

Approaching the Final Frontier

# Near Space

## Choosing the Best Data Loggers for Your Flight

### Data Loggers

There are several data loggers that are capable of fitting inside a lunch bag. By carrying one of these on the mission, you can collect additional science and engineering data. Since the data loggers collect data at a fixed rate (you program this rate into the data logger), you can relate the recorded data to the altitude of the near spacecraft.

Here are two commercially available data loggers to try: the HOBO and the ThermoChron. I have used the ThermoChron on several flights and purchased a HOBO for evaluation. I like my experiences with the ThermoChron and am impressed with my HOBO and plan to use it on my next flight.

### HOBO

Manufactured by Onset, the HOBO is an entire family of data loggers. The ones I describe here are used with the University of Colorado (Boulder, CO) BalloonSat program and are a little larger than a box of

matches. I weighed mine and it was only 0.9 ounces. I imagine my sensor would weigh more.

The least expensive HOBO is the H08-001-02, which is about \$59.00. This is a single channel HOBO that records just the temperature. It is not much larger than a box of matches, so you can tuck it anywhere inside the near space (NS) craft.

After recovering the mission, you download the temperature records from the HOBO and copy them into your spreadsheet into a new column. Align the temperature data with the MET data as closely as you can. This means that you need to program the HOBO using the time according to a GPS receiver.

Now you can create a chart of air temperature as a function of altitude. Because the air temperature in NS drops below the lower range of the HOBO (-4° F versus -60° F), it is best used to record temperatures inside the NS craft. However, you can still determine the altitude of the troposphere, because the interior temperature of the NS craft tracks the outside air temperature.

Another HOBO data logger is the H08-003-02 (about \$85.00). This one records the air temperature and relative humidity. It makes more sense to record the relative humidity of the air outside of the NS craft than inside, so place the HOBO outside the NS craft and you'll have to live with the bottoming out of the air temperature data.

If you feel comfortable building your own sensors, then consider purchasing the H08-006-04, which costs about \$95.00. This is a four

external channel data logger. With it, you can record either current or voltage from four separate sensors. For example, CU Boulder, CO adds solar cells to their BalloonSats. These do not provide power for the BalloonSat, instead, the HOBO records the current generated by the solar panel during the mission.

As the light intensity increases in NS, the solar cells produce more current. The voltage and current limits of the HOBO are 2.5 volts and 20 mA. You can use sensors that create more than 20 mA or 2.5 V if you use a current or voltage divider in your sensor design.

HOBOs are programmed with the Box Car program (P/N BC3.7-ON, \$14.00), which is purchased separately from the HOBO. I ran a test on my HOBO for this article. Here's what I discovered programming it. After installing the program, it created an icon in Onset Applications. A HOBO is connected to the comm port of a PC through a 1/8" jack. Your copy of Boxcar comes with this adapter cable.

Start the application and begin programming your HOBO by clicking **LOGGER**, then **LAUNCH**. First, look at the Battery Level gauge on the right side of the window. Don't launch a HOBO if the battery is about to die.

Give the deployment a name in the **DESCRIPTION** window. I recommend using the name of the flight. Select an interval. This is the time between measurements.

Measurements could be taken as often as every 1/2 second to as long as every nine hours (there is an

The HOBO. Photo courtesy of Onset.



option to create your own interval).

Each measurement of the HOBO requires one byte of memory. There is enough memory in the HOBO to record for hours or even days (this is called the duration). Here's a listing of selected intervals and durations.

1 sec	2 hr, 15 min
2 sec	4 hr, 30 min
10 sec	22 hr
15 sec	1 day, 9 hr
30 sec	2 days, 19 hr
60 sec	5 days, 15 hr

You can easily record data every two seconds for a NS mission, without running out of memory. However, if you do so, you'll have a lot of data to import to your spreadsheet. For those records that don't align with Tiny Trak posits, you'll have to interpolate the altitude of the measurement. This can get tedious if most of your data requires it. One option is to record measurements frequently, but to only copy measurements that align with the time stamps of the Tiny Trak. (For more on processing Tiny Trak posits, see "Near Space" in the May 2004 issue of *Nuts & Volts*.)

Select Advanced Options and make sure the wrap-around option is not clicked. If it turns out that your NS craft can't be recovered for a day or more, you do not want your flight data overwritten with measurements taken on the ground. Doing so just wastes the time, money, and effort you put into the mission.

It's best if you program the HOBO the night before launch. You can instruct the HOBO to delay recording measurements until a specified time. To do so, click on Delayed Start and then enter the date and time you want the HOBO to begin recording measurements. Be sure your PC clock is set to GPS time, as I believe the HOBO gets the current time from the PC it is being programmed on. This also lets you correlate HOBO records with Tiny Trak time stamps. The date field has the format of month/day/year.

