ANSOFT CORPORATION PRESENTS:

CONVERGE

AN APPLICATIONS WORKSHOP FOR HIGH-PERFORMANCE DESIGN

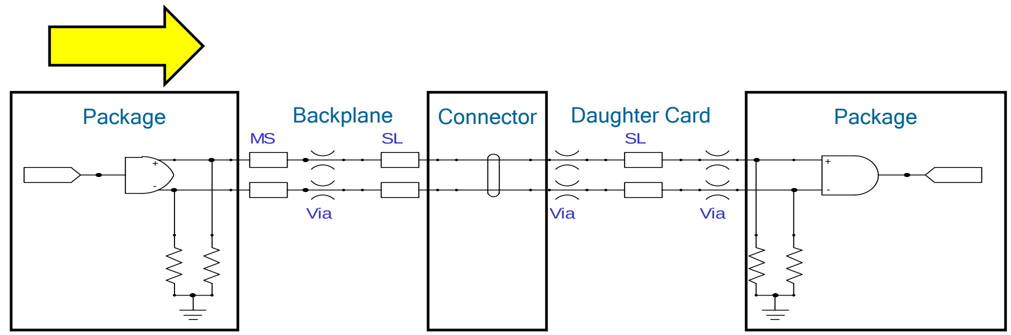
Multi-Gigabit Driver Slew Adjustment Using Embedded Printed Circuit Board Filter

Jim DeLap

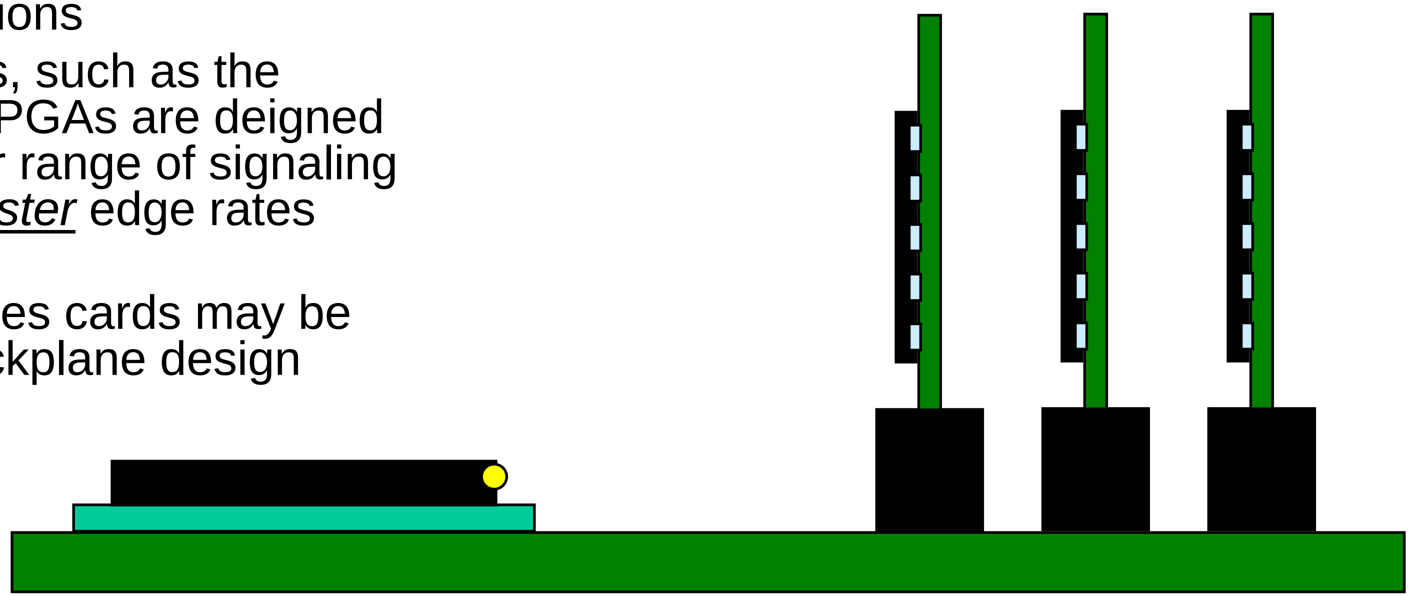
Ansoft Corporation

Motivation

- The Good News:
- Carrier-class networking platforms are built to be upgraded over time
- New line cards can be added to existing backplanes to enhance functionality



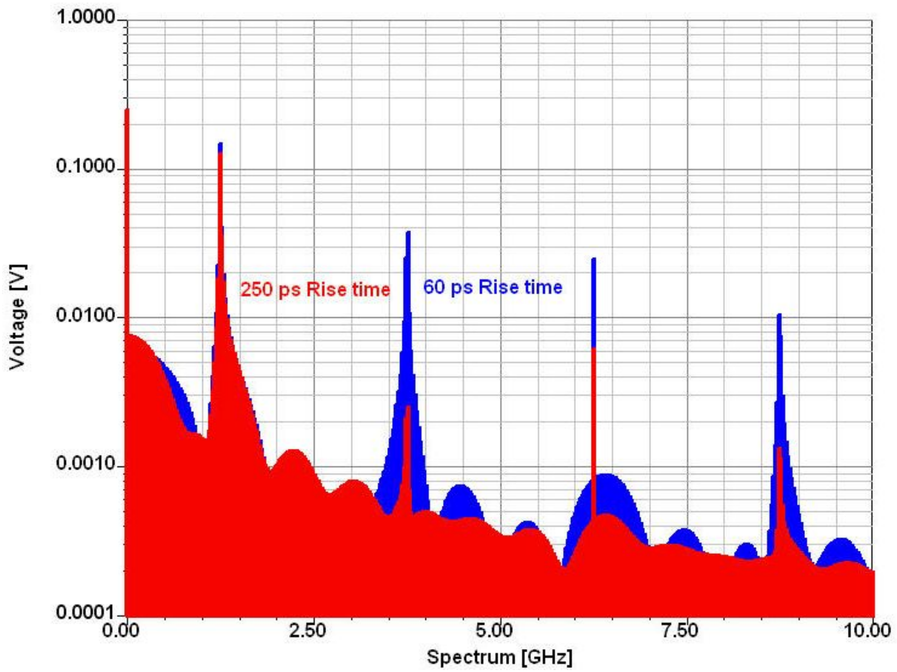
- The Not-so-Good news:
- New lines cards typically use latest generation silicon to meet newer protocol specifications
- These newer parts, such as the Xilinx Virtex™ 4 FPGAs are deigned to target a broader range of signaling rates, and have <u>*faster*</u> edge rates than older parts
- Performance of lines cards may be beyond that of backplane design



Faster ≠ Better

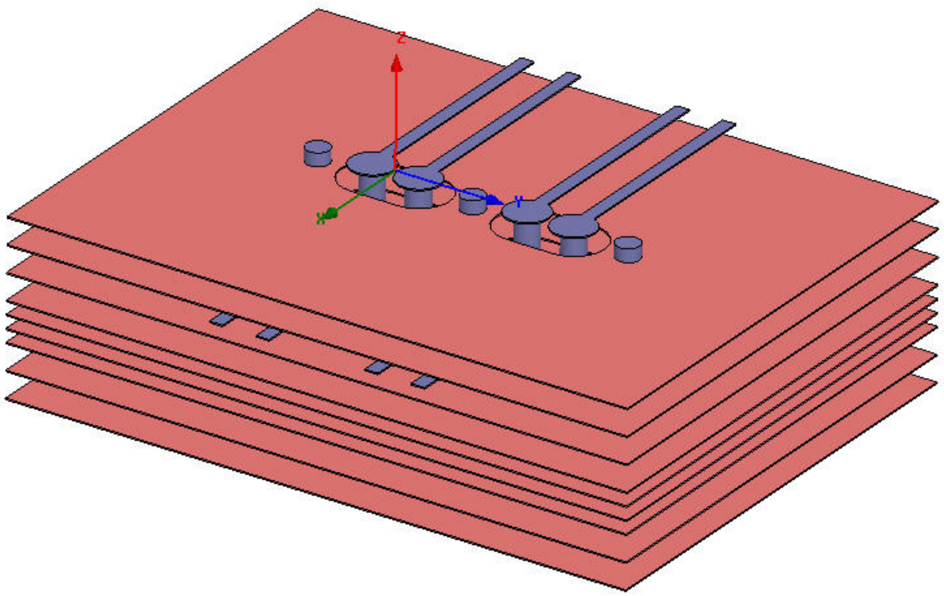
- Newer Silicon chips = Faster
- Faster is better, right ??

