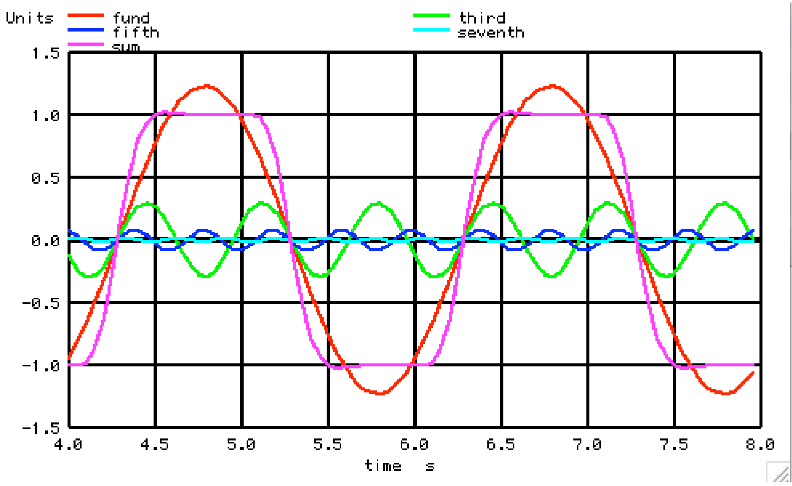
*=======Better_Bessel_GroupDelay============

RUN A SQUARE WAVE THROUGH A 6 POLE BESSEL.
EXTRACT ALL THE OUTPUT HARMONICS.
RECONSTRUCT THE OUTPUT WAVEFORM FROM THE HARMONICS.
VIEW THE DELAY OF ALL THE HARMONICS.
AND DO SO IN ONE SPICE FILE.



The dissection of the harmonics of a lowpass filter's output can show how phase delay effects group delay.

Below is a behavioral model for a 6 pole Bessel filter.

==

```
Better_Bessel_6P_State_Variable
*  dsauersanjose@aol.com
*  www.idea2ic.com
*  8.13.10_10.02AM
*
*    R0A= 1/A0A                   R0B= 1/A0B                 R0C= 1/A0C
*   _/\  /\_                  _/\  /\__                  _/\  /\_
*   |  \/   |    R3A= 1/A0A  |  \/    |       R3B= 1/A0B |   \/   |  R3C= 1/A0C
*   |     __|_/\  /\  /\_____|______  |_/\  /\  /\_______|________|_/\  /\  /\_____________
*   |     |      \/  \/          LPA|  |   \/  \/           LPB|  |  \/  \/              LPC|
*  _|_    |     R2A= 1/A1A          |  |   R2B= 1/A1B          |  |   R2C= 1/A1C          |
* /VIN\   |_/\  /\  /\______|       |  |_/\  /\  /\______|     |  |_/\  /\  /\______|     |
* \___/   |    \/  \/       |       |  |    \/  \/       |     |  |    \/  \/       |     |
*   |     |    R1A= 1       |       |  |   R1  = 1       |     |  |   R1C= 1        |     |
*  _|_    |_/\  /\_         |BPA    |  |_/\  /\_         |BPB  |  |_/\  /\_         |BPC  |
*  ///    |    \/  |        |       |  |    \/  |        |     |  |    \/  |        |     |
*         |  |\  HPA|XONES1A|XONES2A|  |  |\  HPB|XONES1B|XONES2B|  |  |\  HPC|XONES1C|XONES2C|
*         |_|-\     |  ____ |  ____ |  |_|-\    |  ____ |  ____ |  |_|-\    |  ____ |  ____ |
*     SUMA|   \____|_|1/s|_|_|1/s|_|   |   \____|_|1/s|_|_|1/s|_|  |   \____|_|1/s|_|_|1/s|_|
*         |   /       |____|  |____|  SUMB  /      |____|  |____| SUMC  /      |____|  |____|
*       __|+/ OPAA                      __|+/ OPAB                  __|+/ OPAC
*      |  |/                           |  |/                       |  |/
*      |                               |                           |
*     _|_                             _|_                         _|_
*     ///                             ///                         ///
*      Bessel terms
*      (s^2 +1.861s +3.63)*(s^2 +2.76s +2.85)*(s^2 +3.143s + 2.57)
.OPTIONS  GMIN=1e-18    METHOD=euler  srcsteps = 1  gminsteps = 1
```

*=========Using_Simple_Amplifiers===================

Pretty idealized models are being used for Op Amps and integrators.

