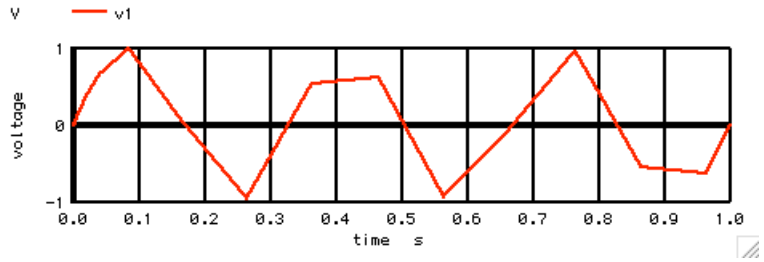
*==========Transient_Timing_SineWave_100msec==========

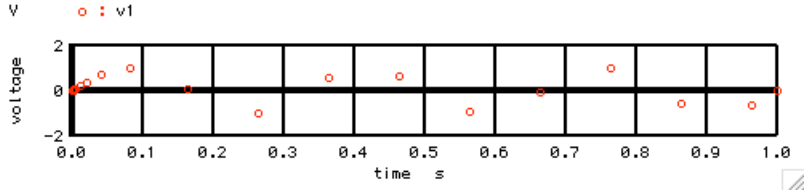
For Sine-waves, it looks like another problem arises if the tmax is set too low.

```
*V_SIN#    NODE_P NODE_N DC      VALUE   SIN(    V_DC    AC_MAG FREQ    DELAY   FDamp)
V_SIN      V1     0      DC      0       SIN(    0       1      3   )
*TRAN      TSTEP  TSTOP  TSTART  TMAX    ?UIC?
.tran      100m   1      0       100m
```

It appears that getting close to nyquist attenuates the sampled signal.

```
plot    v1
plot    v1    pointplot
```



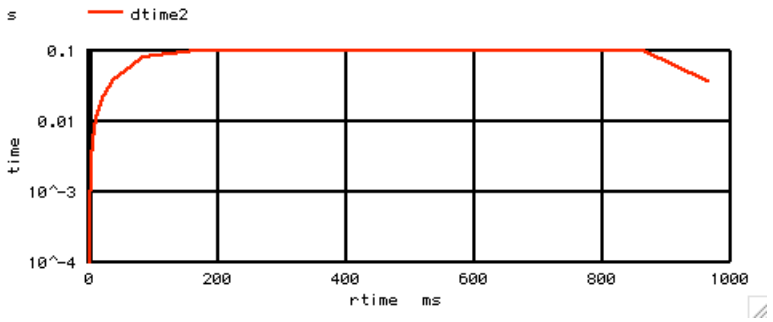


The following shows some critical details in red which are required to do some math processing of the waveform vectors.

The intention is the show the order of magnitudes for the timing.

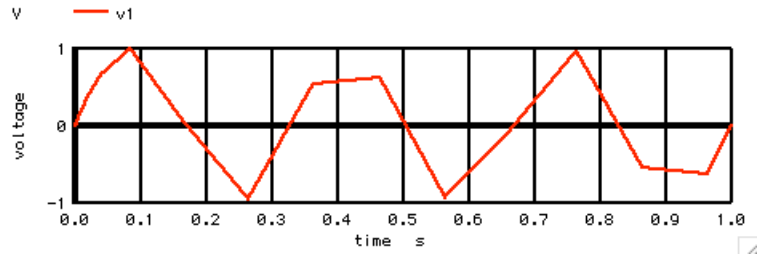
```
let     num = length(time)-2
compose dtime start = 0 stop = $&num step =1
compose rtime start = 0 stop = $&num step =1
let     i = 0
repeat  $&num
let     i = i +1
let     dtime[i] = time[i +1] -time[i]
let     rtime[i] = time[i]
end
let     dtime2 = abs(dtime)+100u
plot    dtime2 vs rtime ylog
```

Except at the beginning, the sine wave appears to want to have a consistent timing.



==========

The RMS value will be wrong due to other reasons than an inconsistent timing.




```
let     vrms1_cdhw = sqrt(mean(v1*v1))
echo   "INPUT RMS SINE prelinear  =  $&vrms1_cdhw"
```

==========

The output will be as follows.

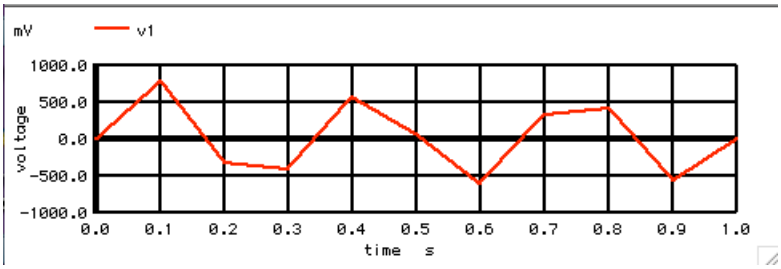
```
INPUT RMS SINE prelinear  =  0.491849
```

==========

A linearize statement will not improve the RMS value.

==========

```
linearize
plot    v1    pointplot
let     vrms1_cdhw = sqrt(mean(v1*v1))
echo   "INPUT RMS SINE postlinear  =  $&vrms1_cdhw"
```




```
INPUT RMS SINE postlinear  =  0.449348
```

==========Full_Netlist_For_Copy_Paste==========

```
RMS_SINE_100msec
.Option srcsteps = 1 set Gmin = 1.0000E-02
*=======Circuit_Netlist=========================

V_SIN   V1   0 DC  0  SIN(   0   1 3   )

*TRAN   TSTEP TSTOP TSTART TMAX ?UIC?
.tran   100m 1 0 100m
.control
run
set     pensize = 2

plot    v1        pointplot
plot    v1

let     vrms1_cdhw = sqrt(mean(v1*v1))
echo   "INPUT RMS SINE prelinear  =  $&vrms1_cdhw"

let     num = length(time)-2
compose dtime start = 0 stop = $&num step =1
compose rtime start = 0 stop = $&num step =1
let     i = 0
repeat  $&num
let     i = i +1
let     dtime[i] = time[i +1] -time[i]
let     rtime[i] = time[i]
end
let     dtime2 = abs(dtime)+100u
plot    dtime vs rtime
```

```
plot    dtime2 vs rtime ylog


linearize
plot    v1 pointplot
plot    v1
let     vrms1_cdhw = sqrt(mean(v1*v1))
echo   "INPUT RMS SINE postlinear  =  $&vrms1_cdhw"

let     num = length(time)-2
compose dtime start = 0 stop = $&num step =1
compose rtime start = 0 stop = $&num step =1
let     i = 0
repeat  $&num
let     i = i +1
let     dtime[i] = time[i +1] -time[i]
let     rtime[i] = time[i]
end
plot    dtime vs rtime

.endc
.end

7.12.10_10.31AM
dsauersanjose@aol.com
Don Sauer
http://www.idea2ic.com/
```