- Faster edge rate → increased high frequency content
- Channel is a LPF
- Increased high frequencies → more energy reflected → more crosstalk



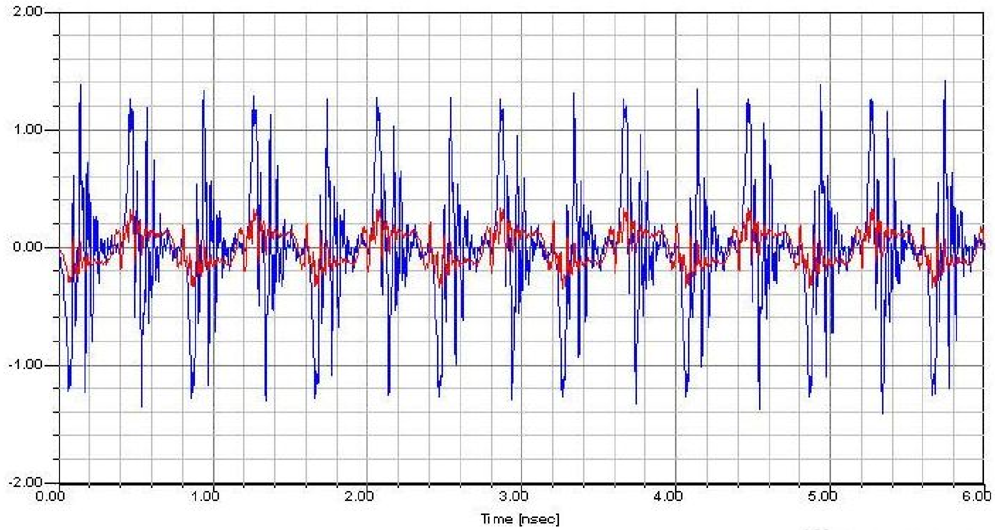
Test of crosstalk

- Testing the concept of faster edge rates being worse for signal integrity
- Take a simple model of adjacent via pairs



- Excite this with a differential pulse on one input and examine the crosstalk

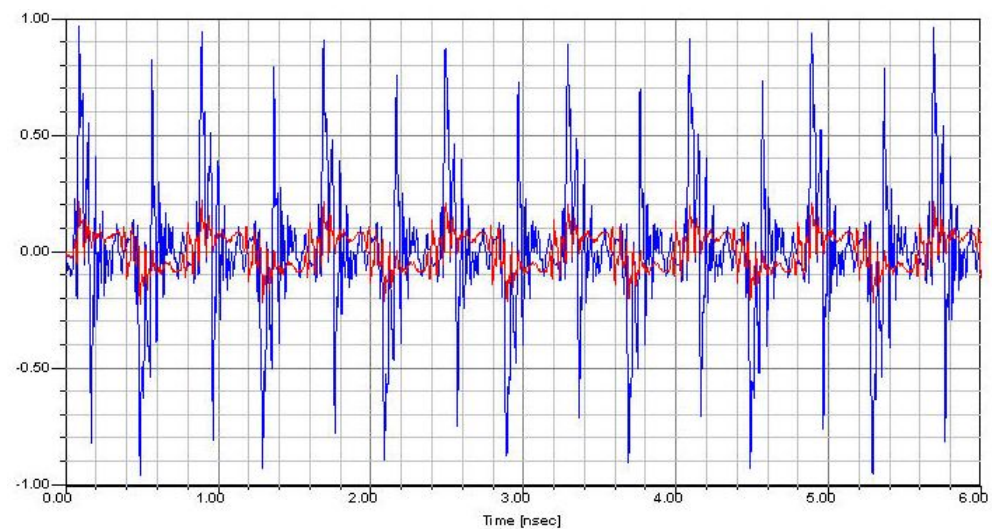
Crosstalk Test



NEXT (mV)

~ 5X worse crosstalk

FEXT (mV)

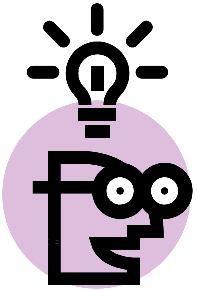


~ 9X worse crosstalk

- Blue trace – 60 ps edge rate
- Red trace – 250 ps edge rate

How do we make faster look like slower ?

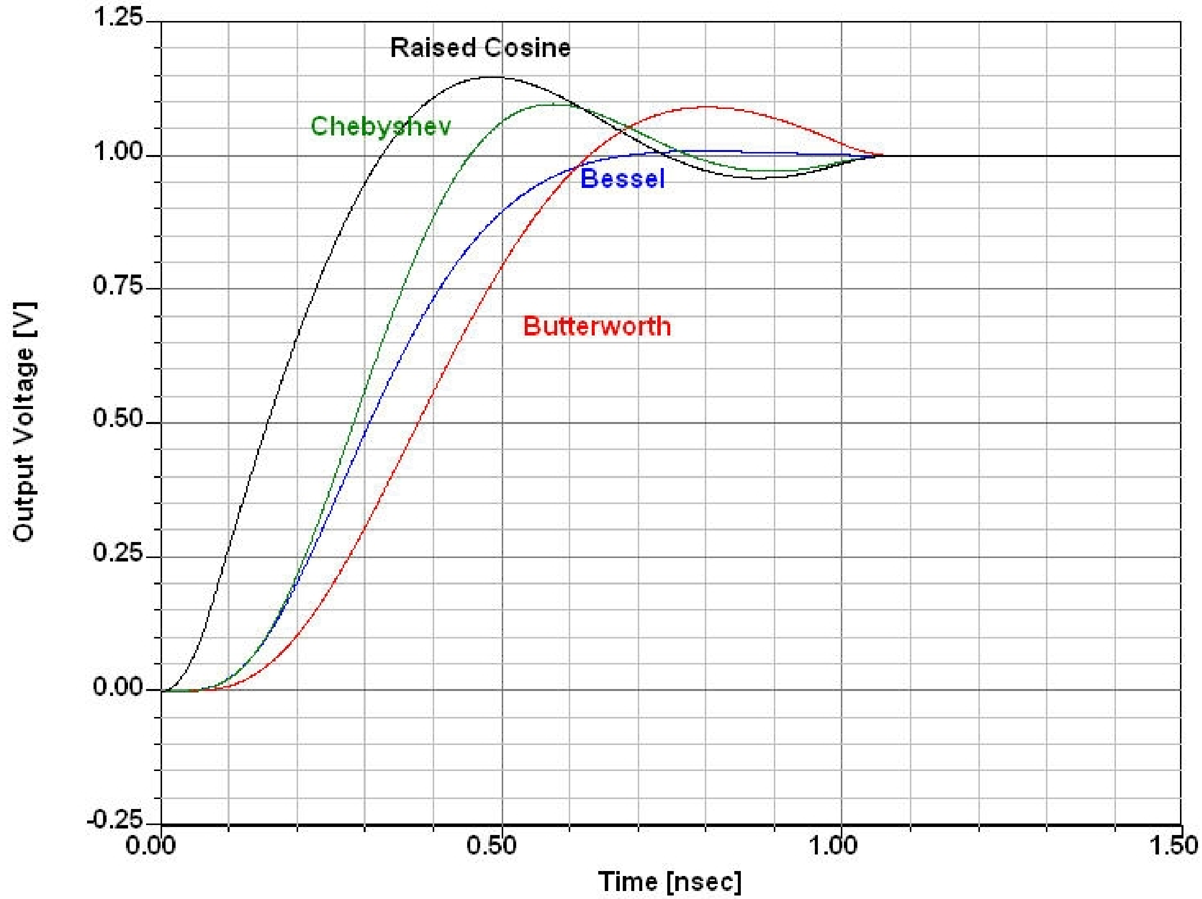
- Channel model is essentially a LPF
- Why don't we design a LPF on the transmitter outputs to slow them down ?



