

Some useful function generators

GEN 10

This subroutine generates composite waveforms made up of weighted sums of simple sinusoids.

```
f # time size 10 str1 str2 str3 str4 . . . .
```

ARGUMENTS

#, time, size - the usual GEN parameters: number, time of creation and number of points

str1, str2, str3, etc. - relative strengths of the fixed harmonic partial numbers 1,2,3, etc., beginning in p5. Partial not required should be given a strength of zero.

GEN 5, GEN 7

These subroutines are used to construct functions from segments of exponential curves (GEN05) or straight lines (GEN07). You can do the same with GEN25, GEN27, but using breakpoint style arguments.

```
f # time size 5 a n1 b n2 c . . .
f # time size 7 a n1 b n2 c . . .
```

ARGUMENTS

#, time, size - the usual GEN parameters.

a, b, c, etc. - ordinate values, in odd-numbered pfields p5, p7, p9, . . . For GEN05 these must be nonzero and must be alike in sign. No such restrictions exist for GEN07.

n1, n2, etc. - length of segment (no. of storage locations, or points), in even-numbered pfields. Cannot be negative, but a zero is meaningful for specifying discontinuous waveforms (e.g. in the example below). The sum $n1 + n2 + \dots$ will normally equal size for fully specified functions. If the sum is smaller, the function locations not included will be set to zero; if the sum is greater, only the first size locations will be stored.

Discrete-point linear interpolation implies an increase or decrease along a segment by equal differences between adjacent locations; exponential interpolation implies that the progression is by equal ratio. In both forms the interpolation from a to b is such as to assume that the value b will be attained in the $(n + 1)^{\text{th}}$ location. For discontinuous functions, and for the segment encompassing the end location, this value will not actually be reached, although it may eventually appear as a result of final scaling.

EXAMPLE

f 1 0 256 7 0 128 1 128 0

This describes a triangular-shaped envelope.

GEN 25, GEN 27

These subroutines are used to construct functions from segments of exponential curves (GEN25) or straight lines (GEN27) in breakpoint fashion. See also GEN05 and GEN07.

f# time size 25 x1 y1 x2 y2 x3 y3 . . .

f# time size 27 x1 y1 x2 y2 x3 y3 . . .

ARGUMENTS

#, time, size - the usual GEN parameters.

x1, x2, x3, etc. - locations in table at which to attain the following y value. Must be in increasing order. If the last value is less than size, then the rest will be set to zero. Should not be negative but can be zero.

y1, y2, y3, etc. - Breakpoint values attained at the location specified by the preceding x value. For GEN25 these must be non-zero and must be alike in sign. No such restrictions exist for GEN27.

NOTE

If p4 is positive, functions are post-normalized (rescaled to a maximum absolute value of 1 after generation). A negative p4 will cause rescaling to be skipped.

EXAMPLES

f 1 0 257 27 0 0 100 1 200 -1 256 0

Describes a function which begins at 0, rises to 1 at the 100th table location, falls to -1, by the 200th location, and returns to 0 by the end of the table. The interpolation is linear.

f 1 0 257 25 0 0.001 100 1 200 .001 256 0.001

Similar to above, but creates exponential curve. No values ≤ 0 are allowed in GEN25

NOTE for all GEN functions above

If p4 (the GEN number) is positive, functions are post-normalized (rescaled to a maximum absolute value of 1 after generation). A negative p4 will cause rescaling to be skipped.