THE UNIVERSITY OF BRITISH COLUMBIA
Department of Physics and Astronomy
6224 Agricultural Roadl, Vancouver, BC, V3Y 2B3, Canada . Tel: 604-822-3853, Fax: 604-822-5324

LIDAR data format and software

Doc Nr: UBC.PHAS.LIDAR.005
Date: 2012-10-01

Researcher Approval:	Paul Hickson	2012-10-01	
UBC Approval:			
	Name	Date	Signature

Authors

Name	Affiliation
Paul Hickson	University of British Columbia
Thomas Pfrommer	University of British Columbia

Change record

Issue	Date	Section / Paragraph affected	Reason / Initiation Documents / Remarks
1	2010-07-27	1, 2	First draft of unified raw data specification
2	2011-05-02	3	Added section describing provided programs
3	2012-10-01	all	Public version

Contents

1. Introduction

The Large Zenith Telescope (LZT) sodium lidar system of the University of British Columbia (UBC) is used to study mesospheric dynamics and sodium density evolution, to provide input for performance modeling and design of laser adaptive optics (AO) for the next generation of large ground-based telescopes. The facility operates on clear nights, generating 2 MB of raw data per second (7.2 GB per hour).

A set of software tools is being developed to facilitate the processing of data from the facility. This document defines and explains the data formats employed, and the functionality and usage of the software.

2. Data format

Two different systems of counting electroncs and data acquisition software have been employed at the lidar facility. From the start of operations in June 2008, until November 2009, a four-channel counting system manufactured by Fast Comtec was used. While this system was the best available at the time, its data-transfer speed was limited, resulting in occasional data loss, and substantial periodic dead time while data were transfered from the counting card to the disk drive. The Fast Comtec system was replaced by a custom-made ultar-fast counting system manufactured by Sigma Space. This new system is capable of continuous data acquistion at GHz data rates.

The raw data format from these two counting systems are quite different, but both produce many files per night (hundreds for the old system). To facilitate data analysis and distribution, we have transformed all data sets to a compressed “unified” format described below.

2.1. Unified format

The unified data format is based on the FITS (Flexible Image Transport System) standard. This consists of a series of 2880-byte data blocks. The first block in a sequence is a header block containing format information in the form of an ASCII character sequence of 80 character sequences encoding keywords and values. Zero or more additional header blocks may follow, then any number of blocks of binary data. The series of header then data blocks can be repeated as many times as needed within a single FITS file.

The raw data produced by the two counting systems contain four (old system) or one (new system) bytes for every data bin for each of four separate channels. Thus there are as many as 16 bytes per altitude bin. The number of bins is ~ 10,000, and typically most are empty. This gives between ~40,000 and ~160,000 bytes for every laser shot. There are 50 shots per second, hence 2 MB/s for the new system and 8 MB/s for the old.

Since the lidar system typically detects at most several thousand photons per laser shot, a more efficient way to store the data is to tag individual photon events. In two bytes, it is possible to encode both the channel number (2 bits) and the bin number (14 bits - up to 16,384 bins). With this approach, the data rate drops to ~2000 bytes per shot, or 100 KB/s. This gives a compression ratio of 20 to 80 times.

The actual number of photons received in each shot is encoded in the FITS headers. This is allows maximum compactness and the most efficient use of space and processing time, but requires somewhat more sophisticated software to encode and decode the data.

The resulting format is summarized in Table 1. Every 50 shots of data (approximately 1.00 s of time) is preceeded by a single 2880-byte header block. The header encodes the time of the first shot in the sequence, and other information such as the bin size and altitude range. It also encodes 50 integers that give the number of photons detected for each of the 50 shots. The processing software reads these numbers and assigns the photons to the appropriate shot.