- What characteristics should we look for ?
  - Slower output edge rate
  - No other signal distortion (overshoot, undershoot, etc.)

Filter Comparisons

- Various filter topologies with similar settings
  - Order = 3
  - Fc = 1 GHz
- Bessel
  - < 1% overshoot
  - Also known as maximally flat group delay



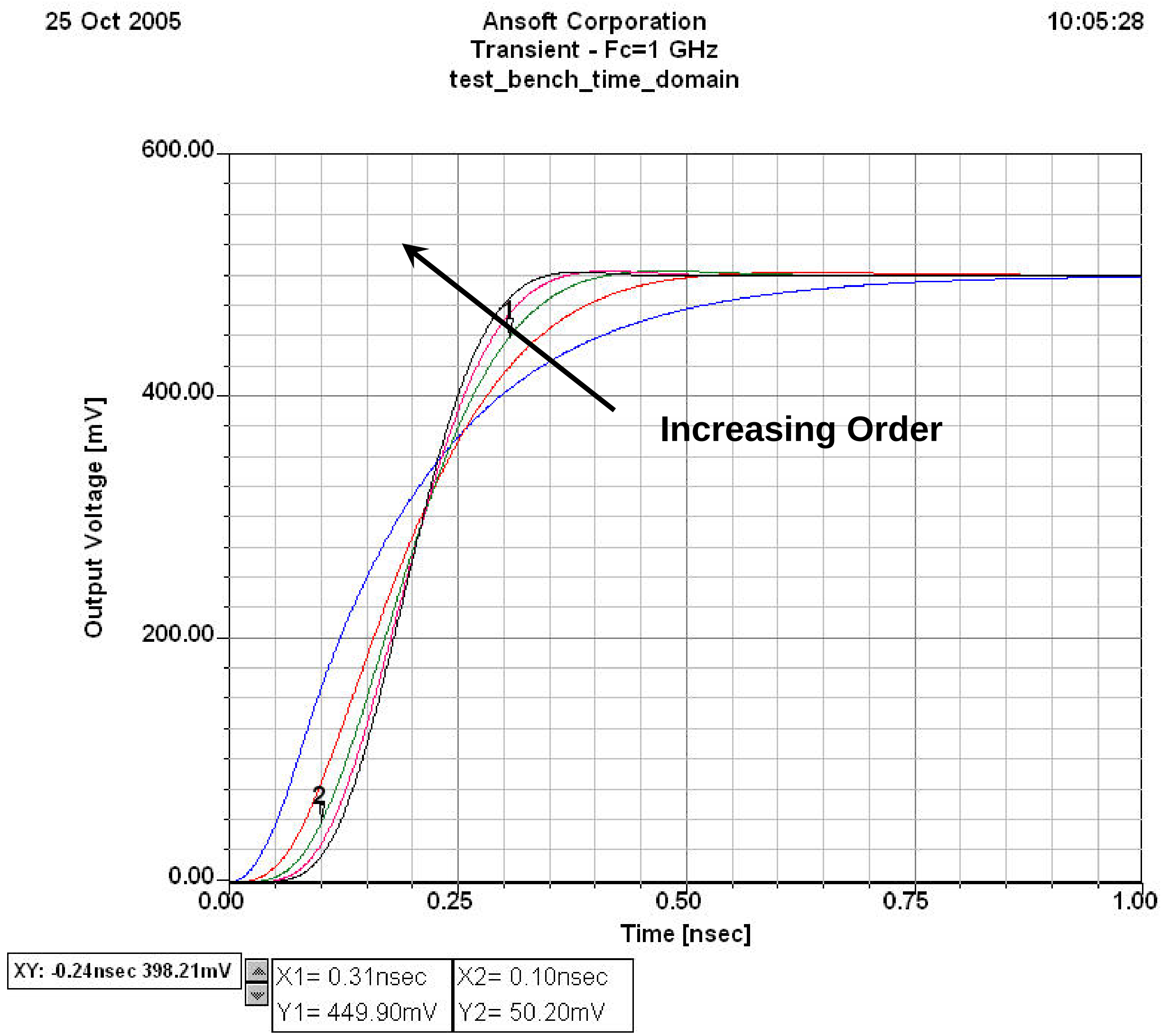
Bessel Filter Design

- What order ?
- What cutoff frequency ?

- Principal design characteristic is output edge rate
  - Xilinx Virtex™ 4 FPGA – Rocket IO™ ~60 ps edge rate
  - Existing infrastructure needs > 250 ps edge rate
- Use Ansoft Designer to investigate various combinations

