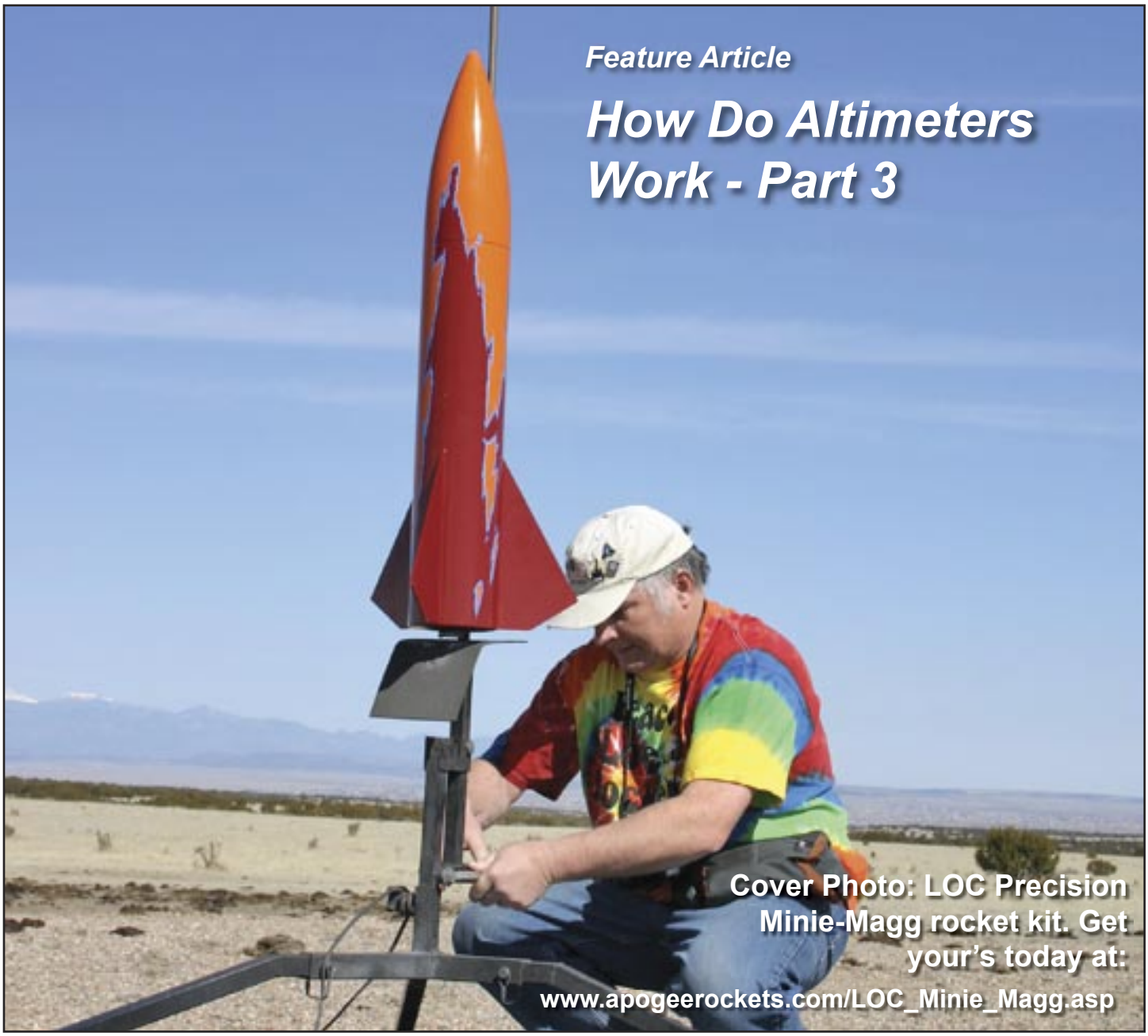
Apogee Components

PEAK OF FLIGHT

NEWSLETTER



Feature Article

How Do Altimeters Work - Part 3

Cover Photo: LOC Precision Minie-Magg rocket kit. Get your's today at: www.apogeerockets.com/LOC_Minie_Magg.asp

Apogee Components, Inc. — Your Source For Rocket Supplies That Will Take You To The "Peak-of-Flight"
3355 Fillmore Ridge Heights
Colorado Springs, Colorado 80907-9024 USA
www.ApogeeRockets.com e-mail: orders@apogeerockets.com

ISSUE 257 MARCH 23, 2010

How Do Electronic Altimeters Work - Part 3

By Norm Diedzic Jr.

{Ed: This is part three in the series of articles that explains how altimeters work. Part 1 was in Peak-of-Flight Newsletter #240 at www.ApogeeRockets.com/Education/Downloads/Newsletter240.pdf. Part 2 was in Newsletter #242 which can be downloaded at www.ApogeeRockets.com/Education/Downloads/Newsletter242.pdf. }

In our previous two installments, we have been slowly circling around the heart of the altimeter. This is where all the logic processing occurs. In our block diagram, we called this part of the altimeter the Computer or Processing unit. Actually more common names in industry for these would be *microprocessors, microcontrollers* or *embedded controllers*. Many times they simply are called MCU as an abbreviation for *MicroController Unit,* which we will use in this article.

