SREAL File Converter

Overview

This application note describes how to operate the SREAL (BDS[1]) File Converter. This document also describes the format of the BDS files produced by the CSA, and how the converter actually converts the BDS files into CSV[2] files.

Content

- Installing the SREAL (BDS) File Converter
- Using the SREAL (BDS) File Converter
- Understanding how the File Converter operates

Installing the SREAL (BDS) File Converter

There are two ways to install the file converter. One installation method involves the use of the dBm Optics CD, and the other involves installing the application from a ZIP file or Executable.

From the dBm Optics CD

To install the file converter from the dBm Optics CD you must first insert the CD into your CD-ROM or DVD-ROM drive. If the Welcome Utility does not automatically appear, then it can be accessed by opening "My Computer" and double-clicking on your CD-ROM/DVD-ROM drive letter. Once the Welcome Utility is open select "Install/Uninstall dBm Applications," then select "Install/Uninstall the BDS File Converter." The file converter setup utility will then load, simply follow the on-screen instructions in order to install the converter.

From a ZIP file

To install the file converter from a ZIP file you must first install WinZip (or equivalent) onto your computer. WinZip can be obtained from http://www.winzip.com/, the WinZip Evaluation Edition is entirely free and does not expire. After WinZip has been installed, the icon of the ZIP file will change into a cabinet being crunched, simply double-click on the file to open it in WinZip. Once WinZip has loaded, click the "Install" button to install the file converter. The setup utility will then load, simply follow the on-screen instructions in order to install the converter.

From an EXE file

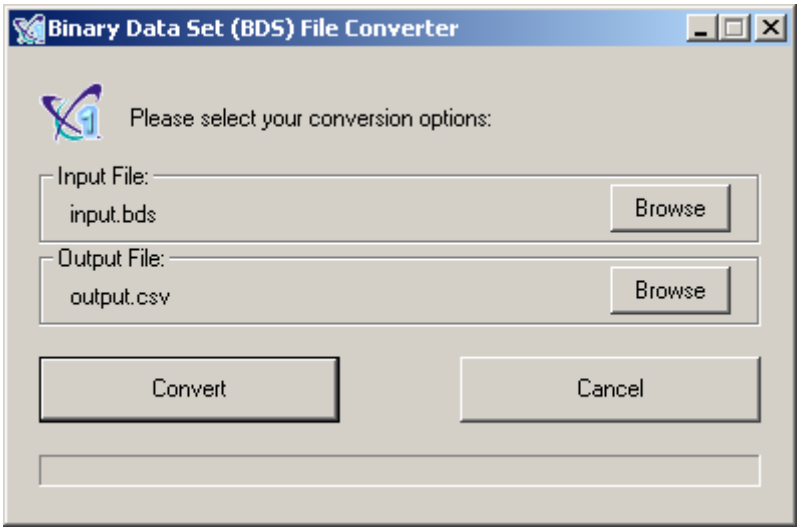
To install the file converter from an EXE file, simply double-click on the EXE file to start the setup utility. Once the setup utility has loaded, follow the on-screen instructions to install the converter.

---

[1] BDS: Binary Data Set. A Binary Data Set is a set of binary data created in a pre-determined format, this data must be converted into ASCII before it can be viewed with a program that reads ASCII documents (such as Microsoft® Notepad or Excel).

[2] CSV: Comma Separated Value. A Comma Separated Value file is commonly generated for use with Microsoft® Excel. Also known as "comma-delimited" files, these documents store data in a format that represents a table. Each comma character (,) indicates a change in the row (horizontal axis) and each new line indicates a change in column (vertical axis).

Using the SREAL (BDS) File Converter

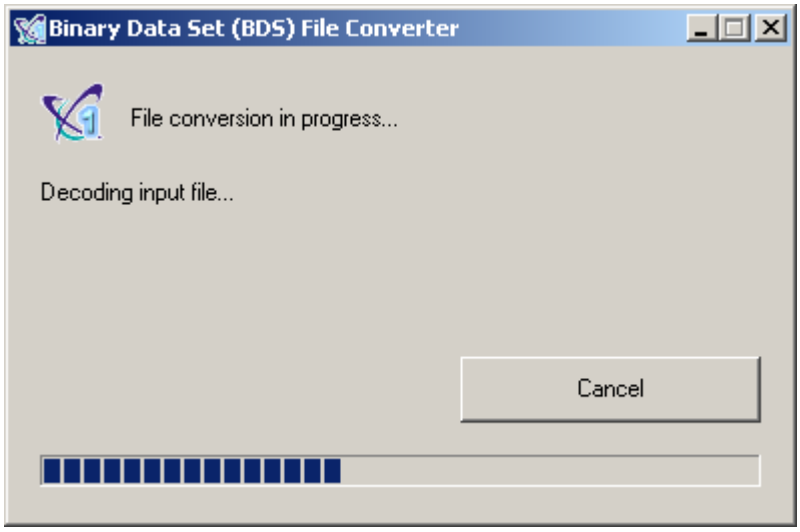


Input File
The BDS file that you would like to convert to a CSV file. Click the "Browse" button choose the correct file.