Table 1. Format of FITS header block for each set of 50 laser shots

Keyword	Value	Explanation
SIMPLE	'T'	Required for all FITS files
BITPIX	16	Two bytes per data value
NAXIS	1	One data dimension
NAXIS1	integer	total number of photons for these 50 shots
DATE	string	UTC Date of observation
TIME	string	UTC time of first shot
ALT	float	Altitude of telescope (m)
LAT	float	Latitude of telescope (deg N)
LONG	float	Longitude of telescope (deg E)
RECVR	string	Reciever and counting system description
CONFIG	string	Beam configuration (single, chopped, split, etc)
BLOC	string	Location of first shot (in asterism)
BANGLE	float	Beam chopping/splitting angle (arcmin)
POWER	float	Laser power (W) (if available)
WAVE	float	Wavelength (nm)
NBIN0	integer	Number of bins in the first set
BSIZE0	integer	Size of first set of bins (ns)
BSTART0	integer	Time for first bin in the first set (ns)
NBIN1	integer	Number of bins in the second set
BSIZE1	integer	Size of second set of bins (ns)
BSTART1	integer	Time for first bin in the second set (ns)
COUNTS0	string of 5 integers	Number of photons in shots 0-4
COUNTS1	string of 5 integers	Number of photons in shots 1-9
COUNTS2	string of 5 integers	Number of photons in shots 10-14
COUNTS3	string of 5 integers	Number of photons in shots 15-19
COUNTS4	string of 5 integers	Number of photons in shots 20-24
COUNTS5	string of 5 integers	Number of photons in shots 25-29
COUNTS6	string of 5 integers	Number of photons in shots 30-34
COUNTS7	string of 5 integers	Number of photons in shots 35-39
COUNTS8	string of 5 integers	Number of photons in shots 40-44
COUNTS9	string of 5 integers	Number of photons in shots 45-49
END		End of header data

For an average count rate of ~1000 photons per shot, there will be ~100,000 data bytes per second. The actual number must be padded to a multiple of 2880, which results in ~ 35 blocks. Including the header gives a typical data size of 36 blocks (~ 101 kB) per second. Thus the unified format will require ~ 360 MB per hour of data under typical conditions. No compression is necessary as the format is already highly compact.

3. Software

Several programs are provided to aid in data analysis. For speed and efficiency in handling large data files, these programs are written in the C language. This section provides a guide to the programs and their use.

To compile the programs, copy the files to a directory, go to this directory and use following commands (on Unix systems):

```
cc -o centroid centroid.c -lm
cc -o density density.c -lm
```

```
cc -o fileinfo fileinfo.c -lm
```

This creates the binary executable files ‘centroid’, ‘density’, and ‘fileinfo’. These files can then be moved to a standard directory (such as /usr/local/bin). Be sure that the directory containing the executable files is in your ‘path’ environment variable. They can then be executed by typing their name on the command line, followed by optional then required parameters. Examples are given below.

3.1. centroid.c

The ‘centroid’ program is used to generate a time series of the mean sodium altitude. This altitude is computed for every laser shot (typically 50 per second) and written to a text file. Each line of the output file corresponds to one shot, in time sequence. Each line contains four text fields that correspond to:

1. the UT time in seconds
2. the number of (sodium) photons detected
3. the centroid altitude in metres
4. the standard error of the centroid altitude.

By entering options on the command line, the user can specify the starting time and length of the time series. The default is the entire night.

Typeing ‘centroid -h’ or ‘centroid -help’ produces the following output:

```
NAME
      centroid.c - compute sodium centroid altitude

DESCRIPTION
      'centroid' reads lidar data FITS files for the requested night and outputs
      time, total count, centroid altitude and its standard error for each shot.

USAGE
      centroid [options] [directory/]date > output_file

PARAMETERS
      'file' is the date to be processed, in the form 'yyyymmdd'. If the data
      files for this night are not in the current directory, 'files' must
      be preceeded by a path to the directory that contains the data files
      (eg. /lidar/data/20090924 ). It is not necessary to add the extension
      _00.fits, _01.fits, etc. The program will find all FITS data files
      for the indicated date in the specified directory.

OPTIONS
      -h, -help        print help for this command
      -l    length     print at most 'length' hours, from start_time [24]
      -s    start_time start at 'start_time' (hours UTC) [0]
      -t               don't print a table header

EXAMPLES
      centroid /data/lidar/fits/20091008 > 20091008_cen.txt
       - computes a centroid time series using data from all fits files for the
         UT date 2009-10-08 (ie. files having names of the form '20091008_XX.fits'
         located in the directory '/data/lidar/fits/') using the default
```

```
          parameters (shown in square brackets above under OPTIONS) and writes the
          result to the text file '20091008_cen.txt'.

       centroid -s 12.0 -s 2 20091008 > 20091008_cen.txt
        - computes a 2 hour centroid time series, starting at 12:00 UTC, for
          UT 2009-10-08, assuming that the fits files are located in the current
          directory. The result is written to the text file '20091008_cen.txt'.

REVISION
       110502
```

