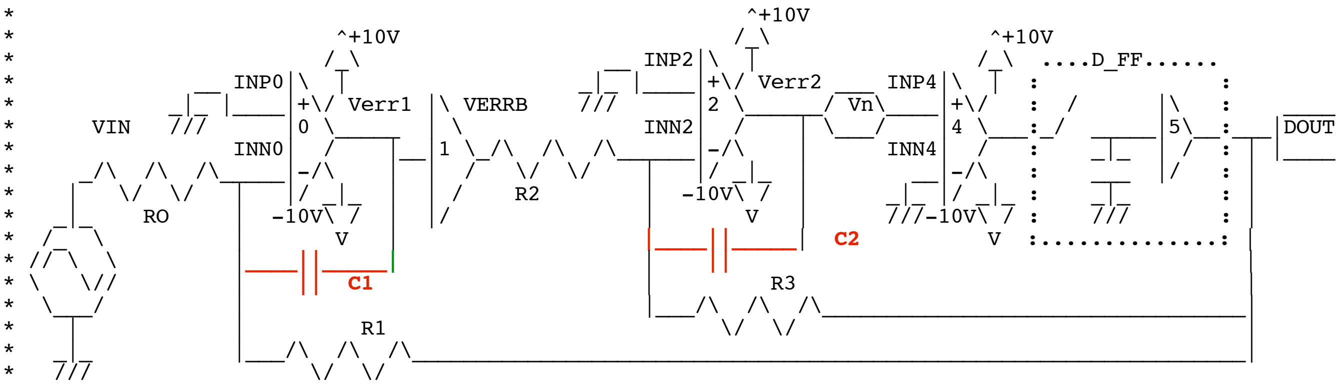
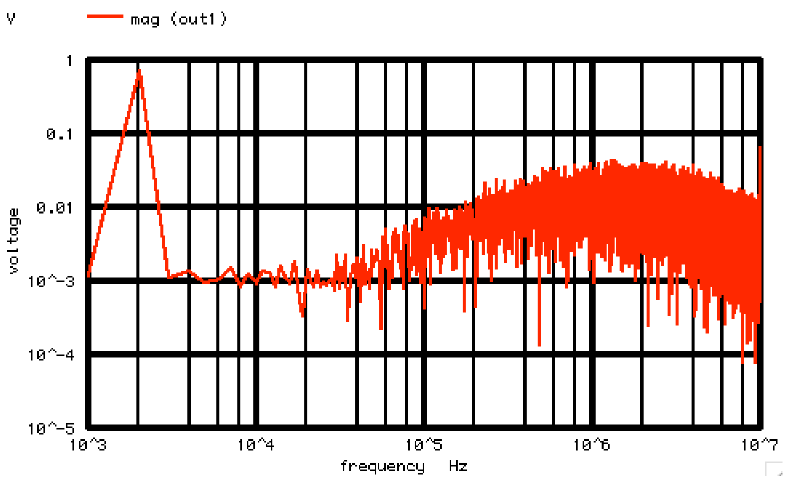
Sigma Delta Second Order Noise Shaping



Well, if one integrator can shape the output noise in the right way,
why not used two?



In this case, the noise voltage is dropping a decade in level
for every decade decrease in input frequency.

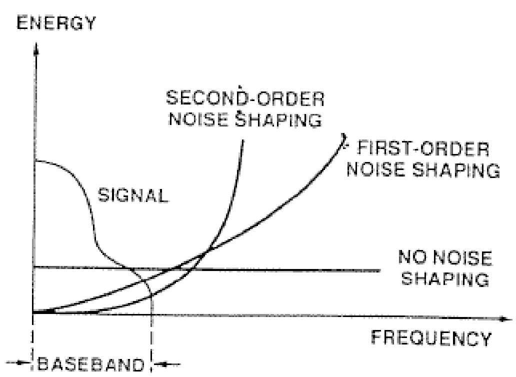


Fig. 9. Spectra of signal and quantization-error components in oversampled quantized signal $y(n)$ for frequencies in vicinity of signal baseband.

