

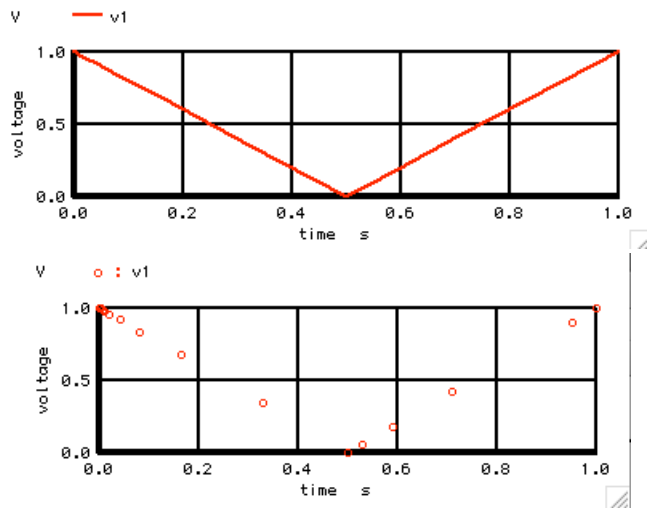
# \*=====Vector\_Processing\_Sort=====

There are some waveform processing functions that can save a lot of time and trouble. Most of them are intuitive. The `sort` function is powerful, but a little harder to visualize how it works.

```
*V_PWL#  NODE_P  NODE_N  DC    VALUE  PWL(  T1  V1  T2  V2  T3  V3  ...>)
V_PUL    V1      0      DC    0      PWL(  0  1  .5  0  1  1  )
*TRAN    TSTEP  TSTOP  TSTART TMAX  ?UIC?
.tran    300m  1      0      300m
```

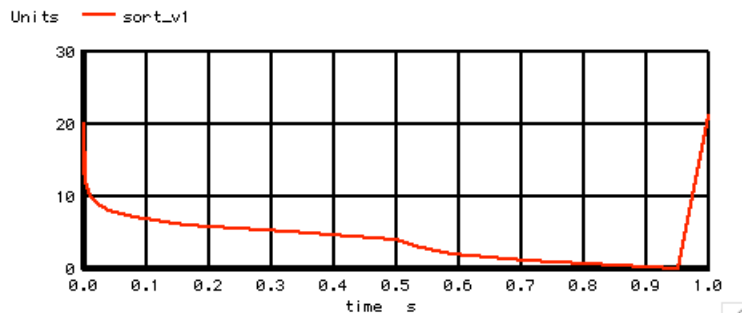
A triangle wave can show where the confusion comes from.

```
plot v1
plot v1 pointplot
```



When one `sorts` a waveform and plots the graph, what does the `sort` waveform mean?

```
let sort_v1= sortorder(v1)
plot sort_v1
```



One can make more sense by printing out the data.

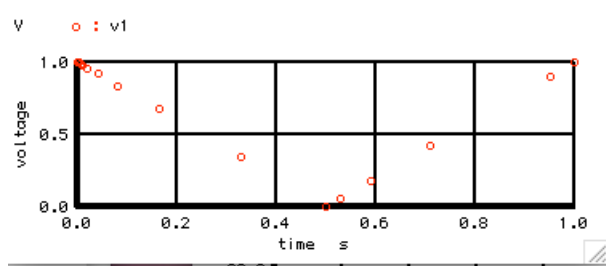
```
print v1 sort_v1
```

Circuit: Sort\_Order

Index	time	v1	sort_v1
0	0.00000e+00	1.00000e+00	1.60000e+01
1	2.00000e-05	9.99960e-01	1.70000e+01
2	4.00000e-05	9.99920e-01	1.80000e+01
3	8.00000e-05	9.99840e-01	1.50000e+01
4	1.60000e-04	9.99680e-01	1.90000e+01
5	3.20000e-04	9.99360e-01	1.40000e+01
6	6.40000e-04	9.98720e-01	1.30000e+01
7	1.28000e-03	9.97440e-01	2.00000e+01
8	2.56000e-03	9.94880e-01	1.20000e+01
9	5.12000e-03	9.89760e-01	1.10000e+01
10	1.02400e-02	9.79520e-01	1.00000e+01
11	2.04800e-02	9.59040e-01	9.00000e+00
12	4.09600e-02	9.18080e-01	8.00000e+00
13	8.19200e-02	8.36160e-01	7.00000e+00
14	1.63840e-01	6.72320e-01	6.00000e+00
15	3.27680e-01	3.44640e-01	5.00000e+00
16	5.00000e-01	0.00000e+00	4.00000e+00
17	5.30000e-01	6.00000e-02	3.00000e+00
18	5.90000e-01	1.80000e-01	2.00000e+00
19	7.10000e-01	4.20000e-01	1.00000e+00
20	9.50000e-01	9.00000e-01	0.00000e+00
21	1.00000e+00	1.00000e+00	2.10000e+01

MacSpice 7 ->

Since timing is inconsistent, the lowest value for v1 happens at time period 0.5, but at the index value of 16. The first value of the sorted data will be 16. The highest values of v1 are at an index of 0 and 21. So the last values of the sorted data are those two numbers.



So the `sort` function give index values which can be used to find voltage values or points in time where a waveform is minimum or maximum. Using it can be much faster than looping arrays.

One can also find the 6th value with the following code.

```
let nthvalue = sort_v1[5]
print v1[nthvalue]
```

Sort\_Order  
Transient Analysis Mon Jul 19 10:28:18 2010

Index	time	v1	sort_v1
0	0.00000e+00	1.00000e+00	1.60000e+01
1	2.00000e-05	9.99960e-01	1.70000e+01
2	4.00000e-05	9.99920e-01	1.80000e+01
3	8.00000e-05	9.99840e-01	1.50000e+01
4	1.60000e-04	9.99680e-01	1.90000e+01
5	3.20000e-04	9.99360e-01	1.40000e+01
6	6.40000e-04	9.98720e-01	1.30000e+01
7	1.28000e-03	9.97440e-01	2.00000e+01
8	2.56000e-03	9.94880e-01	1.20000e+01
9	5.12000e-03	9.89760e-01	1.10000e+01
10	1.02400e-02	9.79520e-01	1.00000e+01
11	2.04800e-02	9.59040e-01	9.00000e+00
12	4.09600e-02	9.18080e-01	8.00000e+00

13	8.19200e-02	8.36160e-01	7.00000e+00
14	1.63840e-01	6.72320e-01	6.00000e+00
15	3.27680e-01	3.44640e-01	5.00000e+00
16	5.00000e-01	0.00000e+00	4.00000e+00
17	5.30000e-01	6.00000e-02	3.00000e+00
18	5.90000e-01	1.80000e-01	2.00000e+00
19	7.10000e-01	4.20000e-01	1.00000e+00
20	9.50000e-01	9.00000e-01	0.00000e+00
21	1.00000e+00	1.00000e+00	2.10000e+01

v1[nthvalue] = 6.72320e-01

=====**Full\_Netlist\_For\_Copy\_Paste**=====

```
Sort_Order
.Option srcsteps = 1 set Gmin = 1.0000E-02
*=====Circuit_Netlist=====
V_PUL V1 0 DC 0 PWL( 0 1 .5 0 1 1 )

*TRAN TSTEP TSTOP TSTART TMAX ?UIC?
.tran 300m 1 0 300m
.control
run
set pensize = 2

plot v1
plot v1 pointplot
let sort_v1= sortorder(v1)
plot sort_v1
print v1 sort_v1

*=====find_6th_largest_Value=====
let nthvalue = sort_v1[5]
print v1[nthvalue]

.endc
.end
```

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