*=============Transient_Timing_PWL_100msec============

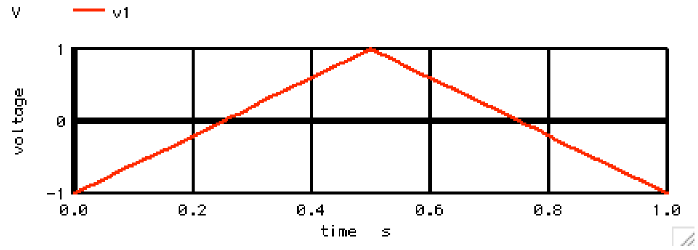
A PieceWise Linear waveform appears to add extra timing points after every time point.

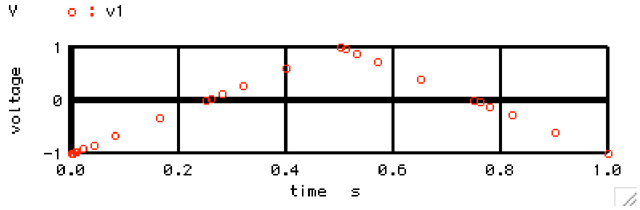
```
========================================================================
*V_PWL#  NODE_P NODE_N DC      VALUE  PWL(   T1    V1    T2     V2    T3     V3      ...>)
V_PUL    V1     0      DC      0      PWL(   0     -1    .25    0     .5     1      .75  0 1 -1 )
*TRAN    TSTEP  TSTOP  TSTART  TMAX   ?UIC?
.tran    100m   1      0       100m
```

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

The change in timing is not as much as for a square-wave.

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





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

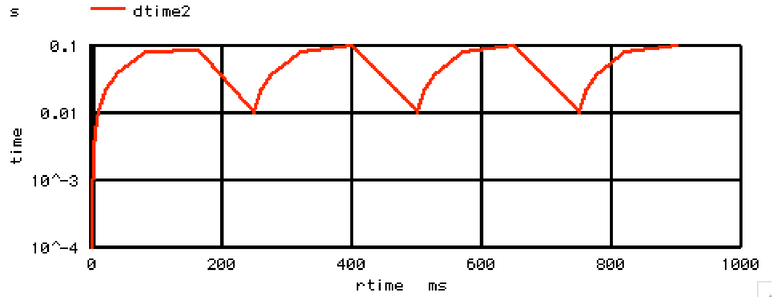
This is where being able to view timing in orders of magnitude comes in handy.

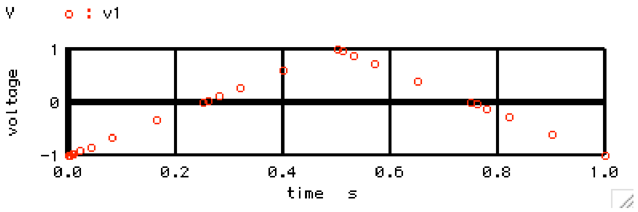
```
========================================================================
let      num = length(time)-2
compose dtime start = 0 stop = $&num step =1
compose rtime start = 0 stop = $&num step =1
let      i = 0
repeat   $&num
let      i = i +1
let      dtime[i] = time[i +1] -time[i]
let      rtime[i]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

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

The timing is only changing an order of magnitude. But it is enough to effect the RMS value.

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





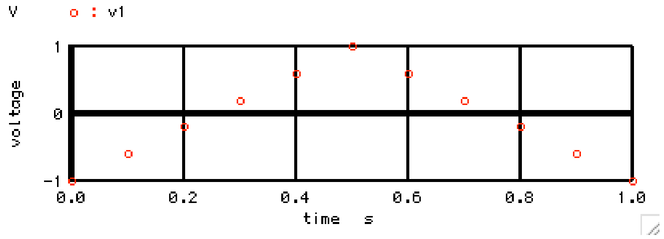
INPUT RMS PWL prelinear = 0.76695

==

The Linearize statement will make the waveform to be near perfect. But there is still the same increase in points at the beginning of the waveform. This will affect the RMS value which should be 0.5773502691896258

==

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



INPUT RMS PWL postlinear = 0.64667

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

```
RMS_PWL_!00ms
.Option srcsteps = 1 set Gmin = 1.0000E-02
*=======Circuit_Netlist========================
V_PUL   V1    0     DC     0    PWL( 0  -1 .25 0 .5 1 .75 0 1 -1 )

*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 PWL 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 PWL 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/