==

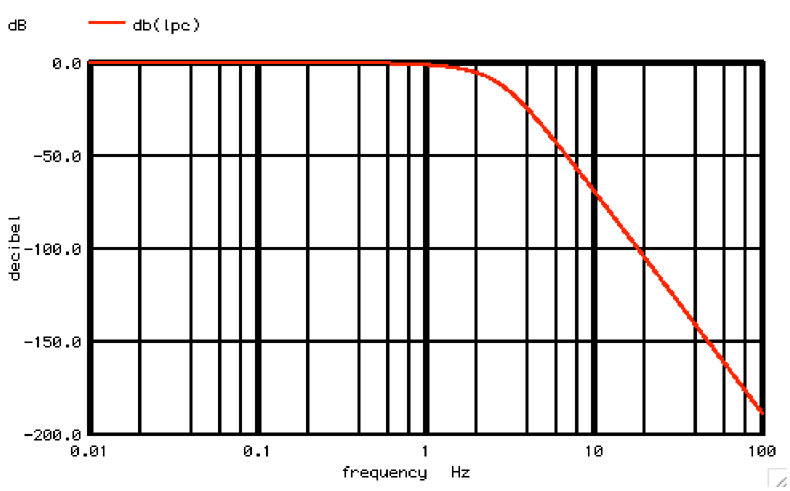
```
*V_PULSE# NODE_P NODE_N DC    VALUE  PULSE( VINIT  VPULSE TDELAY TRISE  TFALL  PWIDTH PERIOD )
V_IN      VIN    0      DC    0      PULSE( -1     1      100u   100u   100u   1      2 ) AC = 1
R0A       VIN    SUMA   .333
R1A       SUMA   HPA    1
R2A       SUMA   BPA    .54
```

