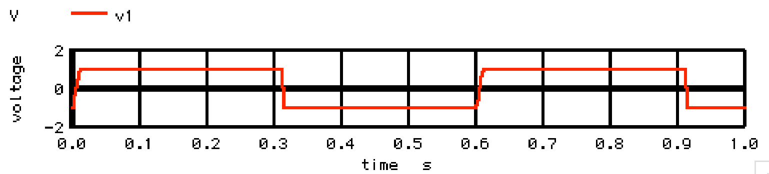
*=============Transient_Timing_SquareWave_100msec============

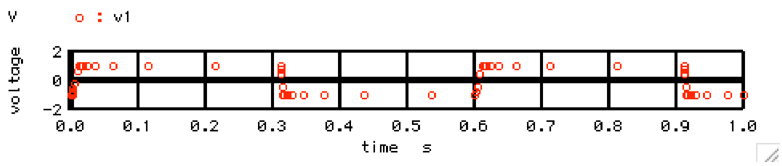
The timing for transient analysis is not consistent. More time points are taken when something is moving. A square wave which has its tmax set low will demonstrate this fact.

```
=================================================================
*V_PULSE NODE_P NODE_N DC      VALUE  PULSE( VINIT  VPULSE TDELAY TRISE  TFALL  PWIDTH PERIOD )
V_PUL    V1     0      DC      0      PULSE( -1     1      1m     10m    3m     300m   600m )
*TRAN    TSTEP  TSTOP  TSTART  TMAX   ?UIC?
.tran    100m   1      0       100m
=================================================================
```

The pointplot statement can show the actual time points.

```
=================================================================
plot    v1
plot    v1    pointplot
```



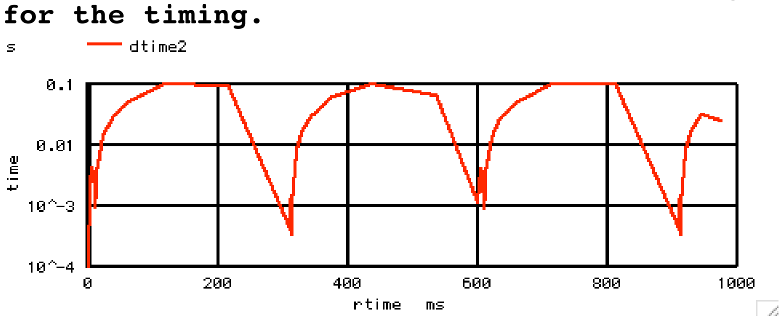



```
=================================================================
```

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

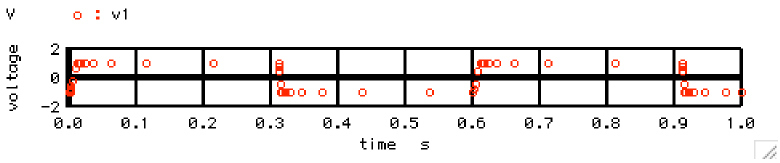
```
=================================================================
let     num = length(time)-2
compose dtime start = 0 stop = $&num step =1
compose rtime start = 0 stop = $&num step =1
let     i = 0
repeat  $&num
let     i = i +1
let     dtime[i] = time[i +1] -time[i]
let     rtime[i] = time[i]
end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

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




```
=================================================================
```

This timing profile can effect things like doing a RMS of a waveform.




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

==

The output will be as follows.

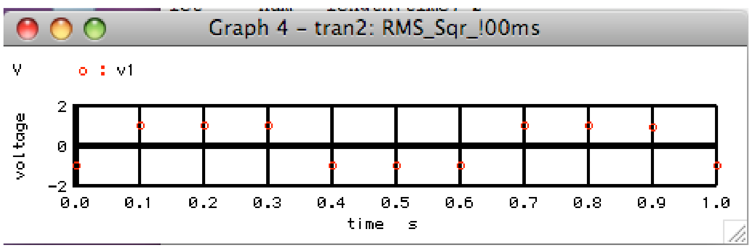
```
INPUT RMS Square prelinear  =  0.930845
```

==

A linearize statement will make the timing and RMS value more consistent.

==

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




```
INPUT RMS Square postlinear  =  1
```

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

```
RMS_Sqr_!00ms
.Option srcsteps = 1 set Gmin = 1.0000E-02
*=========Circuit_Netlist=========================
V_PUL   V1    0     DC     0    PULSE( -1  1 1m 10m  3m  300m 600m )

*TRAN   TSTEP TSTOP TSTART TMAX ?UIC?
.tran   100m  1     0      100m
.control
run
set      pensize = 2
plot     v1
plot     v1    pointplot

let      vrms1_cdhw = sqrt(mean(v1*v1))
echo    "INPUT RMS Square 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     dtime2 vs rtime ylog

linearize
plot     v1    pointplot
let      vrms1_cdhw = sqrt(mean(v1*v1))
echo    "INPUT RMS Square 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/