   2459098 7 23 15 1500 1497 1505)
```

I had problems with my first Adafruit shield. The realtime clock might lose the time and sometimes data writes to the SD card would fail. The difficulty might have been that so little RAM was available (when I compiled the logger "sketch" there was a warning to that effect from the compiler). In any event, I added a function to the software that lets me check the time: at startup, the red LED flashes, waiting for the user to press *Stop*. When that happens, the software turns on the blue LED and flashes the current "tens" of minutes with the red (e.g.: if the time is 3:25, it flashes the red LED twice). It then turns off the blue light, turns in back on, and flashes the number of minutes modulo ten (five flashes in the above example). If the clock has gone off the correct time, it will be *way* off, and this little test will detect it. I have since discovered that an ordinary USB power adapter is *not* adequate for reliable operation. I now use a regulated 5V supply.

When using the logger, I put the photometer into one second integration cycles. This has no effect on the pulse output, and it lets me quickly see bad trends when they occur. Also, the pulse runs continuously. This means you can starting integrating from the moment the mirror is flipped. Furthermore, if the built-in counter overflows, the pulse output is unaffected. I have made linearity tests at high count rates and found excellent results all the way up to 140,000 counts in ten seconds.

32A note regarding use of the *Stop* button: *Stop* is pressed first and then either *Sample* or *Save*. Also, if a sample is in progress, the *Stop* and *Sample* buttons must be held down for up to one second until the red light comes on.

Below is the pin-out for my datalogger. Each LED has a 2.2k resistor in series.

Digital pin	connection
2	Pulse input
3	Stop button
4	Sample button
5	Save button
6	Right hand green LED
7	Left hand green LED
8	Blue LED
9	Red LED

Additional notes:

Amazingly, the real-time-clock does not keep particularly good time! Mine gains about a minute per week. You will likely not be able to initialize the RTC and then let it run indefinitely. Keep the Arduino development environment installed on your computer and reload rtcinit and logger from time to time. [Remember that you can check the current minute of the RTC by resetting the Arduino and pressing the STOP button]

The Arduino compiler may complain about low memory available when processing the logger sketch. With the limit of sixty samples I have not run into trouble. However, once you remove the SD card from the Arduino a reset will be needed after you re-insert it. Otherwise writing data to the card will fail. I don't know if this is a low-memory symptom. Since you will typically only remove the card at the end of an observing session this is not a big deal. As part of your nightly preparations make it a habit to reset the Arduino and you should be fine. If a data write should fail, the two green LEDs will flash five times. Your data are not recoverable - apply a reset.

Below is the logger source code. There are utility routines and a main loop. The top of the main loop checks for *Stop* operations. The middle of the loop performs integrations, and the end of the loop checks for writing a file. You can see that the ten second integrations are actually broken up into one second chunks. This is so the code can check for the *Stop* signal. To conserve memory, counts are only sixteen bits (post-processing software on my computer can compensate for wraparound). Also, the start times of individual samples are only recorded to sixteen bits, the full thirty-two bits being recorded for only the very first sample.

```
/*
 data logger for new wiring
 2 September 2020

 Accumulates samples consisting of three integrations
 */



#include <Wire.h>
#include "RTClib.h"
#include <SPI.h>
#include <SD.h>

// ************************************************
// DEFINITIONS
// ************************************************

// For accessing SD card
#define CHIPSELECT  10

// input button pins
#define STOP   3
#define SAMPLE 4
#define SAVE   5

// LED pins
#define GREEN0 7
#define GREEN1 6
#define BLUE   8
#define RED    9

// maximum number of samples
#define MAX_SAMPLES 60

// integrations per sample
#define INTEGRATIONS  3

// seconds per integrations
#define SECONDS 10
```

```
// *********************************************
// GLOBAL VARIABLES
// *********************************************

// Interrupt counter
// Use to count pulses from the SSP during integration
volatile unsigned short count = 0;

// Real time clock
RTC_PCF8523 rtc;

// sample counter
unsigned char sample=0;

// integration counter
unsigned char integration;

// utility index variable
unsigned char ii;

// 2^16 wrap indicator
unsigned char wrap;

// 32 bits of unix time at first sample
uint32_t allUnixTime;


// whole Julian day 1 Jan 1970 00:00
#define unixtimeJD 2440587L

// Sample data structure
typedef struct Sample
{
  unsigned short lowUnixTime;       // lower 16 bits of Unix time
  unsigned short integrations[3];   // 10 second integrations
};

