SAMPLE FORMAT

Description

Any of these formats can be used for multiplexed signal files, in which samples from two or more signals are stored alternately.

Format 8

Each sample is represented as an 8-bit first difference; i.e., to get the value of sample *n*, sum the first *n* bytes of the sample data file together with the initial value from the header file. When format 8 files are created, first differences which cannot be represented in 8 bits are represented instead by the largest difference of the appropriate sign (-128 or +127), and subsequent differences are adjusted such that the correct amplitude is obtained as quickly as possible. Thus there may be loss of information if signals in another of the formats listed below are converted to format 8. Note that the first differences stored in multiplexed format 8 files are always determined by subtraction of successive samples from the same signal (otherwise signals with baselines which differ by 128 units or more could not be represented this way).

Format 16

Each sample is represented by a 16-bit two’s complement amplitude stored least significant byte first. Any unused high-order bits are sign-extended from the most significant bit. Historically, the format used for MIT-BIH and AHA database distribution 9-track tapes was format 16, with the addition of a logical EOF (octal 0100000) and null-padding after the logical EOF.

Format 24

Each sample is represented by a 24-bit two’s complement amplitude stored least significant byte first.

Format 32

Each sample is represented by a 32-bit two’s complement amplitude stored least significant byte first.

Format 61

Each sample is represented by a 16-bit two’s complement amplitude stored most significant byte first.

Format 80

Each sample is represented by an 8-bit amplitude in offset binary form (i.e., 128 must be subtracted from each unsigned byte to obtain a signed 8-bit amplitude).

Format 160

Each sample is represented by a 16-bit amplitude in offset binary form (i.e., 32,768 must be subtracted from each unsigned byte pair to obtain a signed 16-bit amplitude). As for format 16, the least significant byte of each pair is first.

Format 212

Each sample is represented by a 12-bit two’s complement amplitude. The first sample is obtained from the 12 least significant bits of the first byte pair (stored least significant byte first). The second sample is formed from the 4 remaining bits of the first byte pair (which are the 4 high bits of the 12-bit sample) and the next byte (which contains the remaining 8 bits of the second sample). The process is repeated for each successive pair of samples. Most of the signal files in PhysioBank are written in format 212.

Format 310

Each sample is represented by a 10-bit two’s-complement amplitude. The first sample is obtained from the 11 least significant bits of the first byte pair (stored least significant byte first), with the low bit discarded. The second sample comes from the 11 least significant bits of the second byte pair, in the same way as the first. The third sample is formed from the 5 most significant bits of each of the first two byte pairs (those from the first byte pair are the least significant bits of the third sample). Note that the unused bit in each byte pair is set to zero when using the WFDB library to write a format 310 signal file. The entire process is then repeated for each successive set of three samples.

Format 311

Each sample is represented by a 10-bit two’s-complement amplitude. Three samples are bit-packed into a 32-bit integer as for format 310, but the layout is different. Each set of four bytes is stored in little-endian order (least significant byte first, most significant byte last). The first sample is obtained from the 10 least significant bits of the 32-bit integer, the second is obtained from the next 10 bits, the third from the next 10 bits, and the two most significant bits are unused. This process is repeated for each successive set of three samples.

[Source: http://www.physionet.org/physiotools/wag/signal-5.htm]