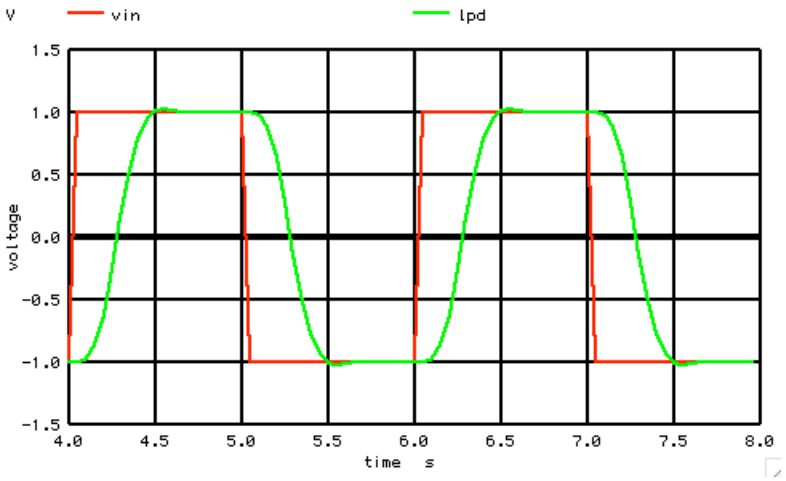
```
R3A         SUMA    LPA     .333
BOPA1A      HPA     0       V =     5*tanh(tanh((-v(SUMA))*100)*100)
XONES1A     HPA     BPA     ONE_S
XONES2A     BPA     LPA     ONE_S
R0B         LPA     SUMB    .35
R1B         SUMB    HPB     1
R2B         SUMB    BPB     .362
R3B         SUMB    LPB     .35
BOPA1B      HPB     0       V =     5*tanh(tanh((-v(SUMB))*100)*100)
XONES1B     HPB     BPB     ONE_S
XONES2B     BPB     LPB     ONE_S
R0C         LPB     SUMC    .389
R1C         SUMC    HPC     1
R2C         SUMC    BPC     .318
R3C         SUMC    LPC     .389
BOPA1C      HPC     0       V =     5*tanh(tanh((-v(SUMC))*100)*100)
XONES1C     HPC     BPC     ONE_S
XONES2C     BPC     LPC     ONE_S
Binv        LPD     0       V =     -V(LPC)
*=========Integrator_Cell=================================================
*                 C1  .1F
*                _||_______  OUT1  _______|OUT|
*         R1     | ||  _|_  |      _|_    |___|      .SUBCKT   ONE_S  IN   OUT
* |IN |_/\  _|___| /   \ |__  /   \
* |___|   \/ISUM|  BOP1 |    |BINV1 |    ---->  |IN |_ ________ _|OUT|
*      1       _|  80dB |   _|  -1  |    ---->  |___| |  1/s   | |___|
*             |  \___/    |  \___/                    |________|
*             |_____      |_____
*                  _|_          _|_        R = 1 = 1/sC    s = 1 =   2*PI*Freq
*                  ///          ///
*
.SUBCKT     ONE_S     IN      OUT
R1          IN        ISUM    1
C1          ISUM      OUT1    .1
BOP1        OUT1      0       V =     5*tanh(tanh((-v(ISUM))*100)*100)
BINV1       OUT       0       V =     -v(OUT1)
.ends
```

*=========Simple_Simulations====================

It is not hard to do AC and Transient tests.

```
=========================================================================
.control
ac          dec     500     .01     100
plot        db(lpc)
*TRAN       TSTEP   TSTOP   TSTART  TMAX
tran        50m     7.95    4       50m
```





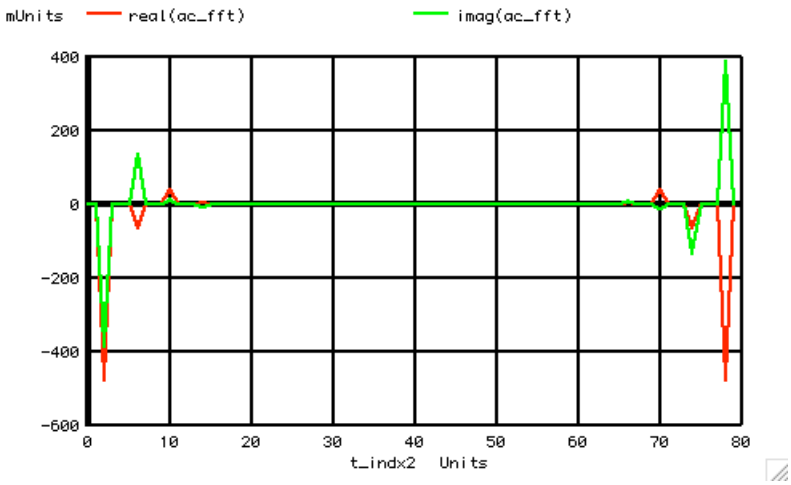
While the roll off of a Bessel is not as sharp as a Butterworth, there is no ringing at the output. It is like the output is just a time shifted input with its corners rounded off.

***=========Look_at_the_Output_Spectrum====================**

The FFT reveals that the fundamental and third and fifth harmonic got through the lowpass filter. So the bessel is not as sharp as a butterworth.