// Main data structure
Sample samples[MAX_SAMPLES];
```

```
// ********************************************************
//   UTILITY ROUTINES
// ********************************************************

////////////////////////////////////////////////////////////////////
// the setup function runs once when you power-up or press reset
void setup() {
  short succeeded;

  // initialize input pins
  pinMode(STOP, INPUT_PULLUP);   // cancel
  pinMode(SAMPLE, INPUT_PULLUP);  // sampling trigger
  pinMode(SAVE, INPUT_PULLUP);  // output trigger

  // initialize LED pins
  pinMode(GREEN0, OUTPUT);        // sample-in-progress LEDs
  pinMode(GREEN1, OUTPUT);
  pinMode(BLUE, OUTPUT);        // interrupt heartbeat LED
  pinMode(RED, OUTPUT);        // cancel LED

  digitalWrite(GREEN0, LOW);
  digitalWrite(GREEN1, LOW);
  digitalWrite(BLUE, LOW);
  digitalWrite(RED, LOW);


  Wire.begin();

  // try to access SD card
  succeeded = 0;
  while (!succeeded)
    succeeded = SD.begin(CHIPSELECT);

   attachInterrupt(0, countIsr, FALLING);  // start interrupt service on pin 2

  //  This little bit of code flashes the red led for the current minute.
  //  First, red light flashes.  User presses "stop", red light goes out and blue
  //  light comes on for 1/2 second, then red flashes are given, one for each
  //  (minute/10) then blue light goes off 1/2 second, turns back on and a red flash
  //  is given for each (minute%10) then blue light turns off and unit
  //  starts normal operation.

  char bit = HIGH;
```

```
  // Flash red while waiting for signal to start
  while (digitalRead(STOP)==HIGH)
    {
      digitalWrite(RED, bit);
      delay(250);
      bit= bit ^ 0x1;
    }

    // Turn off RED, flash BLUE
    digitalWrite(RED, LOW);
    digitalWrite(BLUE, HIGH);

    // wait 1 second
    delay(1000);

    // get current UTC minute modulo 10
    char minutes;
    allUnixTime = rtc.now().unixtime();
    minutes = unixMinutes(allUnixTime);

    flashem(minutes/10);
    digitalWrite(BLUE, LOW);
    delay(500);
    digitalWrite(BLUE, HIGH);
    flashem(minutes%10);

    // signal we are done
    digitalWrite(BLUE, LOW);
}

/////////////////////////////////////////////////////////////
// flash Red light n times
void flashem(char n)
{
 char count=n;
  while (count>0)
    {
      digitalWrite(RED, HIGH);
      delay(300);
      digitalWrite(RED, LOW);
      delay(300);
      count -=1;
    }
}
```

```
/////////////////////////////////////////////////////////////////////
// wait for all buttons to release
void buttonWait()
 {
  while (digitalRead(STOP)==LOW || digitalRead(SAVE) == LOW ||
         digitalRead(SAMPLE)==LOW)
    delay(100);

    // debounce
    delay(250);
}

/////////////////////////////////////////////////////////////////
// turn off LEDs
void lightsOut()
{
  digitalWrite(GREEN0,LOW);
  digitalWrite(GREEN1,LOW);
  digitalWrite(RED,LOW);
}

//////////////////////////////////////////////////////////////////////
// Unix time conversions
unsigned unixDays(uint32_t unixSeconds)
{
  return unixSeconds / 86400L;
}

unsigned unixHours(uint32_t unixSeconds)
{
  return (unixSeconds % 86400L) / 3600;
}

unsigned unixMinutes(uint32_t unixSeconds)
{
  return ((unixSeconds % 86400L) % 3600) / 60;
}

unsigned unixSeconds(uint32_t unixSeconds)
{
  return ((unixSeconds % 86400L) % 3600) % 60;
}

////////////////////////////////////////////////////////////
// Interrupt service routine
void countIsr() {
    count += 1;

  if (count & 0x7)  // 1/8 duty cycle
     digitalWrite(BLUE, LOW);   // turn the LED off
    else
      digitalWrite(BLUE, HIGH);
}
```

```
// *************************************************************
//   MAIN BODY
// *************************************************************

/////////////////////////////////////////////////////////////////////////////////
// the loop function runs over and over again forever
void loop() {