Finally, you can select to Enable

or Disable channels. Disabling used channels creates less data for you to import and increases the duration the HOBO can record data. When you are finished, click on the Start button.

After the mission, connect the HOBO to your PC and offload its data. To do this, click Logger, then Readout. Data from the HOBO will be offloaded. The results are stored as a file on your PC. Give the file a meaningful name so you can find it again later (again, I'd recommend naming it after the mission).

Boxcar can only display one channel of data at a time. To change channels, click View, Display Options, then Channel. Next, select the channel you want to look at. The results are displayed in a graph. If you're happy with the data, then export it to a test file or Excel spreadsheet. Click on File, then Export. From there, select either Microsoft Excel, Lotus 123, or Custom. By selecting Custom, you can export the data to a text file for editing before moving it to a spreadsheet.

Under Custom, I recommend the following settings. Under the Time/Settings, make the Date Format read "no date," as the mission occurred on a single day. The GPS does not

indicate fractions of a second; its time is recorded in whole seconds. Therefore, it is only necessary to make the Time Format, Hr:Min:Sec.

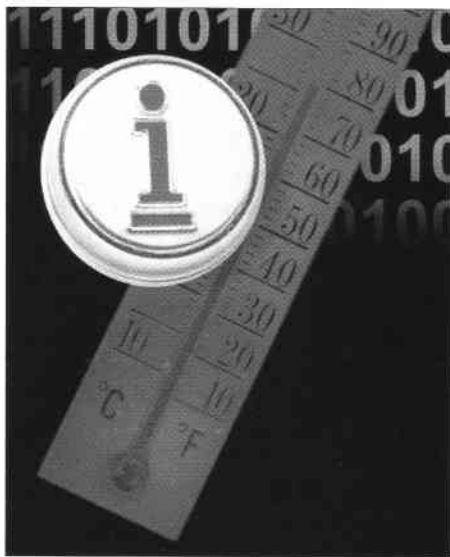
Under Data Settings, select a Data Separator of Comma and select the units (channels) that were used on the mission. Now click the Export button. The resulting file contains data looking like this:

```
Time,Voltage (V) (*1),Voltage (V)
(*2),Voltage (V) (*3),Voltage (V)
(*4)
```

```
19:25:00,0.874,0.659,0.366,0.288
19:25:02,0.132,0.122,0.103,0.103
19:25:04,0.073,0.073,0.073,0.083
19:25:06,0.063,0.073,0.073,0.073
19:25:08,0.063,0.073,0.073,0.073
19:25:10,0.142,0.142,0.063,0.073
```

Open a text editor and copy the data that you want to keep into a spreadsheet. Once that data is formatted correctly, you can Copy and Paste the data from the new spreadsheet into the Tiny Trak posit spreadsheet.

After talking with Onset, I found out that they also make a pressure data logger that is good up to an altitude of 32,000 meters. This HOBO



The Thermochron. Photo courtesy of Maxim IC/Dallas Semiconductor.

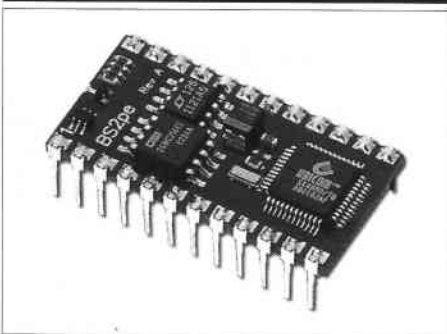
is model number HPA and costs \$249.00. You'll find Onset at [www.onsetcomp.com](http://www.onsetcomp.com). Their website contains several other sensors suitable for the HOBO.

## Thermochron

Another data logger is the Thermochron. Maxim IC purchased Dallas Semiconductor and currently sells their line of one wire devices. The Thermochron is a sealed, stainless steel device containing a clock, temperature sensor, memory, and battery. It's tiny — only about the size of five stacked dimes.

A Thermochron is programmed just like a HOBO. After installing the software, select iButton Viewer under the iButton-TMEX group. Ignore the

The BS2pe.  
Photo courtesy of Parallax, Inc.



Format Window and go straight to the Thermochron Viewer. Select the Wizard Tab. Click the NEXT button and set the time in the Thermochron. Select to set the Thermochron to the PC's clock (be sure to set your PC to the time on a GPS receiver). The funny thing is that I found the Thermochron's time to be more accurate than my laptop's clock.