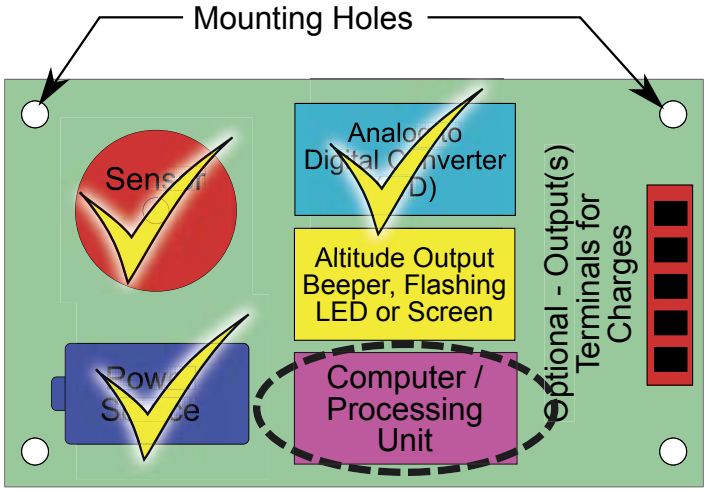


Figure 1: Altimeter block diagram.

So what makes an MCU different from the other devices we have been covering in the series? All of these devices are little black chips with pins sticking out that get soldered on a board of some sort. The differentiator with an MCU is that it can be programmed with software to perform a multitude of tasks. The other devices can be configured, but only perform one task such as voltage regulation or A/D conversion.

Like all of the other devices we have discussed, different MCU's have different features and capabilities. When reviewing an MCU, the features are very similar to looking for a personal computer or laptop. Just like shopping for a PC, the main considerations for the processor are bits, speed and memory. For a PC we look for 32 vs. 64 bits, gigahertz speeds and gigabytes of memory; with an MCU we generally look for 8 or 16 bits, kilohertz or megahertz speeds and kilobytes of memory.

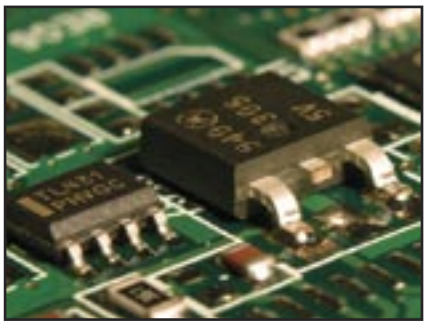


Figure 2: Chips on a board.

But while all PC processors must work with the same few mother board designs and operating systems, in the MCU arena there are no such restrictions, so you find much more variety in "extras" that may be included in an MCU. In fact, many microcontrollers include A/D and D/A converters right in the processor. In this way certain MCU devices become specialized to certain tasks.

Other things that you may need to consider with MCUs that you don't with PC processors are the number of input and output lines, communication ports available, power consumption, programming method and chip packaging.

The input and output lines get connected to things like jumpers for setting altimeter features, LED's for flashing out altitude and signals for firing ejection charges. Certain communication ports may be required to communicate with A/D converters, memory chips or displays. In summary, just like with all of the other devices discussed so far, with an MCU, there are a lot of options and the designer will spend a lot of time looking at different hardware to find the right mix of features and prices to meet their design criteria.

One nice thing most of the MCU manufacturers have done is package their controllers in what is known as a starter kit with a controller mounted on a board with some other devices for testing and evaluation. Generally, software, a manual and programming cable are also included. This is an easy way for the designer to test the MCU and see if it meets their needs.

Continued on page 3

About this Newsletter

You can subscribe to receive this e-zine FREE at the Apogee Components web site (www.ApogeeRockets.com), or by sending an e-mail to: ezine@apogeeRockets.com with "SUBSCRIBE" as the subject line of the message.

Newsletter Staff

Writer: Tim Van Milligan
Layout / Cover Artist: Tim Van Milligan
Proofreader: Michelle Mason

Continued from page 2

How Do Electronic Altimeters Work - 3

Figure 3: A MicroController Unit development kit comes with a lot of extras to make programming easier.

When it comes to programming, microcontrollers have come a long way from their roots. They used to be programmed almost exclusively in something called *assembly language.*

Well written assembly programs (also known as *code*) result in the fastest running systems possible. That is because each instruction in assembly code relates 1:1 to an actual MCU instruction. For this reason, assembly is also called a low level programming language. The instructions are very rudimentary but execute quickly. For instance, if you wanted to add three numbers together in assembly code, you would have to perform all of the following steps:

1. Fetch the first number into a register
2. Add the 2nd number to the first (the result of this operation stays in the register)
3. Add the 3rd number to the register.
4. Move the result to storage for use later.