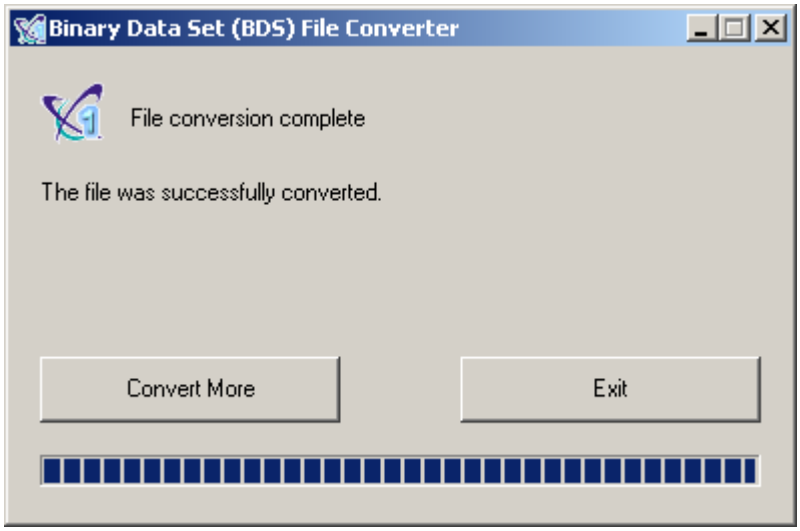
Output File
The name of the CSV file to put the BDS file's converted data into, this file will be created if necessary. Click the "Browse" button in order to choose a location, to choose a name select an existing file to replace or type the name in the "File name:" edit box.

Convert
This button will start the conversion process.

Cancel
This button will exit the application.



This dialog displays the amount of data that has been decoded. Once the blue bar reaches the end of the box the input file will be decoded and the output file will be created. The cancel button will cancel the conversion process and exit the application.



Convert More
This button returns the user to the main dialog in order to select another BDS file to convert into a CSV file.

Exit
This button will exit the application.

Understanding How the File Converter Operates

The file converter goes through a number of steps in order to convert BDS files into CSV files. These processes will be explained in the following portion of this document.

Determining the File Position and the Stored Number of Channels:

In order to determine how much of the file has been processed and how many channels of data are in the file the application gets the file size of your BDS file with the GetFileSize() command. Each byte must be assembled individually in order to handle both LSB[3] and MSB[4] configurations; this means that the amount of data decoded can easily be calculated. In order to calculate the amount of data that has been decoded the application takes the current byte number into the file and sets it as your current position and then uses the total number of bytes as the maximum position. In order to calculate the number of channels that the data contains the application checks the file size against the size of the file should there be for *numChan* number of channels and *numRdgs* number of readings. In order to do this the application must "guess" at the file size for *numChan* channels, this isn't truly guessing since only certain file sizes can be created. The check looks somewhat like this:

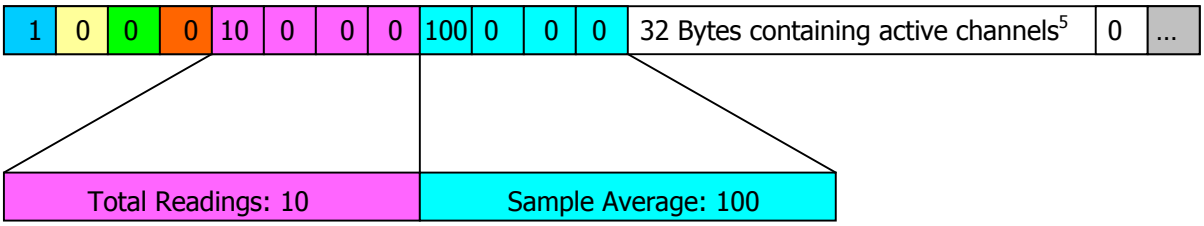
```
…
numChan = 1;
for(;;numChan++) {
  if(((dwSize - HeaderSize) / (8 + numChan * 4)) == (unsigned) numRdgs) {
    printf("Total of %d channels.\n", numChan);
    break;
  }
}
…
```

[3] LSB: Least Significant Byte. A method of ordering data in a Binary Data Set, data is order in the opposite fashion of the MSB configuration. BDS file configuration is automatically determined without user input.
[4] MSB: Most Significant Byte. A method of ordering data in a Binary Data Set, data is order in the opposite fashion of the LSB configuration. BDS file configuration is automatically determined without user input.