Click NEXT and skip setting an alarm. Click NEXT and set the Mission Start Delay. With the Thermochron, you must do a little math, as you set the delay in days, hours, and minutes from the time that you are doing the programming. You don't set the time to begin the mission. Do your math carefully and be sure to begin recording data before the expected launch time.

Click NEXT and set the sample rate. The shortest sample rate is once a minute. I find this is adequate for my missions. Click NEXT and do not set Temperature Alarms. Click NEXT and *do not* Enable Roll-Over. Click NEXT once more and then FINISH. You can now remove the Thermochron from its reader and load it onboard the NS craft.

After recovery, use the same software to download data from the Thermochron by clicking on the Mission Results tab. Here, click on the Read Data Button. After the data is downloaded, you can either generate a graph or export the results. Click either the Quick Graph button or the Export Result button.

After clicking the Export Results button, you'll be asked to give the resulting file a name. Again, use a meaningful name so you can find the file later. Be sure you stop the mission after you download the Thermochron's data, as there's no need to use the internal battery to collect more data when you don't need it. That's it. The results are similar to those of the HOBO.

Output from the Thermochron contains data not needed by the spreadsheet. Delete this information in a text editor. The resulting data looks like this:

## Log Data

Format: [Time/Date , Temperature] (Fahrenheit)

```
08/03/2003 05:54 , 73.4°F
08/03/2003 05:55 , 73.4°F
08/03/2003 05:56 , 73.4°F
08/03/2003 05:57 , 73.4°F
08/03/2003 05:58 , 73.4°F
08/03/2003 05:59 , 73.4°F
```

In WordPad, you'll want to delete all instances of the date. Then change the colon in the time file to a comma. You can also remove the extra space around the comma between the time and temperature by typing " , " in the Find What field and typing " ," in the Replace With field.

Be sure to purchase a fob to hold the Thermochron, as it can easily get lost inside an NS craft. Also, since it's made from metal, place the Thermochron where it cannot short out the battery inside the module. It's a very bad sign when you see the NS craft climb out trailing a line of smoke.

Thermochrons are the ideal data logger if you want to compare the temperatures between objects of different colors or constructions. Each Thermochron has a unique ID printed on the case and this ID is displayed along with the measurements of Thermochron. Be sure to record which Thermochron ID went into which object being tested. You can purchase a Thermochron Starter Kit from Dallas-Maxim. The starter kit comes with the programming cable, a Thermochron, and memory. The programming software is free. The part number is DS1921K and costs \$30.00. You will find information on the Thermochron starter kit at the Dallas/Maxim website: [www.maxim-ic.com](http://www.maxim-ic.com)

## Hitchhiker

More advanced users will want to design their own data loggers. The benefit here is that you can customize your data logger to fit any sensor you can design. After a visit to the Parallax office in December 2003, I developed an idea based

upon input from Parallax's Ken Gracey. I'll develop this idea further and I will call them Hitchhikers.

The BASIC Stamp 2pe contains an additional 16 kb of EEPROM over the BS2p. It contains the same amount of scratch pad RAM as the BS2p, which allows you to record an entire GPS sentence for parsing at a later time. The BS2pe was designed with data loggers in mind. By purchasing a BS2pe and Board of Education (BoE), you can whip up a data logger that can be reconfigured mechanically and logically for each mission. The simplest Hitchhiker is programmed to record data at a fixed time interval. It requires a Push To Initiate button to tell it when to start recording. It is pushed just before the NS craft is released. This way, the time that data collection starts is known and no memory is wasted collecting data before launch.

The more advanced Hitchhiker shares the GPS output of the NS craft. Now, the altitude of the mission is recorded along with the results of experiments. Parallax sells a wide variety of App Mods for their BoE. Best of all, the code needed to integrate the App Mod into the BoE is available on their website. This dramatically reduces the time required to get a Hitchhiker ready for an NS mission.

One example of an appropriate App Mod is the SHT1X, a combined temperature and humidity sensor. The SHT1X can be mounted directly to the BoE or you can solder a cable to its pins before plugging it into the BoE. Using a cable allows the BoE to remain well inside the NS craft while letting the SHT1X sample the air outside.

More than App Mods are available. The texts for the BoE give instructions for creating several other sensors. I created a PCB for some of my past missions that is a light sensor based on LEDs. This lets one of my missions measure how sky brightness changed in blue and violet/near UV as the altitude increased. If you teach the Parallax microcontroller curriculum in a classroom, perhaps you can find a local

amateur NS group that can fly a class project into NS. It's guaranteed to be easier and cheaper than getting a sounding rocket flight. This concept is new, so I have yet to fully develop it. Keep reading *Nuts & Volts* for developments. **NV**