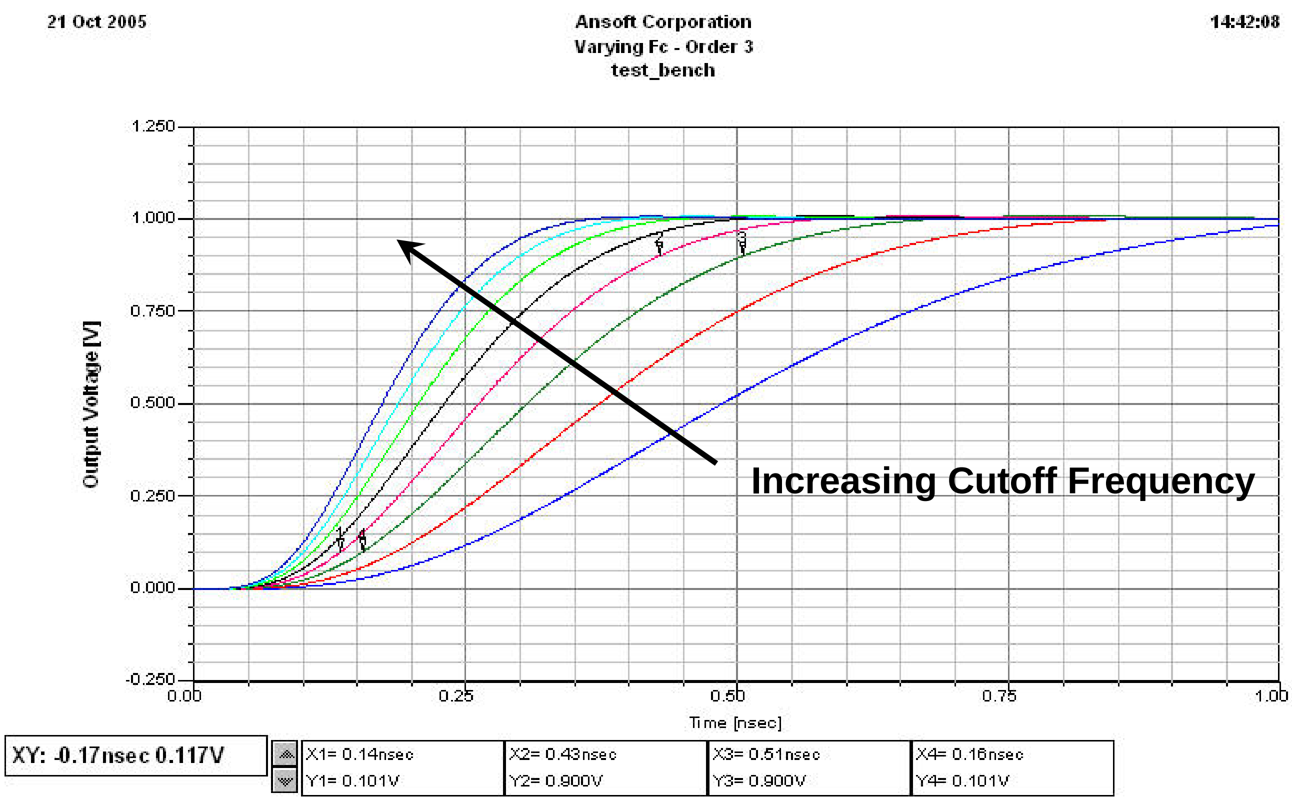
Bessel filter Design
Vary the order

- Fc = 1 GHz
- Order = 3
- Tr = 210 ps



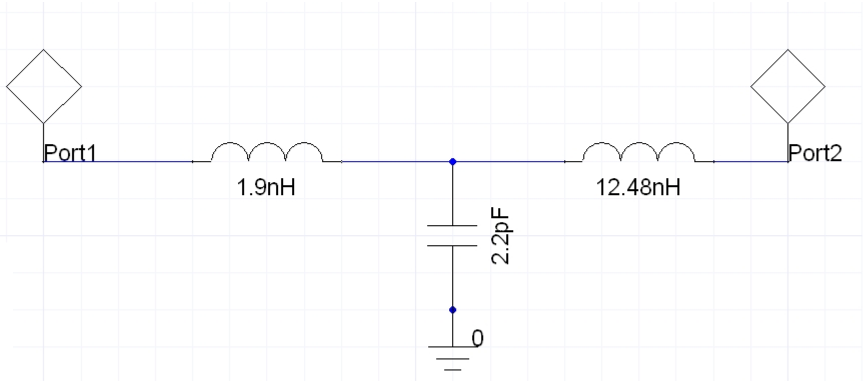
Bessel filter Design
Vary the cutoff frequency

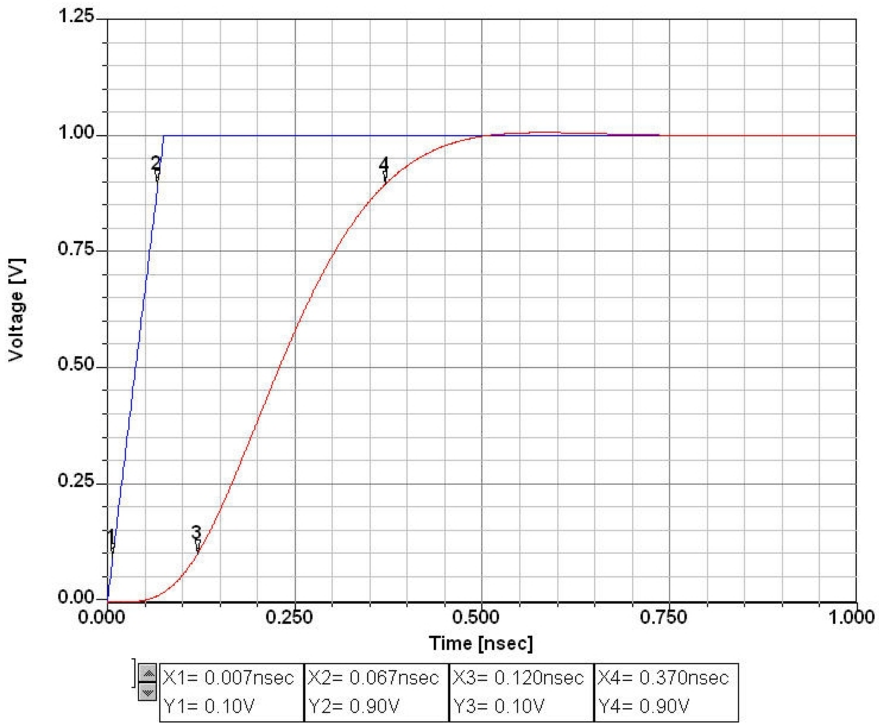
- Order = 3
- Fc = 800 MHz
- Tr = 250 ps

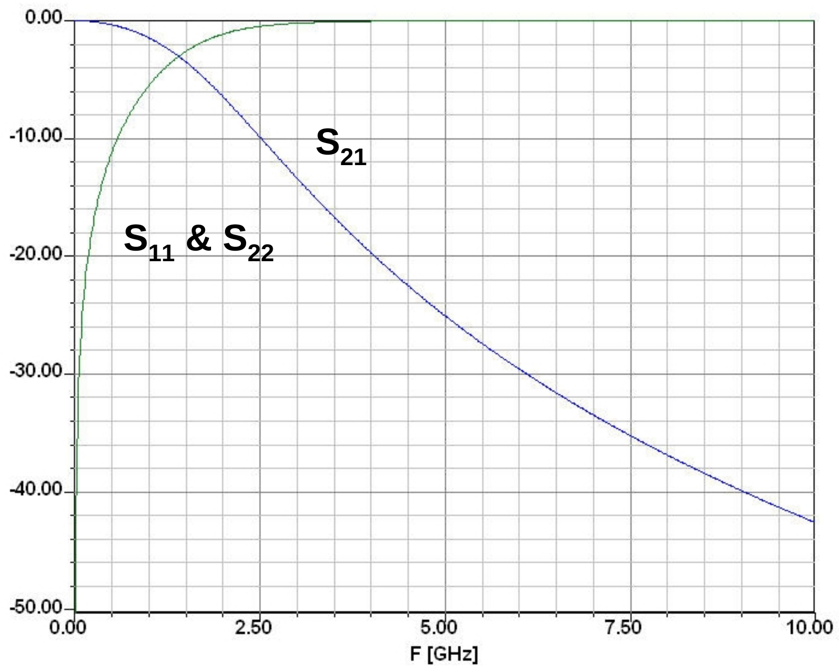


Bessel filter Design
Lumped element implementation

- Order = 3
- Fc = 800 MHz







Bessel filter Design
TLine implementation

- Use transmission lines in a high-low-high configuration
- From transmission line theory, in a stepped impedance low pass filter:

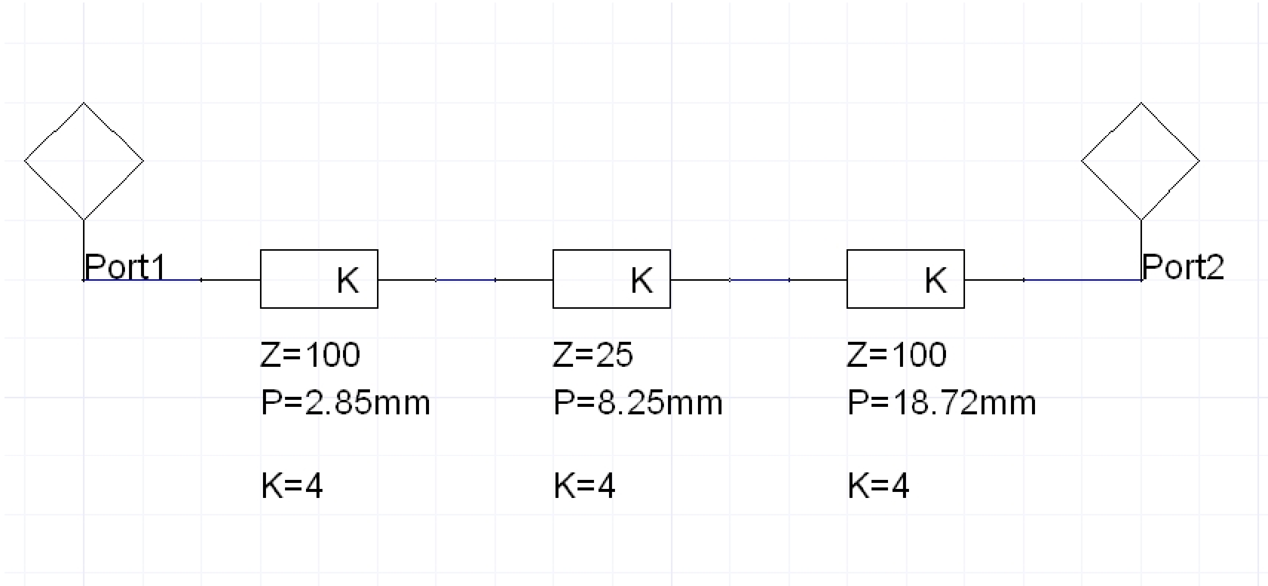
$$l = \frac{L \cdot c_o}{Z_h \cdot \sqrt{\varepsilon_r}}$$