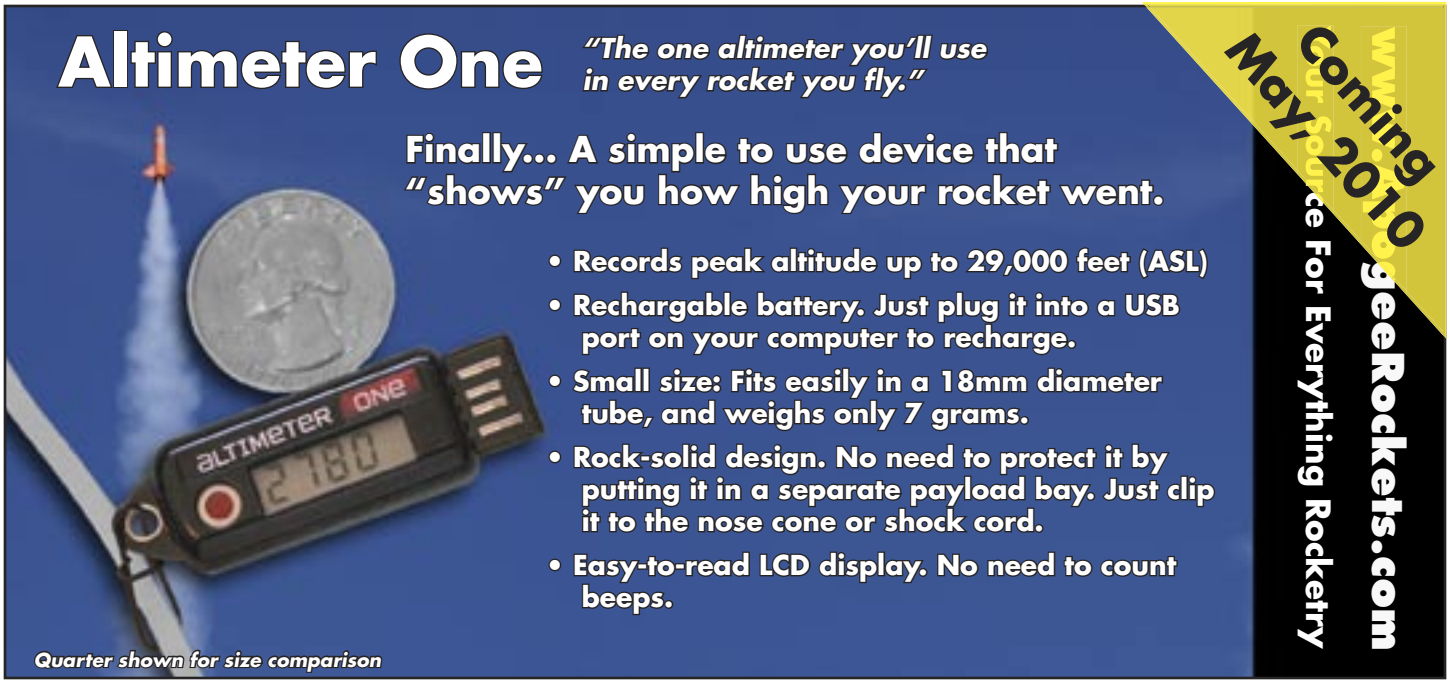
If our numbers are 10, 15, and 33, the assembly code might look like this:

```
MOVLW     0Ah
ADDLW0Fh
ADDLW21h
MOVWF     30h
```

Ouch – if you never saw assembly language before, that doesn't make much sense at all. Most assembly instructions are entered as abbreviations called mnemonics and most numbers are entered in a format called hexadecimal, so it's like everything is in double shorthand. A full, or even cursory depiction of assembly language is definitely beyond the scope of this article.

We also may need to see if the result of our math created a number too big for the MCU to handle. Remember when we discussed bits in the last article and how with 8 bits, we can have 256 states. If we are using an 8 bit controller and our sum gets larger than 256 we should

Continued on page 4



Continued from page 3

How Do Electronic Altimeters Work - 3

know about it. The MCU will turn on a special reserved bit in a place called the *carry flag* when that happens so our program can react accordingly.

Thankfully, as MCUs have increased in capabilities and speed, *higher level languages* are able to be used. The most common of these are BASIC and C. In either of these languages, our attempt to add three numbers together would look more like an equation you learned in grammar school: A = 10+15+33;

These are called high level languages because each statement in these languages corresponds to many instructions at the MCU hardware level. Contrast this with assembly language where each statement *is* an MCU hardware instruction. While assembly language has the potential to create faster executing programs, it also has a steeper learning curve and is harder to *debug* (the process of fixing programming errors). For each different architecture of MCU, there will be different assembly instructions to learn, while with C or Basic, the majority of the language is standardized so a programmer is more free to switch between MCU's and not have to spend a lot of time learning the new architecture.

Ok, enough about the MCU. Let's assume we have the hardware, we know the language and we want to make an altimeter out of this conglomeration of chips, solder and green board. We have a number that represents air pressure, but we need a way to turn that into an altitude. To do this, we need to learn a little more about the atmosphere.

For analysis purposes, the atmosphere is broken up into different zones, each with its own properties. One accepted definition of these zones and their properties is detailed in the *U.S. Standard Atmosphere 1976,* from the Government Printing Office. This 241 page tome tells you more than you will ever want to know about the atmosphere, and contains 165 pages of tables of altitudes, pressures, temperatures and a host of other data of interest to scientists.

Figure 4: U.S. Standard Atmosphere book.