  // check for "stop" when no sample in progress
  // (that's a delete operation)
  if (digitalRead(STOP)==LOW && (digitalRead(SAMPLE) == LOW ||
     digitalRead(SAVE)==LOW))
    {
      // In this situation, we will either delete one sample or all of them

      // Make sure we have something to delete
      if (sample > 0)
        {
         // signal delete acknowleged
         digitalWrite(RED, HIGH);

         // delete last sample?
         if (digitalRead(SAMPLE)==LOW)
            sample-=1;
        else
         // delete all samples
           sample=0;
         }

       buttonWait();
       lightsOut();

    }
```

```
// Check for sample trigger
if ((digitalRead(SAMPLE) == LOW) && (digitalRead(STOP) == HIGH) &&
  (sample<MAX_SAMPLES))
{
  unsigned short c1, c2;   // start and end counts

  // Remember full 32 bit time of first sample
  if (sample == 0)
    allUnixTime = rtc.now().unixtime();

  // record low order 16 bits of sample time
  samples[sample].lowUnixTime = rtc.now().unixtime() & 0xFFFF;

  // record three integrationss
  for (integration=0; integration<INTEGRATIONS; integration++)
    {
     // update sample count LEDs
     digitalWrite(GREEN0,(integration+1) & 0x1);
     digitalWrite(GREEN1,((integration+1) >> 1) & 0x1);

     // take integration
     c1 = count;                // counter start

     // one-second delay loops
     for (ii=0; ii<SECONDS; ii++)
     {
       delay(1000);   // 1 second

       // look for cancel
       if (digitalRead(STOP)==LOW && digitalRead(SAMPLE) == LOW)
         {
           // signal cancel acknowleged
           digitalWrite(RED, HIGH);

           buttonWait();

           // bogus integration count signals aborted run
           integration = INTEGRATIONS+1;
           break;  // exit SECONDS loop
         }
     }  // end SECONDS loop
```

```
      // get final count
      c2 = count;

      if (integration < INTEGRATIONS+1)
        {
          // watch for 16 bit wraparound
          if (c2 > c1)
            samples[sample].integrations[integration] = c2 - c1;
          else
            samples[sample].integrations[integration] = (long)c2 - (long)c1 +
                 65536;
        }

     } // end integration loop

     // sample succeeded if we got 3 integrations
     if (integration == INTEGRATIONS)
       sample++;

     lightsOut();
  }


// Check for file write trigger
if ((digitalRead(SAVE)==LOW) && (digitalRead(STOP) == HIGH) && (sample > 0))
  {
   // Write data
   File myFile;
   char filename[13];

   {
     String string;
     unsigned long days;
     unsigned char hours;
     unsigned char minutes;

     // construct MJD+UTC string DDDDDHMM (single digit hour= 0..9,A..M)
     days = unixDays(allUnixTime);
     hours = unixHours(allUnixTime);
     minutes = unixMinutes(allUnixTime);

     days += unixtimeJD - 2400000;

     // JD wraps at 12:00 UTC
     if (hours>=12)
       days += 1;

     string += days;
```

```
// if hour>9, use characters starting at A for 10
     if (hours>9)
       string+=char(0x41+hours-10);
     else
       string+=hours;

     // deal with leading 0 of minutes
     if (minutes<10)
       string+=char(0x30);

     string+=minutes;

     string += ".pho";

     string.toCharArray(filename,13);
   }

  // OPen the file
   myFile = SD.open(filename, FILE_WRITE);

   if (myFile)
      {
        unsigned short days;
        unsigned char hours;
        unsigned char minutes;
        unsigned char seconds;
        char space[2]=" ";

        // signal file open success
        digitalWrite(GREEN0,HIGH);
        digitalWrite(GREEN1,HIGH);
        delay(500);
```

```
        // Now write the data
        for (ii=0; ii<sample; ii++)
        {
          // check for wrap in low order of unix time
          if (samples[ii].lowUnixTime < (allUnixTime & 0xFFFF))
            wrap = 0x1;
          else
            wrap = 0;

          // get upper 16 bits from allUnixTime, lower 16 bits from sample
          hours = unixHours(samples[ii].lowUnixTime +
            (allUnixTime & 0xFFFF0000 ) + wrap*65536);

          minutes = unixMinutes(samples[ii].lowUnixTime +
            (allUnixTime & 0xFFFF0000 ) + wrap*65536);

          seconds = unixSeconds(samples[ii].lowUnixTime +
            (allUnixTime & 0xFFFF0000 ) + wrap*65536);