$$l = \frac{Z_l \cdot C \cdot c_0}{\sqrt{\varepsilon_r}}$$

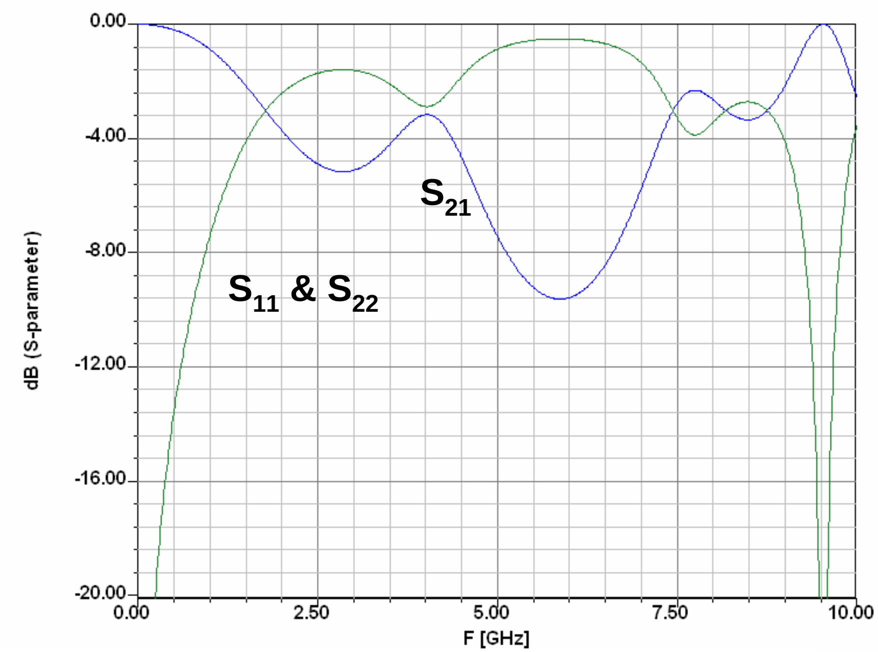
- Where:
  - L is the desired series inductance
  - C is the desired shunt capacitance
  - C_0 is the speed of light
  - ε_r is the effective dielectric constant of the transmission line
  - Z_h is the characteristic impedance of the high section
  - Z_l is the characteristic impedance of the low section

Bessel filter Design
TLine implementation

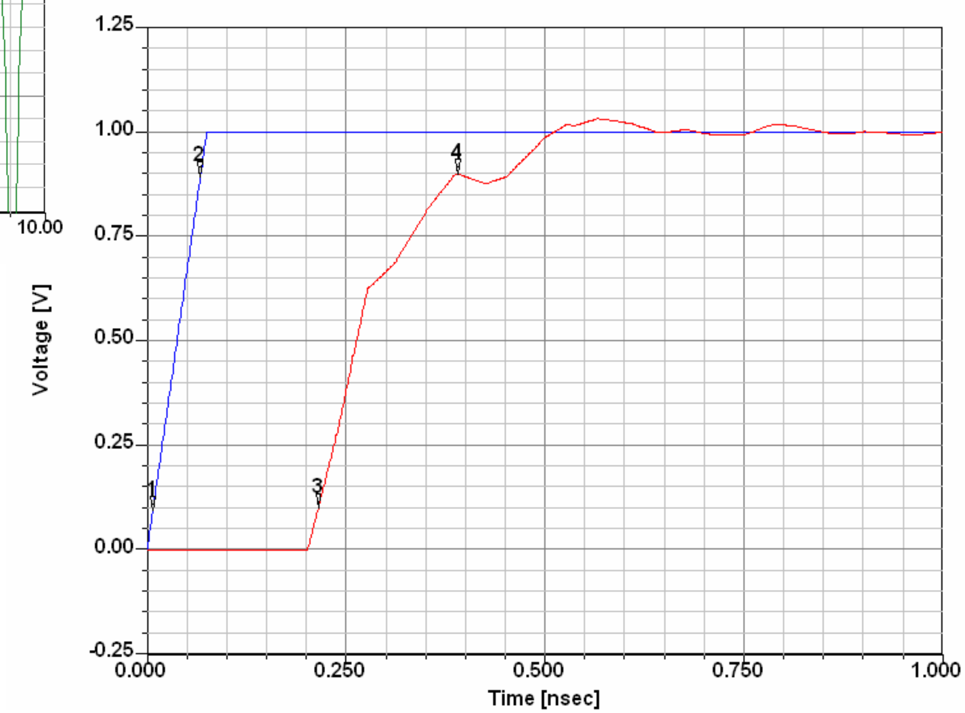
- Using the lumped values, and implementing with transmission lines, we get: (assuming $\varepsilon_r = 4$)
  - Z_h = 100 Ohms → l_1 = 2.85 mm
  - Z_l = 25 Ohms → l_2 = 8.25 mm
  - Z_h = 100 Ohms → l_3 = 18.72 mm



Bessel filter Design
TLine implementation - Results

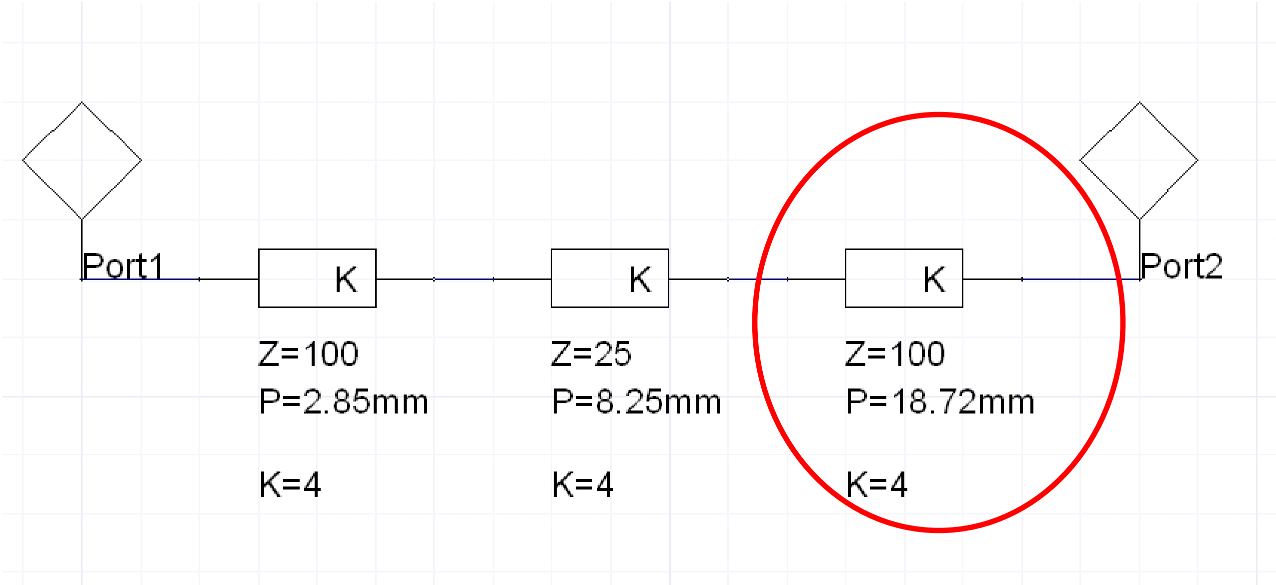


What Happened !!!