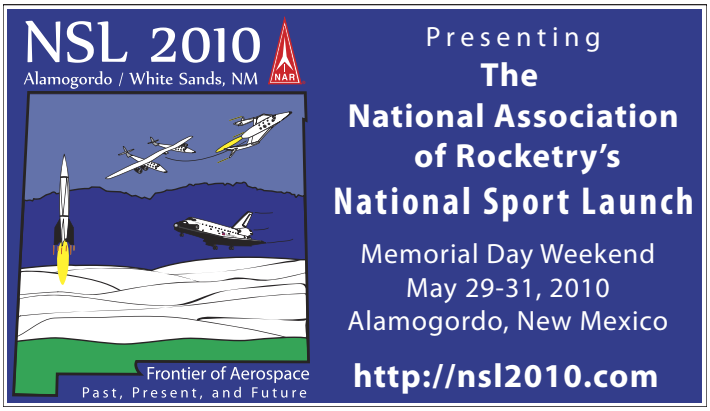
The zones are broken up based on how the temperature changes within each zone. The zone closest to earth, the *troposphere*, is what we are most interested in. This layer extends from the surface of the earth to 11,000m (about 6.8 miles) above sea level.

The way the temperature changes within a layer is called the *temperature gradient*. For the layer of air from sea level to 11km the temperature gradient is given as:

$$L = -6.5 \cdot \frac{K}{km'} \qquad [1]$$

The K in this equation is degrees Kelvin, which is a

Continued on page 5



Continued from page 4

How Do Electronic Altimeters Work - 3

temperature scale referenced to *absolute zero.* Absolute zero is the temperature at which all motion, even within molecules and atoms, stops. Standards are generally defined to absolute scales if they can be. So zero degrees Kelvin is as cold as it gets. To convert to Celsius, you have to add 273.15 to the Kelvin temperature. The size of a degree Kelvin is exactly the same as the size of a degree Celsius but the entire scale is just shifted down 273.15 degrees.

The km' in the equation is something called *geopotential* km. This is one of those little engineered properties that scientists use to make their equations nicer and deserves a little more explanation.

Remember back to when we stated the air above us is a column that is being pulled down to the earth by gravity? So in essence, pressure is mostly due to the gravity pulling down on the air particles, but as we increase our altitude, the pull of gravity is less and less. Trying to take the changing gravity into account complicates the equations we use to calculate pressure changes, so the scientists came up with an interesting work around. They made a new, non-linear height scale, the geopotential height, which changes as you get farther from the surface of the earth. The idea is that for any small change in geopotential height, the gravitational effect will be the same whether you are at sea level or some other altitude. The definition of the geopotential meter from the standard may help: "The unit of measurement of geopotential is the standard geopotential meter (m') which represents the work done by lifting a unit mass 1 geometric meter, through a region in which the acceleration of gravity is uniformly 9.80665 m/s2".

Using geopotential altitude in the equations has the double simplification of eliminating gravity changes and density from the equations.

So what equation do we use to relate pressure and altitude? It is called the *hydrostatic equation.* It has two forms depending on whether the temperature is changing in a layer. Since we are mostly concerned with the troposphere, the temperature is changing as given by L above, and the hydrostatic equation is:

$$\ln\left(\frac{P}{P_b}\right) = -\frac{G \cdot M}{R \cdot L} \ln\left(\frac{T_b + L \cdot (H - H_b)}{T_b}\right) \qquad [2]$$

With the following definitions:

ln: The natural logarithm function
P: the measured pressure
P_b: the pressure at the base of the layer (for us we use

Continued on page 6

Looking For A Fun Rocket Kit?

Roam In Our Forest of Over 190 Different Types

- Unique and exotic kits from over 20 different manufacturers
- Skill Levels range from "easy" to "fiendish"
- Sizes from 1/4A motor to level-2-high-power
- We build & fly them to find out what they're like, saving you grief
- More new ones arriving all the time
- Educational bulk packs available too

www.ApogeeRockets.com

www.ApogeeRockets.com/All_rocket_kits.asp

Continued from page 5

How Do Electronic Altimeters Work - 3

the *standard* pressure at sea level = 1 atmosphere)

G: Acceleration of gravity at sea level

M: Mean molecular weight of air

R: Perfect gas constant

L: Temperature gradient (defined above)

T_b: Temperature at the base of the layer (for us we use the *standard* temp. at sea level = 15°C (59°F)

H: The geopotential altitude

Hb: The geopotential altitude at the base of the layer (0 m in our case).

If we are measuring pressure with our altimeter, then the trick becomes to solve the equation for H which is the only unknown. I'll save you the step by step evaluation which is beyond the scope of this article and give you the equation for altitude based on pressure (sometimes called *pressure altitude*).

$$H = \frac{T_b \cdot e^{\left\{-\frac{R \cdot L}{G \cdot M} \cdot \ln\left[\frac{P}{P_b}\right]\right\}} - T_b}{L} \qquad [3]$$

Note also that this is the geopotential altitude, so we must also convert to the geometric altitude using the following equation:

$$Z = \frac{E \cdot H}{E - H} \qquad [4]$$

Where Z is the geometric altitude, H is the geopotential altitude and E is the radius of the earth (taken as 6357 km in the 1976 Standard).

For small model rocket altitudes, the difference between geopotential and geometric altitude is very small. You have to get to around 4560 ft. [1390 m] before the difference reaches 1ft. [30 cm] Although for mid and high power rockets, the difference becomes larger (by the time you get to 20,000 ft. [6096 m] the difference between geopotential and actual altitude is about 19 ft. [5.8 m]), it is still usually much smaller than the resolution of the altimeter which becomes more coarse the higher the flight. By the time a rocket reaches 20,000 ft. the accuracy of the altimeter may only be +/- 500 ft.