==========

```
llinearize
plot       vin    lpd
let        numb2   = length(vin)
print      numb2
let        t_indx2 = vector($&numb2)
let        ac      = lpd +j(0)
let        ac_fft  = fft(ac)
plot       real(ac_fft) imag(ac_fft) vs t_indx2
```



***=========The_Harmonics_Can_Be_Dissected====================**

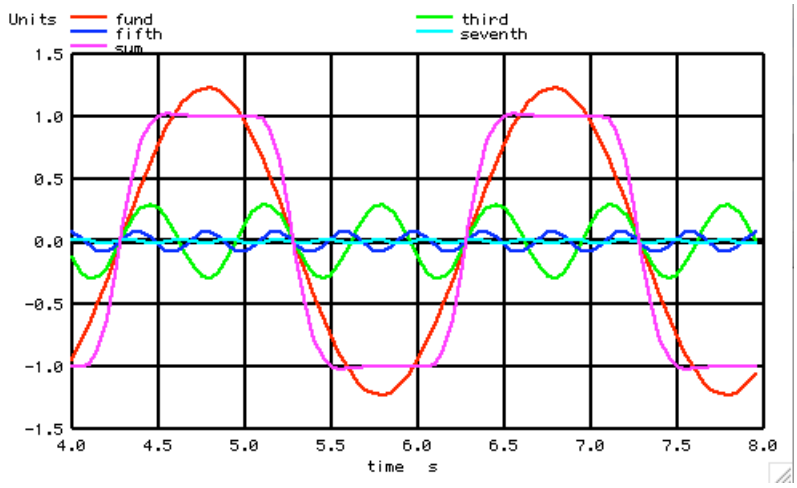
The dissection of the harmonics shows why there is no ringing.

==========

```
let        funbin = 2
```

```
let       unvect                = unitvec($&numb2)
let       fundspec              = unvect*0 +j(0)
let       fundspec[2]           = real(ac_fft[2])        +j(imag(ac_fft[2] ))
let       fundspec[numb2-2]     = real(ac_fft[numb2-2])  +j(imag(ac_fft[numb2-2] ))
let       fund                  =  ifft(fundspec)
let       thirdspec             = unvect*0 +j(0)
let       thirdspec[6]          = real(ac_fft[6])        +j(imag(ac_fft[6] ))
let       thirdspec[numb2-6]    = real(ac_fft[numb2-6])  +j(imag(ac_fft[numb2-6] ))
let       third                 = ifft(thirdspec)
let       fifthspec             = unvect*0 +j(0)
let       fifthspec[10]         = real(ac_fft[10])        +j(imag(ac_fft[10] ))
let       fifthspec[numb2-10]   = real(ac_fft[numb2-10])  +j(imag(ac_fft[numb2-10] ))
let       fifth                 = ifft(fifthspec)
let       seventhspec           = unvect*0 +j(0)
let       seventhspec[14]       = real(ac_fft[14])        +j(imag(ac_fft[14] ))
let       seventhspec[numb2-14] = real(ac_fft[numb2-14])  +j(imag(ac_fft[numb2-14] ))
let       seventh               = ifft(seventhspec)
let       sum  = fund + third + fifth + seventh
set       scale  time
plot      fund  third fifth seventh  sum

plot      fund  third fifth seventh  sum xlimit 5.2 5.7
.endc
```

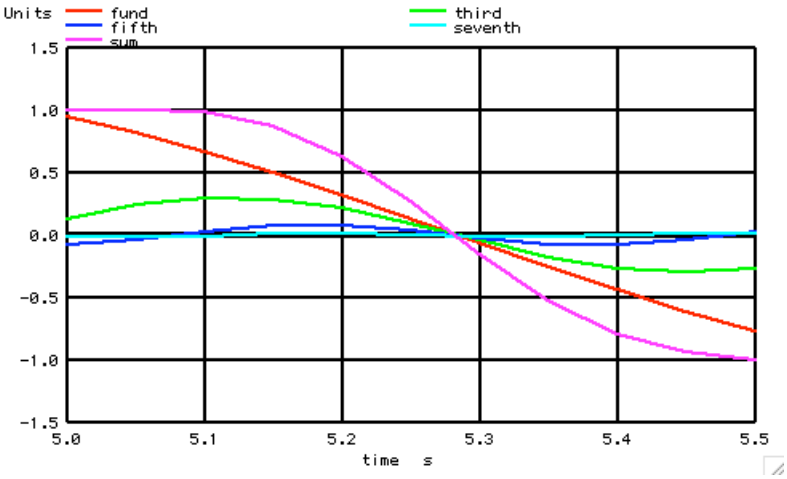


Notices that all the zero crossings are at the same time. Notice the fifth opposes the third and the same goes for the seventh.