3.2. density.c

The ‘density’ program is used to extract photon counts in temporal and spatial bins. By entering options on the command line, the user can specify the binning time interval and the number, size and starting altitude of the (vertical) spatial bins.

Typeing ‘density -h’ or ‘density -help’ produces the following output:

```
NAME
       density.c - generate a binned density map for a specified night

DESCRIPTION
       'density' reads the specified data file and prints an ASCII listing of
       photon counts binned to a specified resolution. Since the data are
       dominated by photon noise, the standard error for the count in each bin
       is the square root of the count. Each line of output data has the following
       format:

       UT_seconds   count[0] count[1] count[2] .... count[nbins-1]

USAGE
       density [options] [directory/]date > output_file

PARAMETERS
       'file' is the date to be processed, in the form 'yyyymmdd'. If the data
       files for this night are not in the current directory, 'files' must
       be preceeded by a path to the directory that contains the data files
       (eg. /lidar/data/20090924 ). It is not necessary to add the extension
       _00.fits, _01.fits, etc. The program will find all FITS data files
       for the indicated date in the specified directory.

OPTIONS
       -a    asize    size of altitude bin (m) [90]
       -h, -help      print help for this command
       -l    lower    altitude of lowest bin (m) [80000]
       -n    nbins    number of altitude bins [278]
       -t    tsize    size of time bin (s) [1]
       -v    level    level of verbosity [0]
```

```
EXAMPLES
      density /data/lidar/fits/20091008 > 20091008_density.txt
       - computes a density map using data from all fits files for the UT date
         2009-10-08 (ie. files having names of the form '20091008_XX.fits' located
         in the directory '/data/lidar/fits/') using the default parameters
         (shown in square brackets above under OPTIONS) and writes the result
         to the text file '20091008_density.txt'.

      density -a 100 -n 200 -t 10 20091008 > 20091008_100_200_20.txt
       - computes a density map for UT 2009-10-08, assuming that the fits files
         are located in the current directory, using 200 altitude bins of width
         100 m and 10 sec time bins. The starting altitude is the 80 km default.
         The result is written to the text file '20091008_100_200_20.txt'.

REVISION
      110403 P. Hickson
```

3.3. fileinfo.c

The 'fileinfo' program prints information about lidar FITS files. Specifically, it prints a list, one line for each file, containing the following information:

1. the file name
2. the UTC time of the start of the data
3. the UTC time of the end of the data
4. the number of hours of data recorded
5. the mean number of photons per second received from the sodium layer

To use this command, type 'fileinfo' followed by a list of FITS files. The list can contain wildcards such as 'fileinfo *.fits' (in Unix).

Typeing 'fileinfo -h' or 'fileinfo -help' produces the following output:

```
NAME
      fileinfo.c - list statistics for files

DESCRIPTION
      'fileinfo' reads the listed lidar data FITS files and prints the start
      time, end time, number of hours of data, and the mean number of photon
      counts per laser shot.

USAGE
      fileinfo [options] files...

PARAMETERS
      files...          a list of lidar FITS files (separated by white space)

OPTIONS
      -h, -help         print help for this command
      -t                don't print a table header
```

```
EXAMPLES
      fileinfo /data/lidar/fits/*.fits
       - print file information for all FITS files in the '/data/lidar/fits/'
         directory

      fileinfo 2010*.fits
       - print file information for all fits files in the current directory
         having names that start with '2010'.

REVISION
      110502
```