Even though many of the values in the pressure altitude equation are constants, it still includes exponential and natural logarithm functions. Not many MCU's will have these functions built in, let alone floating point math. If they do have these functions, they will take many CPU cycles to complete and may slow things down enough to affect the rocket's flight events or miss the peak altitude.

The very simplest way to handle this is to offload that work onto the user. An altimeter may beep out a number that relates to the pressure reading, and then the rocketeer looks that value up in the table of altitudes provided with the altimeter. While this makes it simpler for the programmer, the user might not want to have to keep track of a booklet of values, and if lost will not be very happy with the

Continued on page 7

Continued from page 6

How Do Electronic Altimeters Work - 3

manufacturer.

The MCU may be programmed save only raw pressure transducer readings during ascent. Then, only when the MCU has determined that the rocket is descending will it perform the time consuming task of converting the transducer reading into an altitude.

Another way to overcome a slow MCU is to use something called a *look up table*. As the name says, a table is generated of pressures and altitudes and stored in the permanent memory of the altimeter. As the pressure readings are made, they are compared to the table to find the altitude. Obviously the table cannot be infinitely large, so another technique called *interpolation* is used with the look up table to find the altitude between table entries. When a pressure is read, the table is scanned to find the pressures above and below the measurement. Then the interpolated altitude can be found with simple +,-,x and ÷ operations as follows:

$$A_n + \frac{P_{Measured} - P_n}{P_{n+1} - P_n} \cdot (A_{n+1} - A_n) \qquad [5]$$

The designer could also use an approximation function that uses simpler math to calculate the altitude

Looking closely at these equations, you might think that they are only valid at sea level on a day when the temperature at that sea level location is 15°C and the pressure at sea level is 1 atmosphere. After all, the equations are based on these values. Let's look first at how the altimeter actually uses the equations, and then we can do an analysis on how different initial conditions affect an altitude reading.

We have to make intelligent use of the data at hand to end up with a useful altimeter. Just like a successful launch will follow a set checklist, the well-designed altimeter will follow a well-planned procedure for sensing storing and reacting to the signals provided to it.

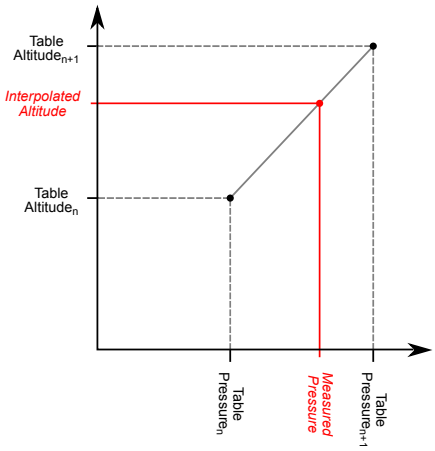


Figure 5: Interpolation between two points in the chart.

The best way to document these procedures is in something called a flow chart. Back in school a flow chart was required with most programming assignments and was usually done late at night after the real program code was hacked out over diet sodas and Snickers™ bars. Time has shown me that this is truly the wrong way to proceed, and that a well thought out flow chart actually should come first!

Figure 6 shows a very simplified flow chart for the steps a peak holding altimeter with drogue and main outputs might go through during a flight. The diagram shows the sequence of steps that are performed for starting up the altimeter, detecting the various stages of the flight and outputting the maximum attained altitude. Note that each of the drawn blocks may actually be its own very complicated algorithm.

For example, detecting the launch is not as simple as checking if the current pressure reading is less than the stored initial reading. There must some consideration for noise in the sensor readings as well as the situation where the rocket is not flown and is manually lifted off the launcher. You definitely don't want the altimeter to fire the drogue & main in that situation. Several methods can be used for

Continued on page 8

Continued from page 7

How Do Electronic Altimeters Work - 3

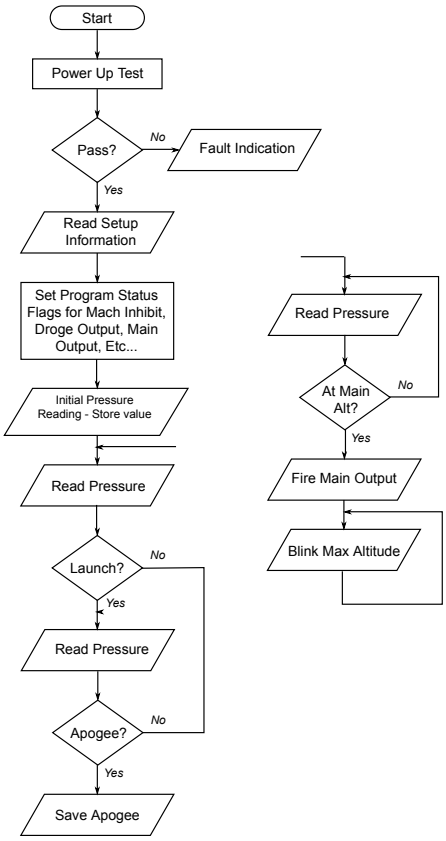


Figure 6: Simplified flow chart for writing the instructions.

this, such as waiting for a specific altitude to be reached or a certain number of decreasing pressure readings to be taken in a row before launch is detected.