***=========Now_Look_At_Harmonic_Delays====================**

Here is a better view of the different time delays. Not only are the zero crossing happening at the same time, the alternating harmonic are flattening the output.

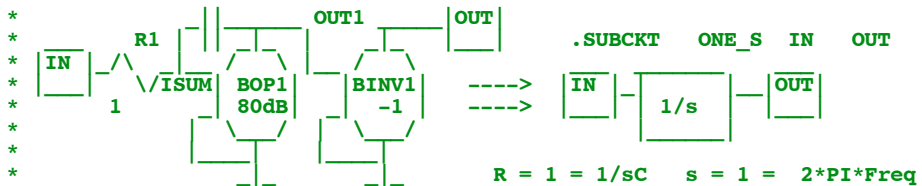
```
================================================================
plot      fund  third fifth seventh  sum xlimit 5.2 5.7
.endc
```



==========**Full_Netlist_For_Copy_Paste**==========

```
Better_Butterworth_6P_State_Variable
*  dsauersanjose@aol.com
*  www.idea2ic.com
*  8.13.10_10.02AM
*
*     R0A= 1/A0A                    R0B= 1/A0B                    R0C= 1/A0C
*    _/\  /\_                      _/\  /\__                     _/\  /\_
*   _/  \/   |     R3A= 1/A0A     |  \/      |        R3B= 1/A0B |  \/   |   R3C= 1/A0C
*  |      __|_/\  /\  /\_____|_____|      _/\  /\  /\_______|_____|     _/\  /\  /\_____________
*  |     |     \/  \/           LPA |     |   \/  \/               LPB |     \/  \/                LPC
*  |     |      R2A= 1/A1A           |     |    R2B= 1/A1B             |      R2C= 1/A1C
*  |  _  |     _/\  /\  /\_____      |     |   _/\  /\  /\_____       |     _/\  /\  /\_____      |
* /VIN\ |      \/  \/  |            |     |    \/  \/  |             |      \/  \/  |            |
* \___/ |      R1A= 1  |            |     |    R1 = 1  |             |      R1C= 1  |            |
*  |    |     _/\  /\_ |      BPA   |     |   _/\  /\_ |      BPB    |     _/\  /\_ |      BPC   |
*  ///  |      \/  |   |            |     |    \/  |   |             |      \/  |   |            |
*       |  |\  HPA |XONES1A|XONES2A  |     |  |\  HPB |XONES1B|XONES2B  |     |  |\  HPC |XONES1C|XONES2C  |
*       |  -\    |  ___  |  ___  |  |     |  |  -\    |  ___  |  ___  |  |     |  |  -\    |  ___  |  ___  |  |
*      SUMA|  \___|_|1/s|_|_|1/s|_|  |     |  |  \___|_|1/s|_|_|1/s|_|  |     |  |  \___|_|1/s|_|_|1/s|_|  |
*          |  /   |___|   |___|    SUMB    |  |  /   |___|   |___|    SUMC    |  |  /   |___|   |___|
*          |__|+/ OPAA              |__|+/ OPAB              |__|+/ OPAC
*             |/                       |/                       |/
*
*            ///                     ///                     ///
*      Butterworth terms
*      (s^2 + 0.5176s + 1)(s^2 + 1.4142s + 1)(s^2 + 1.9319s + 1)
.OPTIONS  GMIN=1e-18    METHOD=euler  srcsteps = 1  gminsteps = 1
*======== ====== ====== ====== ====== ====== ====== ====== ======
*V_PULSE# NODE_P NODE_N DC     VALUE  PULSE( VINIT  VPULSE TDELAY TRISE  TFALL  PWIDTH PERIOD )
V_IN      VIN    0      DC     0      PULSE( -1      1      100u   100u   100u   1      2 ) AC = 1
R0A       VIN    SUMA   1
R1A       SUMA   HPA    1
R2A       SUMA   BPA    1.9319
R3A       SUMA   LPA    1