Bessel filter Design
TLine implementation - Results

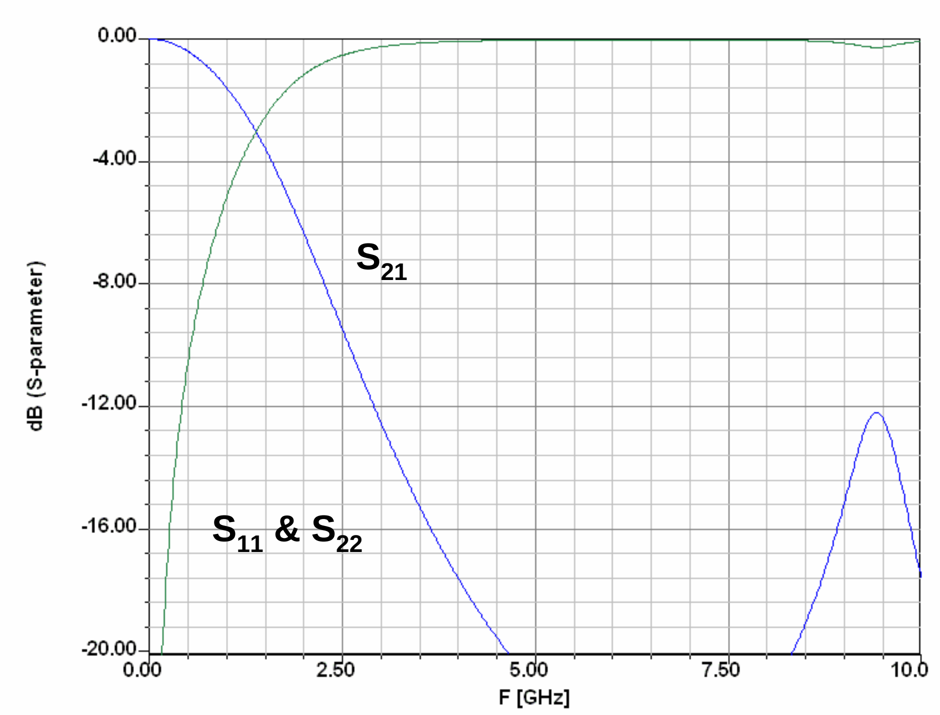
- Here's what happened:



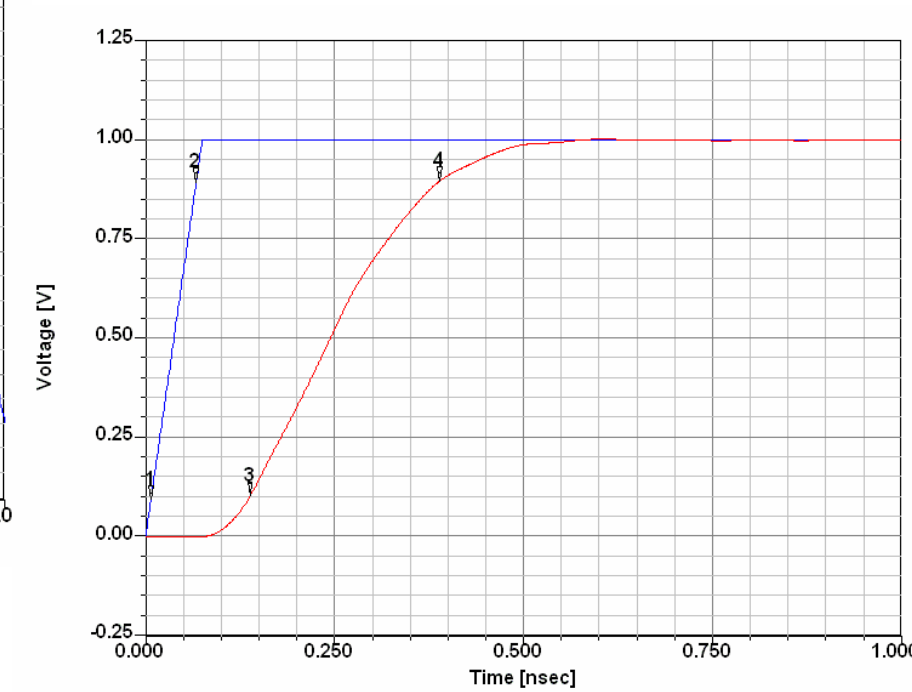
- Big long section of transmission line causes lower frequency resonances

Bessel filter Design
TLine implementation - Results

- What happens with transmission line filter when we replace that line with the lumped inductor ?



We'll worry about this a little later

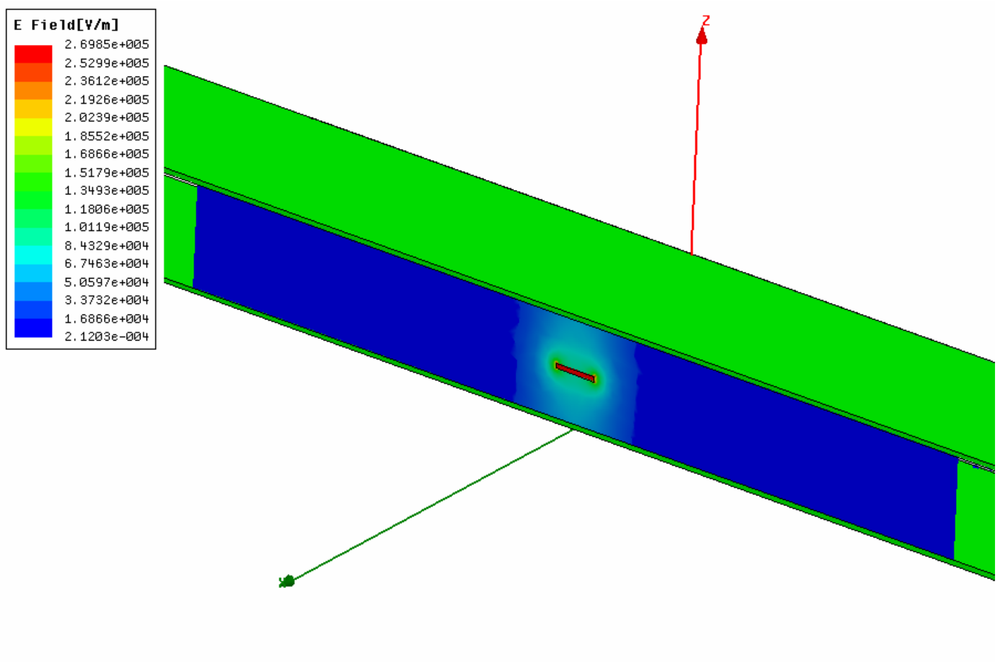


Bessel filter Design
PCB Impementation

- Assumptions
  - Multi-layer process
  - FR4 materials
  - Signal layers are routed in stripline
- First series inductor and shunt capacitor are easy to synthesize
- Utilize HFSS for transmission line calculations and optimizations