The designer may also want to see a decrease in pressure over a short period of time and ignore changes that occur over longer timeframes. The longer time frame changes are more likely due to changes in local barometric pressure and not a rocket launch.

While the launch detection is the most critical of the sequences in an altimeter, each of the other tasks will have its own set of "gotcha" elements to contend with. Serious thought and very thorough testing must go into any new designs to make sure that they function reliably and predictably every time the altimeter is used.

Now let's move back to how the actual determination of altitude is accomplished and how changes in the local weather could affect the readings.

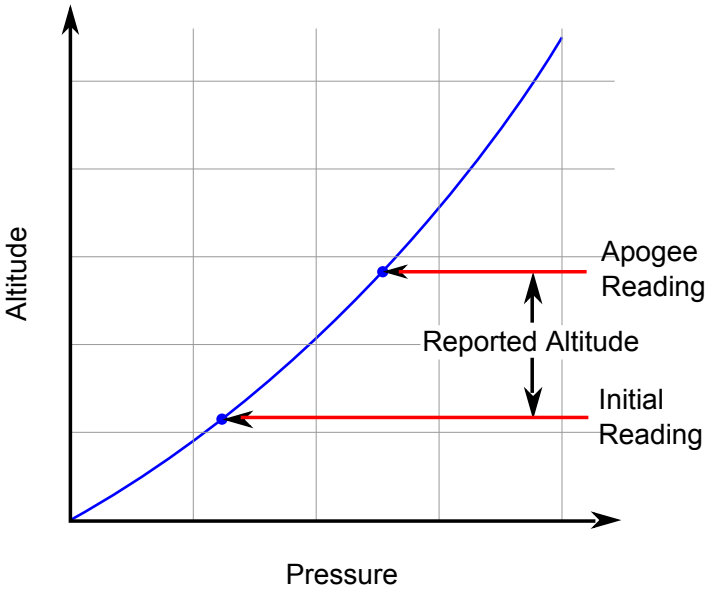


Figure 7: Altitude calculation from pressure readings.

The first thing to realize is that a barometric altimeter is always looking at the *difference* between two pressures, and is not trying to relate a specific pressure to an exact altitude. While the MCU may assign an altitude number to the initial reading, it will actually report to you the difference between the initial reading and the apogee reading (See Figure 7 Altitude Calculation). If this weren't the case then, as mentioned before, the only place an altimeter would be valid would be at the altitude and temperature the device was designed for.

As an example let's say the initial pressure measured is 98.354 kPa [14.265 psi]. According to the standard, this represents an altitude of 250 m above sea level. Then the apogee reading taken is 93.196 kPa [13.517 psi], which

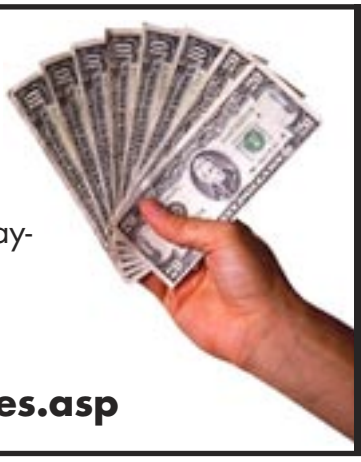
Continued on page 9

We're Paying Cash

For Great Articles for This Newsletter

Are you a writer looking for some serious pocket change? We're paying up to $350 for good how-to articles for this newsletter. If you're interested, see our submission guidelines on the Apogee web site.

www.ApogeeRockets.com/Newsletter_Guidelines.asp



www.ApogeeRockets.com

Continued from page 8

How Do Electronic Altimeters Work - 3

represents an altitude of 700 m above sea level. The reported flight altitude would be 700-250 or 450m [1476 ft].

Now we'll move on to the questions that have been sitting in the back of my mind for years and the impetus for me writing this entire altimeter explanation series. What happens when you use a barometric altimeter and your launch site is **not** at standard temperature and pressure?

On first glance, we're tempted to say that the changes in initial environment are cancelled out because we are subtracting the apogee pressure reading from the initial pressure reading. That would be true if the pressure vs. altitude plot was a straight line, but as you can see, it is a curve. So how do changes in initial conditions affect our measurements?

It is easiest to tackle temperature and pressure separately. First we'll look at what happens when the temperature is different from standard conditions but the pressure is at standard conditions. Luckily, we have the hydrostatic equation (equ. [3]) to help us with this analysis. By changing the T_b value and plotting the equation, we can see how a change in temperature changes the altitude reading.

Figure 8 shows a graph of pressure vs. geopotential altitude for the *standard* temperature of 15 °C as well as 0 °C and 30 °C. We see that the curve shifts up when the temperature is high and down when the temperature is low.