BOPA1A    HPA    0      V =    5*tanh(tanh((-v(SUMA))*100)*100)
XONES1A   HPA    BPA    ONE_S
XONES2A   BPA    LPA    ONE_S
R0B       LPA    SUMB   1
R1B       SUMB   HPB    1
R2B       SUMB   BPB    .707
R3B       SUMB   LPB    1
BOPA1B    HPB    0      V =    5*tanh(tanh((-v(SUMB))*100)*100)
XONES1B   HPB    BPB    ONE_S
XONES2B   BPB    LPB    ONE_S
R0C       LPB    SUMC   1
R1C       SUMC   HPC    1
R2C       SUMC   BPC    .5176
R3C       SUMC   LPC    1
BOPA1C    HPC    0      V =    5*tanh(tanh((-v(SUMC))*100)*100)
XONES1C   HPC    BPC    ONE_S
XONES2C   BPC    LPC    ONE_S
Binv      LPD    0      V =    -V(LPC)

*==========Integrator_Cell==================================================
*            C1 .1F                ___
```




```
.SUBCKT     ONE_S    IN      OUT
R1          IN       ISUM    1
C1          ISUM     OUT1    .1
BOP1        OUT1     0       V =    5*tanh(tanh((-v(ISUM))*100)*100)
BINV1       OUT      0       V =    -v(OUT1)
.ends



*==========A_Bessel_is_Best_for_Low_Phase_Distortion====
.control
ac          dec    500     .01   100
plot        db(lpc)
*TRAN       TSTEP  TSTOP  TSTART TMAX
tran        50m    7.95   4      50m


linearize
plot        vin    lpd
let         numb2   = length(vin)
print       numb2
let         t_indx2 = vector($&numb2)
let         ac      = lpd +j(0)
let         ac_fft  = fft(ac)
plot        real(ac_fft) imag(ac_fft) vs t_indx2

let         funbin = 2
let         unvect                = unitvec($&numb2)
let         fundspec              = unvect*0 +j(0)
let         fundspec[2]           = real(ac_fft[2])        +j(imag(ac_fft[2] ))
let         fundspec[numb2-2]     = real(ac_fft[numb2-2])  +j(imag(ac_fft[numb2-2] ))
let         fund                  =  ifft(fundspec)
let         thirdspec             = unvect*0 +j(0)
let         thirdspec[6]          = real(ac_fft[6])        +j(imag(ac_fft[6] ))
let         thirdspec[numb2-6]    = real(ac_fft[numb2-6])  +j(imag(ac_fft[numb2-6] ))
let         third                 = ifft(thirdspec)
let         fifthspec             = unvect*0 +j(0)
let         fifthspec[10]         = real(ac_fft[10])        +j(imag(ac_fft[10] ))
let         fifthspec[numb2-10]   = real(ac_fft[numb2-10])  +j(imag(ac_fft[numb2-10] ))
let         fifth                 = ifft(fifthspec)
let         seventhspec           = unvect*0 +j(0)
let         seventhspec[14]       = real(ac_fft[14])        +j(imag(ac_fft[14] ))
let         seventhspec[numb2-14] = real(ac_fft[numb2-14])  +j(imag(ac_fft[numb2-14] ))
let         seventh               = ifft(seventhspec)
let         sum  = fund + third + fifth + seventh
set         scale  time
plot        fund  third fifth seventh  sum

plot        fund  third fifth seventh  sum xlimit 5.2 5.7
.endc



.end
```