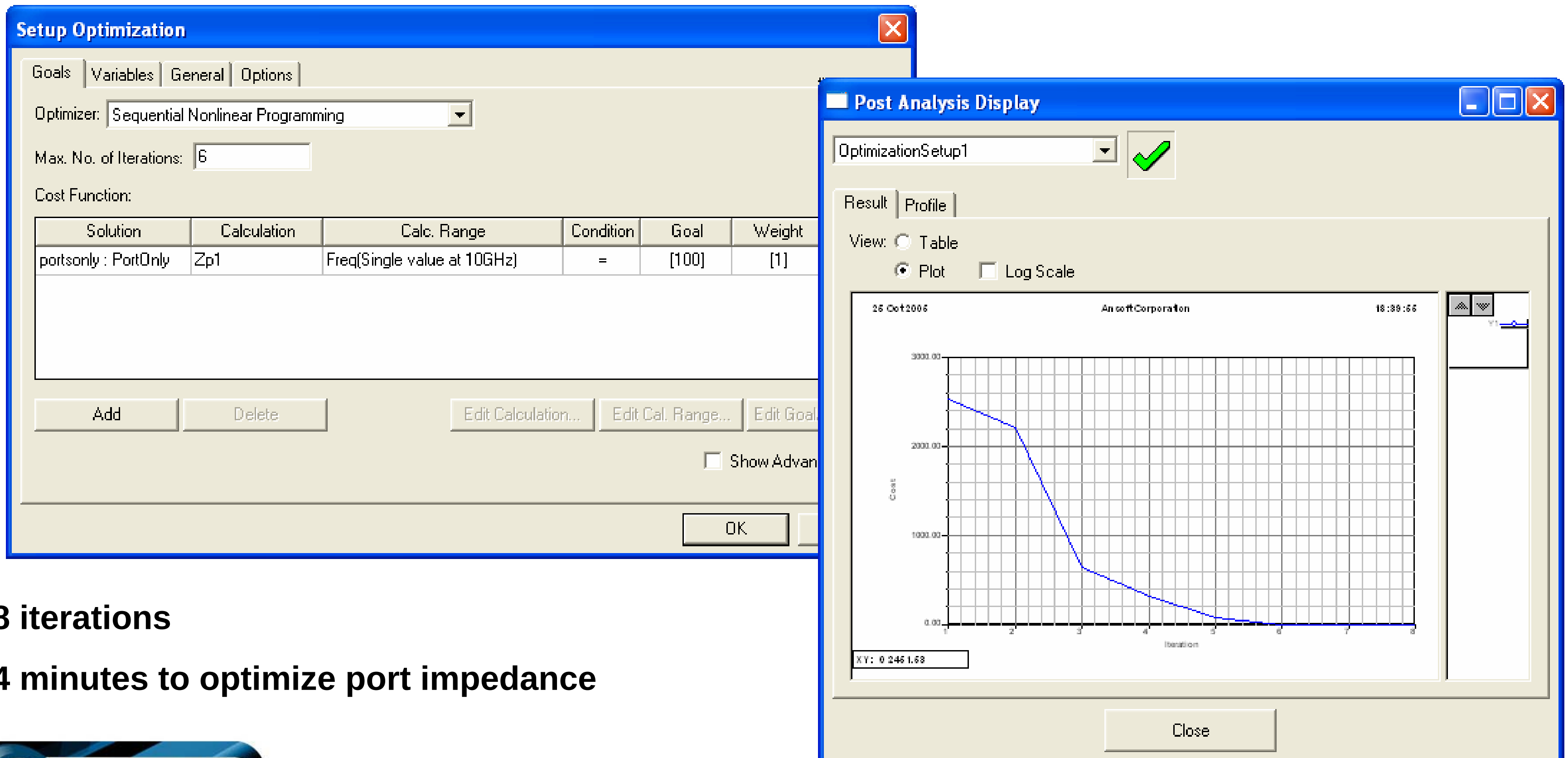
HFSS
simple stripline

- Create parameterized section of stripline
- Utilize “ports only” optimization of Impedance to quickly obtain correct dimensions



HFSS
Optimize stripline impedance

- Use new Sequential Nonlinear Programming optimization algorithm to optimize impedance

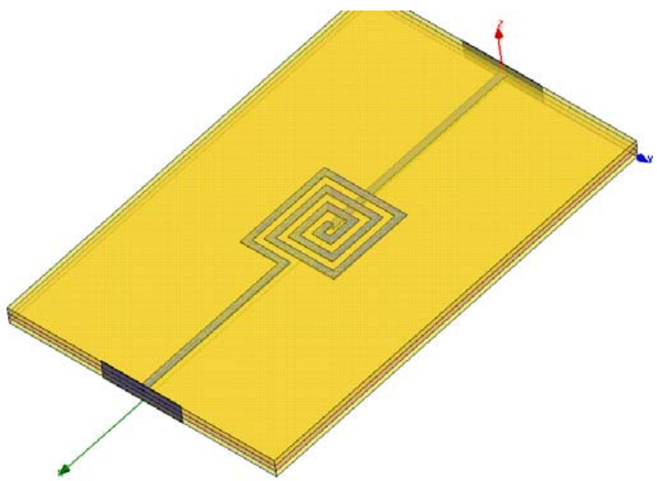


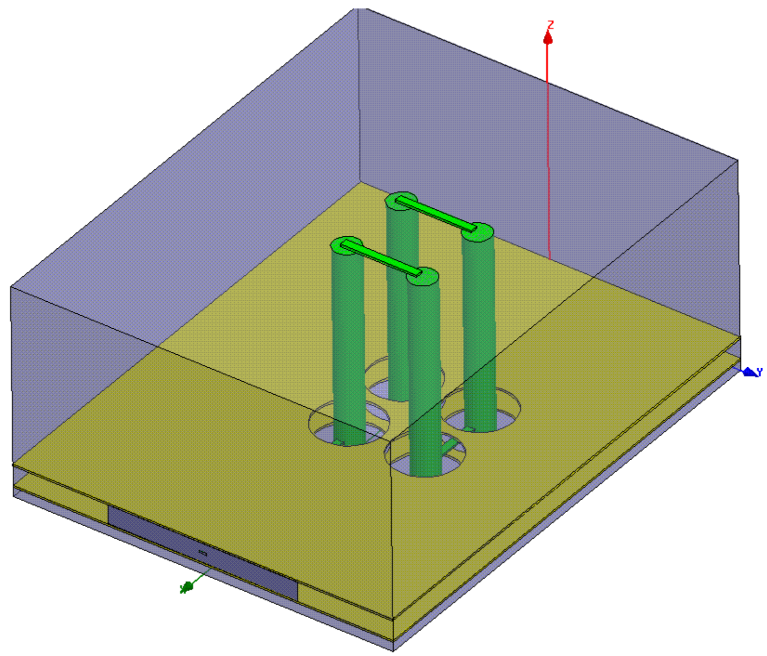
8 iterations

4 minutes to optimize port impedance

HFSS - Big L

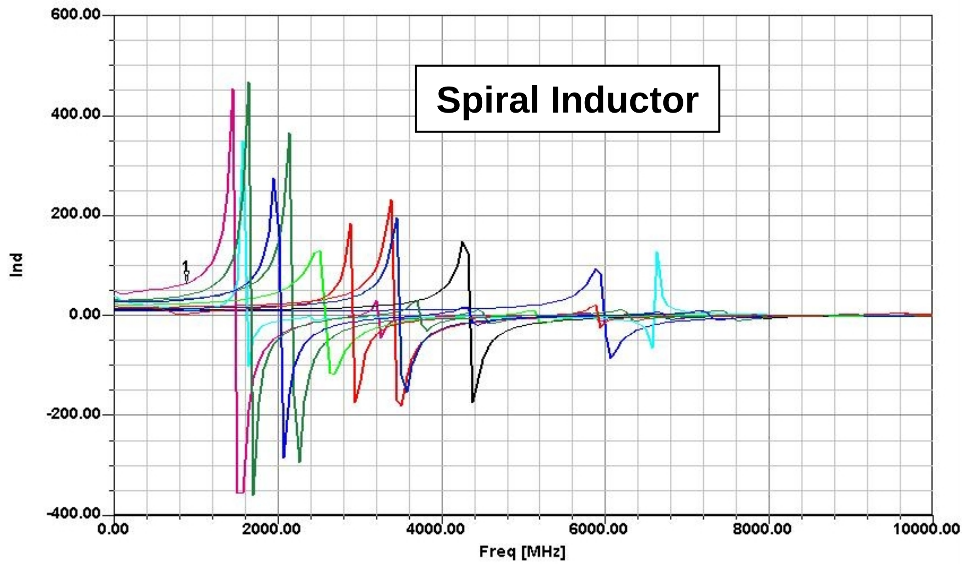
- How do we synthesize the 12.48 nH inductor ?
  - Try simulating a spiral inductor in stripline
  - Try using via inductance