This fact encourages oversampling to make it easier
for a low pass digital filter to remove much of the quantization noise.

```
===================MacSpiceCode============================================

Sigma_Delta2
*                                                         ^+10V
*                          ^+10V                         /_\                  ^+10V
*                         /_\                    __ INP2|\  T                /_\  ....D_FF......
*                  __ INP0|\  T                 _|_ |____|+\/ Verr2  ___  INP4|\  T  :              :
*               _|_ |____|+\/ Verr1  |\ VERRB    ///     |2 \______/ Vn\____|+\/   :  /      |\    :  ____
*      VIN    ///        |0 \______   \              INN2|   /      |\___/    |4 \___:_/  _____|5\__:___|DOUT|
*                  INNO  |   /     |__|1 \_/\  /\  /\______|-/\      |     INN4|   /   :   _|_   |  /  : |  |____|
*     _/\  /\  /\_______|-/\       |  |   /   \/  \/    |  |/ _|_    |      __|-/\    :   ___   |/   : |
*     |   \/  \/    |    |/ _|_    |  |  /     R2       |  -10V\ /   |     _|_ |/ _|_ :   _|_        : |
*    _|_     RO     |   -10V\ /    |  | /               |       V    |     ///-10V\ / :   ///        : |
*   /   \           |        V     |                    |_____||_____|  C2         V  :..............: |
*  //\  \           |_____||_______|                    |     ||                                     |
*  \    \//         |     ||  C1                        |          R3                                |
*   \___/           |                                   |___/\  /\  /\_______________________________|
*     |             |          R1                            \/  \/                                  |
*    _|_            |___/\  /\  /\_______________________________________________________________|
*    ///                 \/  \/

*================Need_A_voltage_Source_to_alter======================"
VN1                VERR2  INP4   DC     0
*V_PULSE#          NODE_P NODE_N DC     VALUE  PULSE( VINIT  VPULSE TDELAY TRISE  TFALL  PWIDTH PERIOD )
VCNTL              CNTL   0      DC     0      PULSE( 0      1      .5p     1n    1n     5n     50n      )
*V_SIN#            NODE_P NODE_N DC     VALUE  SIN(   V_DC   AC_MAG FREQ   DELAY  FDamp)
VIN                VIN    0      DC     0      SIN(   0      .7      2k     )

RO                 VIN    INNO   300k
BO                 VERR1  0      V =    1*tanh(100*tanh(100*tanh(100*tanh(100*tanh( -V(INNO))))))
C1                 INNO   VERR1   .03n
R1                 INNO   OUT1   300k
B1                 VERRB  0      V =    -V(VERR1)
R2                 VERRB  INN2   300k
B2                 VERR2  0      V =    1*tanh(100*tanh(100*tanh(100*tanh(100*tanh( -V(INN2))))))
C2                 INN2   VERR2  .03n
R3                 INN2   OUT1   300k
B4                 OUTO   0      V =    1*tanh(100*tanh(100*tanh(100*tanh(100*tanh( V(INP4)  )))))

*S_NUMB            NODE1  NODE2  CNTL_P CNTL_N MODEL  ON/OFF
BCNTL              NCNTL  0      V  =   1-V(CNTL)
SA                 OUTO   OUTA   CNTL   0      SW
CA                 OUTA   0      100p
BA                 OUTB   0      V  =   V(OUTA)
SB                 OUTB   OUTC   NCNTL  0      SW
CC                 OUTC   0      100p
BC                 OUT1   0      V  =   V(OUTC)

.MODEL             SW     SW(    VT=.2 VH=.1m  RON=10m  ROFF=1MEG)

.control
set                pensize = 2
echo               "=======================Want_10000_.1us_steps==============================="
let n =            10000
let tstep =        .1us
let period_t =     n*tstep
let Bin_Hz =       1/period_t
let nyquist =      .5/tstep
echo               "Total_Period_s =       $&period_t"
echo               "Bin_Resolutio_Hz =     $&Bin_Hz"
echo               "Sample_Period_s =      $&tstep"
echo               "Nyquist_Hz =           $&nyquist"
echo               "=======================Create_PWL_array_and_Index_and_Plot=============="
let pwl_1 =        vector(2*n)*tstep*0.5
let ii =           vector(2*$&n)
*plot              pwl_1  vs ii
echo               "=======================Add_1Vrms_Noise_to_PWL_array====================="
let index =        0
repeat             $&n
let                pwl_1[1+2*index] = 30m*(rnd(127)+rnd(127)+rnd(127)+rnd(127)+rnd(127)+rnd(127)+rnd(127)+
rnd(127)-507.5)/102.879
let index =        index + 1
end
echo               "=======================Install_the_PWL_array============================"
alter              @vn1[pwl] = pwl_1
echo               "=======================Run_and_Plot====================================="
let period_s =     tstep/2
let trans_per =    tstep/20
tran               $&trans_per $&period_t   0  $&trans_per
```

```
plot             out1 vin
plot             out0   out1/1.1  cntl vin xlimit .495m .505m
plot             out0   out1/1.1  vin xlimit .49m .51m

echo             "=======================FFT_and_Plot====================================="
linearize
let              FFT_BandWidth_Hz =      10Meg
let              FFT_resolution_Hz =     1k
echo             "FFT_BandWidth_Hz=      $&FFT_BandWidth_Hz"
echo             "FFT_resolution_Hz=     $&FFT_resolution_Hz"
set              specwindow=             "rectangular"
*set              specwindow=             "hamming"
spec             $&FFT_resolution_Hz     $&FFT_BandWidth_Hz   $&FFT_resolution_Hz     v(out1)
let expect_V =   (sqrt(2)/sqrt(500k/1k))/(1+(frequency/550k)*(frequency/500k)*(frequency/500k)*(frequency/
500k)*(frequency/500k))
plot             mag (out1) loglog
echo             "===================done============================================="

.endc
.end

echo             "=======================FFT_and_Plot====================================="
linearize
let              FFT_BandWidth_Hz =      10Meg
let              FFT_resolution_Hz =     1k
echo             "FFT_BandWidth_Hz=      $&FFT_BandWidth_Hz"
echo             "FFT_resolution_Hz=     $&FFT_resolution_Hz"
set              specwindow=             "rectangular"
spec             $&FFT_resolution_Hz     $&FFT_BandWidth_Hz   $&FFT_resolution_Hz     v(inn1)
let expect_V =   (sqrt(2)/sqrt(500k/1k))/(1+(frequency/550k)*(frequency/500k)*(frequency/500k)*(frequency/
500k)*(frequency/500k))
plot             mag (inn1) expect_V loglog
echo             "===================done============================================="
```

7.22.11_12.59PM

dsauersanjose@aol.com
Don Sauer