Once the application finds a total number of channels that fits the number of readings (obtained prior to this check) then the application knows exactly how many channels of data to expect.

The Header Information

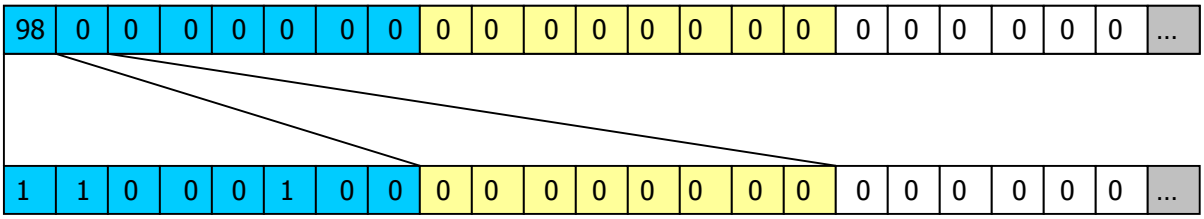
The header of the BDS file contains 44 bytes of important information for the application to perform its job. Here is a quick representation of the bytes, and their function.



Light Blue: BDS file version number (currently 2).
Light Yellow: Endian type, 0 is LSB and 1 is MSB.
Green: Power unit (0 = dBm, 1 = W, 2 = mW, 3 = uW, 4 = nW, 5 = pW).
Red: Wave/Time unit (0 = nm, 1 = THz, 2 = Secs, 3 = Min).
Purple: 4 bytes containing the number of readings.
Blue/Green: 4 bytes containing the number of samples that were averaged.

Determining Which Channels Are Active

The identification for which channels are active is stored in a 32-byte long section of the BDS file. Each character represents 8 channels for a chassis, and the four sets of 8 characters represent the four different chassis (each chassis has a possible total of 8 bytes and 8 channels per byte, or a maximum of 64 channels). The data set looks somewhat like this:



Part 1: The first 8 bytes represent chassis 0, the second set of bytes represent chassis 1, and so on. Byte 1 is then converted into binary, which contains the information for the first 8 channels (part 2).

Part 2: The first 8 bits (1 byte) represent the byte retrieved from chassis 0 (channels 1-8) the second set of bits represents the second byte retrieved from chassis 0 (channels 9-16). This data is interpreted as all of the channels de-activated except channels 1, 2, and 6.

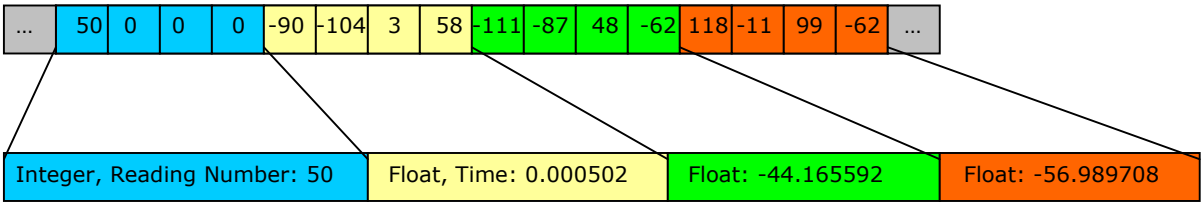
[5] See "Determining which channels are active" on the next page.

Channel Data

The channel data is composed of an 8-byte header and a data section composed of 4(*numChan*) bytes.

Total length (each reading): 8 + 4 * *numChan*.
Total length (all readings): *numRdgs* *(8 + 4 * *numChan*).
Total length (including header[6]):44 + *numRdgs* *(8 + 4 * *numChan*).

Example (2 channels):



Green: CSA Channel 1 Reading
Red: CSA Channel 2 Reading

dBm Optics
300 South Public Rd
Lafayette, CO 80026
303-464-1919

[6] Main file header, specific reading header included in all numbers.