Let's do an analysis of three flights, all of which are flown from a site with an elevation of 250m Above Sea Level (ASL) and all of which have a true apogee of exactly 750m ASL. For all three flights, the actual altitude flown

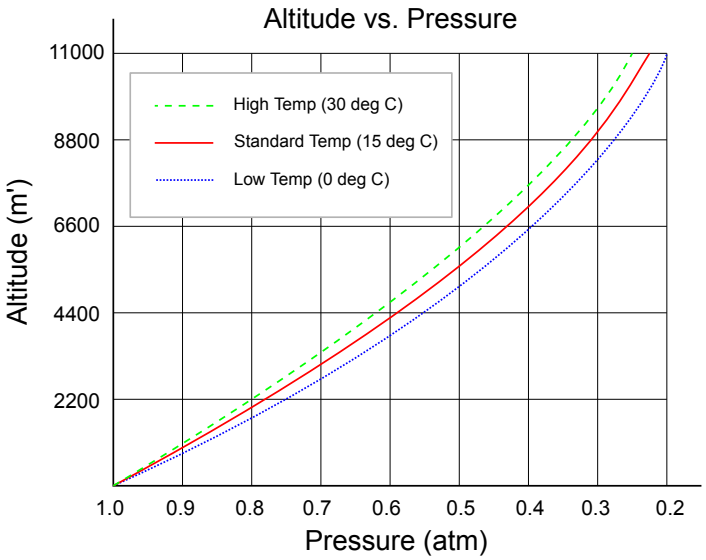


Figure 8: Temperature variation effects of pressure.

Continued on page 10

Continued from page 9

How Do Electronic Altimeters Work - 3

is 450m but one is flown on a standard day (15 °C at sea level), one on a cold day (0 °C at sea level), and one on a hot day (30 °C at sea level).

Figure 9 shows the pressures, calculated heights Above Sea Level (ASL) and calculated altitudes for the three conditions described above. For each case, the pressures were determined from the appropriate temperature curve in Figure 8. Then those pressures were used along with equations [3] and [4] to determine what altitude an altimeter would report for the measured pressures. Remember, the altimeter is only designed to use the equations from a standard atmosphere, so it uses the lower pressures from the cold day and higher pressures from the hot day as if they were measured on a *standard day.*

	Std. 15 °C	Cold 0 °C	Hot 30 °C
Initial Pressure (kPa)	98.361	98.196	98.503
Calc Height (m ASL)	250.0	263.8	237.6
Final Pressure (kPa)	93.205	92.763	93.584
Calc Height (m ASL)	700.0	738.5	665.3
Calculated Altitude (m)	450.0	474.7	427.7

Figure 9: Temperature Variations

So, as you can see in Figure 9, temperature has a very large effect on the altitude an altimeter will report. In this specific example, there is about 10% difference between the cold day altimeter readout vs. the hot day readout. Remember that in all three cases, the true altitude was 450m AGL.

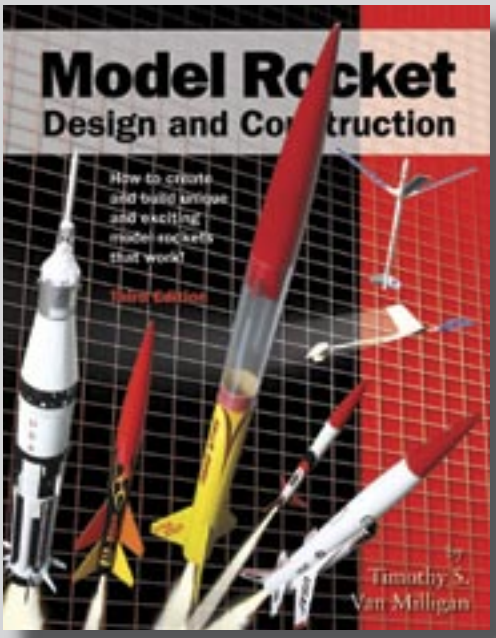
Figure 10 on the next page gives a visual interpretation of how temperature affects the readout of an altimeter. Each triangle has the same number of air molecules within it. As seen, cold air shrinks and hot air grows. This has the effect of moving the pressure at a certain altitude down and up, respectively. Again, the altimeter only knows about the standard atmosphere so pressures measured in non-standard conditions are mapped to the standard model by the altimeter.

For model and high power rocketry the big implication here is that you really cannot compare reported altitudes taken under different conditions even if the same altimeter is used. There is also the chance that, when you use a barometric altimeter to fire a main chute in very cold weather, it may fire at a lower than expected altitude.

For aviation, which still uses a lot of barometric altimeters, failure to take cold weather into account can be a deadly mistake. Pilots must make corrections to their altimeter readings on very cold days or they could run into obstacles (mountains) they thought they were high enough to clear.

On the pressure side, flying in a low pressure situation is like flying in the cold (altimeter reads high) and conversely, flying in high pressure is like flying on a hot day (altimeter reads low). To investigate the numbers, in equation [3] above, you would modify P_b by replacing the standard 101.35 kPa sea level pressure with the reported sea level pressure or measured pressure at the launch site adjusted

Continued on page 11



Model Rocket Design and Construction

By Timothy S. Van Milligan

New 3rd Edition Now Shipping!

This new 328 page guidebook for serious rocket designers contains the most up-to-date information on creating unique and exciting models that really work. With 566 illustrations and 175 photos, it is the ultimate resource if you want to make rockets that will push the edge of the performance envelope. Because of the number of pictures, it is also a great gift to give to beginners to start them on their rocketry future.

