June 14, 2012

**Disclaimer**

The test data are provided “as is,” as a courtesy to users. In no event will the CCSDS, its member Agencies, or the provider of the data be liable for any consequential, incidental, or indirect damages arising out of the use of or inability to use the data.

### General

The data is intended to test implementations of the CCSDS 121.0-B-2 *Lossless Data Compression* recommended standard (Blue Book Issue 2, May 2012) under different configurations. The set of test data does not exhaustively cover all possible parameter combinations.

The file extensions “.dat” and “.rz” indicate source data and compressed data respectively. Thus, each source file to be compressed has a name of the format

 *source\_filename*.dat

and associated with each source file is one or more compressed files with name(s) of the format

 *compressed\_filename*.rz

In cases where a single source data file has been compressed more than once (using different sets of options), the compressed filename indicates which set of options was used.

Each source data file contains a sequence of data sample values without any header. All source data samples are unsigned integers. When the input dynamic range *n* is 8 bits or less, each sample is stored using a single byte. When 8<*n*≤16 and 16<*n*, samples are stored as 2-byte and 4-byte integers, respectively, in little-endian format. (No endian issue arises for encoded data.)

Note that the *Basic* and *Restricted* coding options defined in the standard yield identical compressed output when *n*>4.

### Test Data

The test data are partitioned into three sub-folders: *AllOptions, LowEntropyOptions* and *ExtendedParameters*, described in the following subsections.

#### AllOptions Folder

The *AllOptions* folder contains 32 source input files corresponding to input dynamic range *n* from 1 to 32. For each value of *n*, the test set exercises all valid options at least once. Each source file name is of the form

 test\_p*LLL*n*XX*.dat

where *LLL* indicates the number of samples in the file (either 256 or 512) and *XX* indicates the sample bit depth *n*. The number of samples is 512 for *n*>16, and 256 otherwise.

Each source file is compressed using block length *J*=16, the unit-delay predictor, and bit depth corresponding to the given value of *n*. For *n*>4, the corresponding compressed output file name is of the form

 test\_p*LLL*n*XX*.rz

and for *n*≤4, there are two compressed files:

 test\_p*LLL*n*XX*-basic.rz test\_p*LLL*n*XX*-restricted.rz

corresponding to compression using Basic and Restricted coding options respectively.

#### LowEntropyOptions Folder

The *LowEntropyOptions* folder includes three test files to test the second-extension and zero-block coding options designed for low entropy input data. In these test files, each data sample has value 0 or 1. Each source file name is of the form

 Lowset*I*\_8bit.dat

where *I* is 1, 2, or 3. The source file includes 432 data samples when *I*=1 and 1024 samples otherwise.

Each source file is compressed using block length *J*=16 and the unit-delay predictor. For each *I* and each *n*>4, there is a corresponding compressed output file with name of the form

 Lowset*I*\_8bit.n*XX*.dat

where *XX* indicates the sample bit depth *n*. For each *I* and each *n*≤4, there are two compressed files:

 Lowset*I*\_8bit.n*XX*-basic.dat Lowset*I*\_8bit.n*XX*-restricted.dat

corresponding to compression using Basic and Restricted coding options respectively.

#### ExtendedParameters Folder

The *ExtendedParameters* folder includes the source file

 sar32bit.dat

which contains 512×512 samples, each a 32-bit unsigned integer. This file is used to test the maximum allowed values of block length *J* and reference interval *r*.

There are two compressed files:

 sar32bit.j16.r256.rz sar32bit.j64.r4096.rz

The first file corresponds to compression using *J*=16, *r*=256, and the second uses *J*=64, *r*=4096. Note that at the end of coding these *r* blocks (each block with *J* samples), the coded bit stream is filled with zero bits as needed to reach the next byte-boundary.

### Bug Report

Users can report problems to the current Data Compression Working Group Chair via email.