          // Format= "JD UTC_hour UTC_min UTC_sec integration0..integrationN"
          if (hours > 12)  // JD flip at 12:00 UTC
            myFile.print(unixDays(allUnixTime) + unixtimeJD + 1);
          else
            myFile.print(unixDays(allUnixTime) + unixtimeJD);
          myFile.print(space);
          myFile.print(hours);
          myFile.print(space);
          myFile.print(minutes);
          myFile.print(space);
          myFile.print(seconds);
          myFile.print(space);
          myFile.print(space);

          for (integration=0; integration<INTEGRATIONS; integration++)
          {
            myFile.print(samples[ii].integrations[integration], DEC);
            if (integration < 2)
              myFile.print(space);
          }
          myFile.println();
         }
         myFile.close();

       lightsOut();

       // We now have no samples
       sample = 0;
      }
     else
       flashem(5);
}

// flash the green Leds when maximum samples reached
if (sample == MAX_SAMPLES)
  {
    char bit = rtc.now().secondstime() & 0x1;
    digitalWrite(GREEN0,bit);
    digitalWrite(GREEN1,bit);
  }
}
```

Below is the code ("rtcinit") used to initialize the realtime clock. After the clock is set, it waits for *Stop* to perform the time check.

```
// Initialize real time clock
//  3 September 2020

#include <Wire.h>
#include "RTClib.h"


// input definitions
#define STOP   3
#define SAMPLE 4
#define SAVE   5

// LED definitions
#define GREEN0 6
#define GREEN1 7
#define BLUE   8
#define RED    9


RTC_PCF8523 rtc;
uint32_t theTime;
DateTime now;


///////////////////////////////////////////////////////////////////////
// Unix time conversions
unsigned unixDays(uint32_t unixSeconds)
{
  return unixSeconds / 86400L;
}

unsigned unixHours(uint32_t unixSeconds)
{
  return (unixSeconds % 86400L) / 3600;
}

unsigned unixMinutes(uint32_t unixSeconds)
{
  return ((unixSeconds % 86400L) % 3600) / 60;
}
```

```
///////////////////////////////////////////////////////////////
// flash Red light n times
void flashem(char n)
{
 char count=n;
  while (count>0)
    {
      digitalWrite(RED, HIGH);
      delay(300);
      digitalWrite(RED, LOW);
      delay(300);
      count -=1;
    }
}


void setup () {

  Serial.begin(9600);
#ifdef AVR
  Wire.begin();
#else
  Wire1.begin(); // Shield I2C pins connect to alt I2C bus on Arduino Due
#endif
  rtc.begin();


  // following line sets the RTC to the date & time this sketch was compiled...
  // plus 8 hours for UTC plus 9 seconds for init time minus 1 hour for DST
  #define DAYLIGHT 1
  rtc.adjust(DateTime(__DATE__, __TIME__) + TimeSpan(8*3600)
    - TimeSpan(DAYLIGHT*3600));


  // initialize input pins
  pinMode(STOP, INPUT_PULLUP);   // cancel
  pinMode(SAMPLE, INPUT_PULLUP);  // sampling trigger
  pinMode(SAVE, INPUT_PULLUP);  // output trigger

  // initialize LED pins
  // green lines are used for serial connection
  pinMode(BLUE, OUTPUT);
  pinMode(RED, OUTPUT);
  pinMode(GREEN0, OUTPUT);
  pinMode(GREEN1, OUTPUT);

   char bit = HIGH;
```

```
// Flash red while waiting for signal to start
while (digitalRead(STOP)==HIGH)
  {
    digitalWrite(RED, bit);
    delay(250);
    bit= bit ^ 0x1;
  }

  // Turn off RED, flash BLUE
  digitalWrite(RED, LOW);
  digitalWrite(BLUE, HIGH);

  // wait 1 second
  delay(1000);

  // get current UTC minute modulo 10
  char minutes;
  theTime = rtc.now().unixtime();
  minutes = unixMinutes(theTime);

  flashem(minutes/10);
  digitalWrite(BLUE, LOW);
  delay(500);
  digitalWrite(BLUE, HIGH);
  flashem(minutes%10);

  digitalWrite(BLUE, LOW);
}
```