For more information, and to order this hefty book, visit the Apogee web site at: www.ApogeeRockets.com/design_book.asp

Apogee Components
3355 Fillmore Ridge Heights
Colorado Springs, Colorado 80907 USA

telephone: 719-535-9335
website: www.ApogeeRockets.com

APOGEE COMPONENTS

Continued from page 10

How Do Electronic Altimeters Work - 3

to sea level.

Although I'm not aware of any altimeters that take local conditions into account, it would be possible through jumper settings or perhaps a computer interface to enter the local temperature and elevation so that the MCU could correct its calculations for local conditions. There is also the possibility of using a temperature sensor with an additional analog input at the MCU, but the local elevation would still need to be entered to make the proper corrections.

Conclusion

So with even the air we fly in stacked against us, should we give up on barometric altimeters? To the contrary, despite all the possible sources of error and complexity of designing and creating an altimeter, they are useful and valuable tools for model and high power rocketeers. Even if the reported altitude is not exactly correct, these devices will still accurately detect apogee to fire a drogue chute and will be able to fire a main on the way down. Also, the temperature swing in the example represents the extremes of flying conditions, so generally errors from air conditions will not be as much as the 10% in the example.

This third and last installment brings us to the end of the road in describing and analyzing barometric altimeters, what they are made of, how they work their magic, and what limitations they are subject to. I have tried to make as much of the science and math accessible to as many readers as possible while remaining true to the subject.

If you have any comments or questions regarding this series, please feel free to drop me a line at njdjr@sbcglobal.net.

References

The Hydrostatic Equations Ralph L. Charmichael, January 23, 2003

http://www.grc.nasa.gov/WWW/K-12/airplane/atmos.html

http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19770009539_1977009539.pdf

http://www.tripoli.org/records/single.shtml

http://www.bluecoat.org/reports/Long_98_Cold.pdf

A Study Of Barometric Altimeter Errors In High Latitude Regions by Young Yee & Eric Yee

About The Author:

Norm Diedzic Jr. is an avid rocket flyer and a member of NIRA (Northern Illinois Rocketry Association; NAR Section 117). He has written several technical articles for their LAC winning newsletter, *The Leading Edge*, as well as a treatese on hemispherical parachute construction that appears in the book *Model Rocket Design and Construction*.

You may also recognize Norm as the inventor of a method of restraining small rocket motors in minimum diameter models, called the Dziedzic cut-out method (see *Model Rocket Design and Construction,* 3rd Edition, page 100)

He has an M.S. in Mechanical Engineering from University of Illinois at Chicago, and is currently the Engineering Manager at CMA/Flodyne/Hydradyne, an industrial distribution company serving Northern

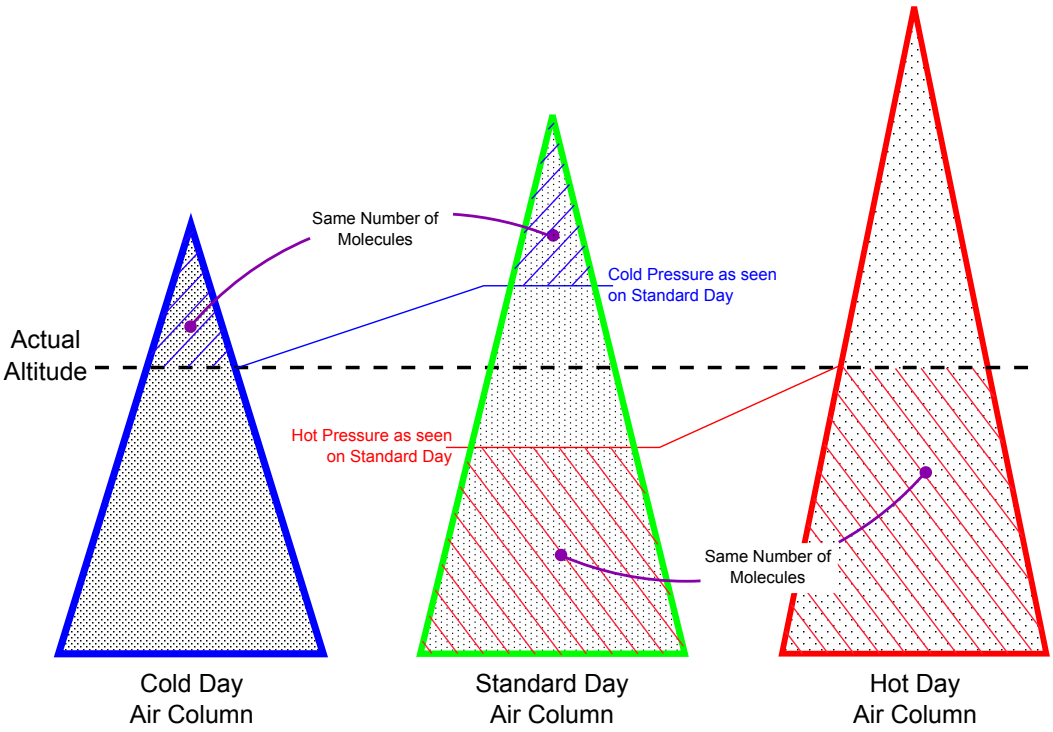


Figure 10: Temperature effects on altimeter readings.