tsbin1.0.pdf – Binary Frequency and Data
Updated 65/2607/10

BINARY FORMAT DESCRIPTION

Introduction

This section describes an optional binary format for the numerical portion under the [Network Data] and [Noise Data] keywords. A binary format is useful for large files as it can reduce memory storage requirements 20 to 33 percent of the original ASCII data file.

The binary format is designated by the [Binary] keyword and is supported in [Version] 2.1 and above Touchstone files. The rules and limitations are discussed under the keyword descriptions.

Conversions to and from the binary format shall preserve all existing ASCII content in the file except for the numerical portions under the [Network Data] and/or [Noise Data] keywords. The [Binary] section of a Touchstone 2.1 file shall contain binary data only. Any conversion utility shall not process (shall ignore) comment characters, and the text which follows, to the end of the commented line. Blank lines shall also be ignored by binary conversion utilities.

[Binary]

Rules for Version 1.0 Files:
The [Binary] keyword is not permitted in Version 1.0 files.

Rules for Version 2.0 and Greater Files:
The [Binary] keyword is not permitted in Version 2.0 files. The [Binary] keyword is optional for Version 2.1 and greater files.

The [Binary] keyword indicates that network data is presented in binary format, for purposes of file size compression and faster file parsing.

The [Binary] keyword ~~is the first~~shall ~~keyword~~ follow~~ing~~ immediately after ~~either~~ the [Network Data] and/or the [Noise Data] keywords, excepting blank lines and comments, whenever the numerical data indicated by those keywords is encoded in a binary format. The [Binary] keyword is the only keyword permitted within the hierarchical scopes of the [Network Data] and [Noise Data] keywords.

The [Binary] keyword may appear only once under the [Network Data] keyword and only once under the [Binary Data] keyword.

[Binary] shall be followed by three arguments separated from the keyword and each other by whitespace.

The first argument indicates the numerical precision of the frequency information.

The second argument indicates numerical precision of the data.

The third argument indicates the assumed significance ordering of the bits within each byte.

Only one of the two strings below is permitted for each of the first two (precision) arguments; these shall include the numerical values and '-' (dash) character as shown:

32-Bit: also known as single precision floating point
64-Bit: also know as double precision

Only one of the two strings below is permitted for the third (byte order) argument; this shall include the '-' (dash) character as shown:

Big-Endian: most significant byte first
Little-Endian: least significant byte first

Example #:

```
[Binary] 64-Bit 32-Bit Little-Endian
```

The example above indicates 64-bit precision frequency and 32-bit precision floating point data in little-endian order.

The [Binary] keyword arguments shall be followed by a line-termination sequence. Immediately following the line-termination sequence shall be a single byte with value 0 (e.g., binary 00000000) to indicate that the information that follows will be in binary format.

No other keywords or comments are permitted after line-termination sequence following the [Binary] keyword's arguments.

The file shall be terminated by the [End] keyword, regardless of the presence of the [Binary] keyword.

Example #:

```
[Version] 2.1
# MHZ S RI R 50
[Number of Ports] 4
[Number of Frequencies] 1
! FREQ S11 S12 S13 S14
! S21 S22 S23 S24
! S31 S32 S33 S34
! S41 S42 S43 S44
!
[Network Data]                      ! numerical data in (hex) binary format
[Binary] 64-Bit 32-Bit Little-Endian
00 00 00 00 00 00 f0 3f 48 0f a9 3c 98 a4 72 bc 52 3d 74 3f ff 28 45 be fc
2d 17 bb ec b5 f6 3b bf 41 b8 bb 72 1b a5 ba 68 3d 74 3f 0c 29 45 be 73 0f a9
3c ee a4 72 bc ca 3c b8 bb 64 22 a5 ba a1 39 17 bb f2 b7 f6 3b 01 2c 17 bb bf
b5 f6 3b 6c 40 b8 bb 28 1d a5 ba b9 0f a9 3c 83 a4 72 bc 54 3d 74 3f e4 28 45
be 47 40 b8 bb ff 1d a5 ba ed 3b 17 bb 4e b8 f6 3b 6a 3d 74 3f 06 29 45 be cd
11 a9 3c 7b a6 72 bc
```

Formatted: Font: 10 pt

~~00 00 00 80 3f 65 8f ed 08 e9 21 95 3f 1a a5 4b ff 92 54 8e bf 85 c3 1c 4e aa~~
~~87 ec 3f 2b 58 99 df 1f a5 c8 bf 1b 37 c5 7b bf e5 62 bf 5e 74 42 80 bd d6 7e~~
~~3f 78 25 7d e5 37 08 77 bf 27 ca 48 37 6e a3 54 bf 3c 67 0b 08 ad 87 ec 3f d7~~
~~81 18 8d 21 a5 c8 bf f1 47 51 67 ee 21 95 3f 32 16 13 bc 9d 54 8e bf b2 31 47~~
~~40 99 07 77 bf 9d d5 02 7b 4c a4 54 bf e5 71 67 12 34 e7 62 bf 90 4a a7 31 fe~~
~~d6 7e 3f 0c 4f f9 21 80 e5 62 bf 31 8c 26 dd b7 d6 7e 3f 58 c0 cf 7b 0d 08 77~~
~~bf 23 be 0e fa a4 a3 54 bf d5 53 b3 20 f7 21 95 3f d4 c8 19 50 90 54 8e bf bb~~
~~a8 cc 83 aa 87 ee 3f d4 04 9b 84 1c a5 c8 bf ae 96 94 eb 08 08 77 bf df 58 01~~
~~d2 bf a3 54 bf cc 6b 8a 9f 7d e7 62 bf 8a 4a 97 bc 09 d7 7e 3f 72 4c bb 3d ad~~
~~87 ec 3f 01 ed 58 b6 20 a5 c8 bf 9b 5f c5 90 39 22 95 3f 43 81 2d 65 cf 54 8e~~
~~bf~~

[End]

The example corresponds to the following ASCII text, with the addition of the [Binary] keyword, its arguments and the hex data shown:

```
[Version] 2.1
# MHZ S RI R 50
[Number of Ports] 4
[Number of Frequencies] 1
! FREQ S11 S12 S13 S14
! S21 S22 S23 S24
! S31 S32 S33 S34
! S41 S42 S43 S44
!
[Network Data]
1.000000e+001
2.063717e-002 -1.480975e-002 9.540607e-001 -1.925392e-001
-2.306818e-003 7.529011e-003 -5.623072e-003 -1.259668e-003
9.540620e-001 -1.925394e-001 2.063725e-002 -1.480983e-002
-5.622481e-003 -1.259875e-003 -2.307512e-003 7.529252e-003
-2.306700e-003 7.528990e-003 -5.622914e-003 -1.259719e-003
2.063738e-002 -1.480973e-002 9.540608e-001 -1.925388e-001
-5.622897e-003 -1.259744e-003 -2.307649e-003 7.529295e-003
9.540621e-001 -1.925393e-001 2.063837e-002 -1.481020e-002
[End]
```