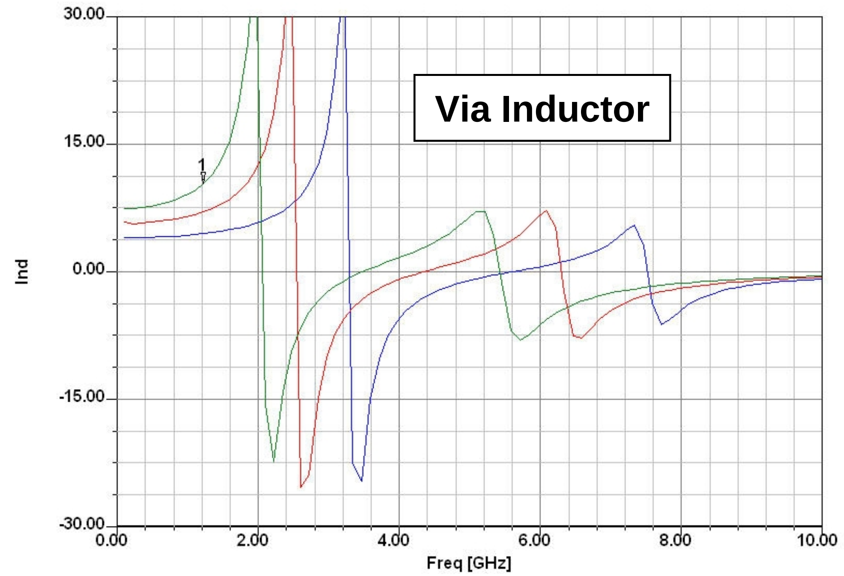




HFSS problems with big L

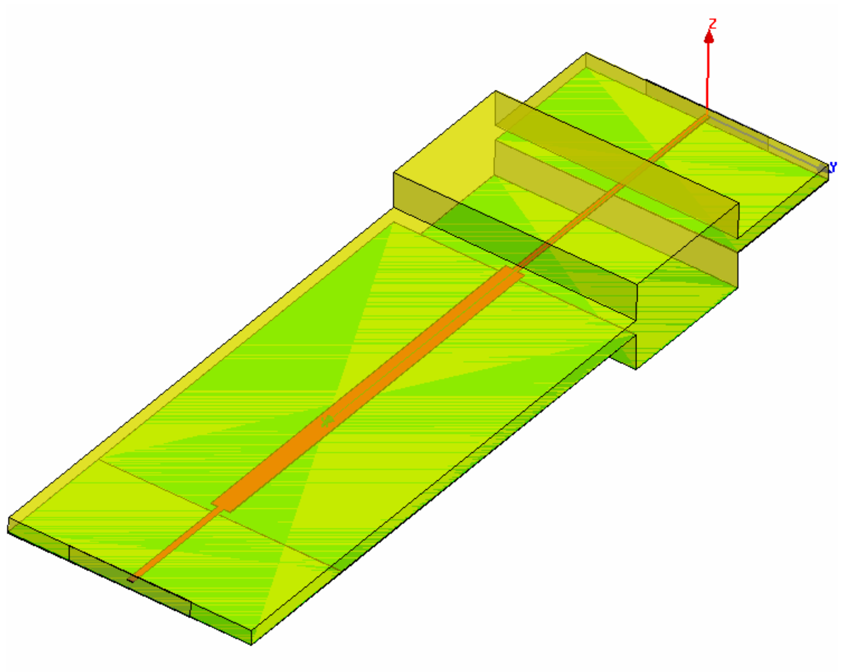
- Both types of printed inductors have the problem that as you increase length to increase inductance, resonances occur, degrading broadband performance



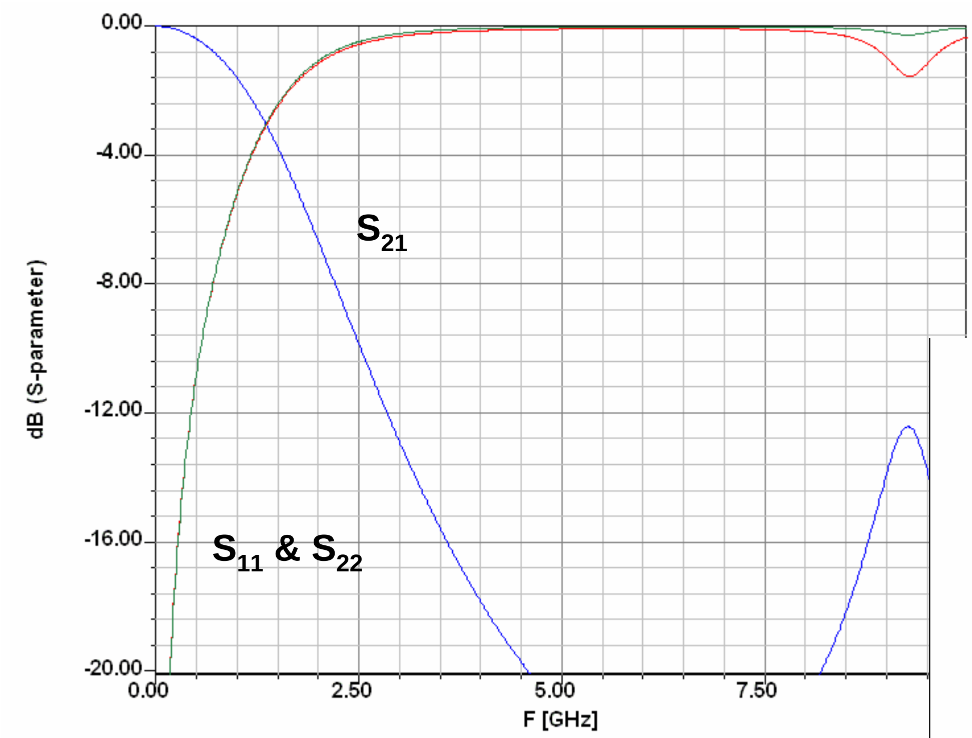


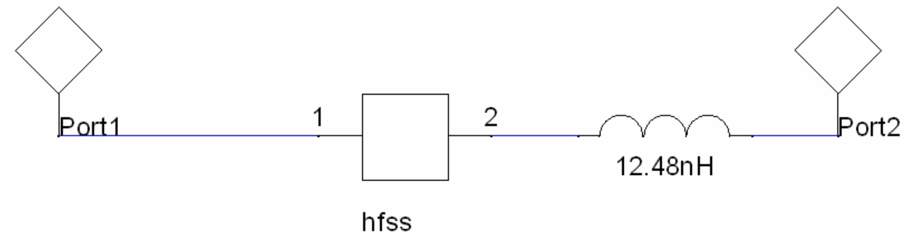
HFSS series L / shunt C

- Increased separation between ground planes to synthesize 100 Ohms
- This can be manufactured by pulling ground planes on nearby layers in real stackup

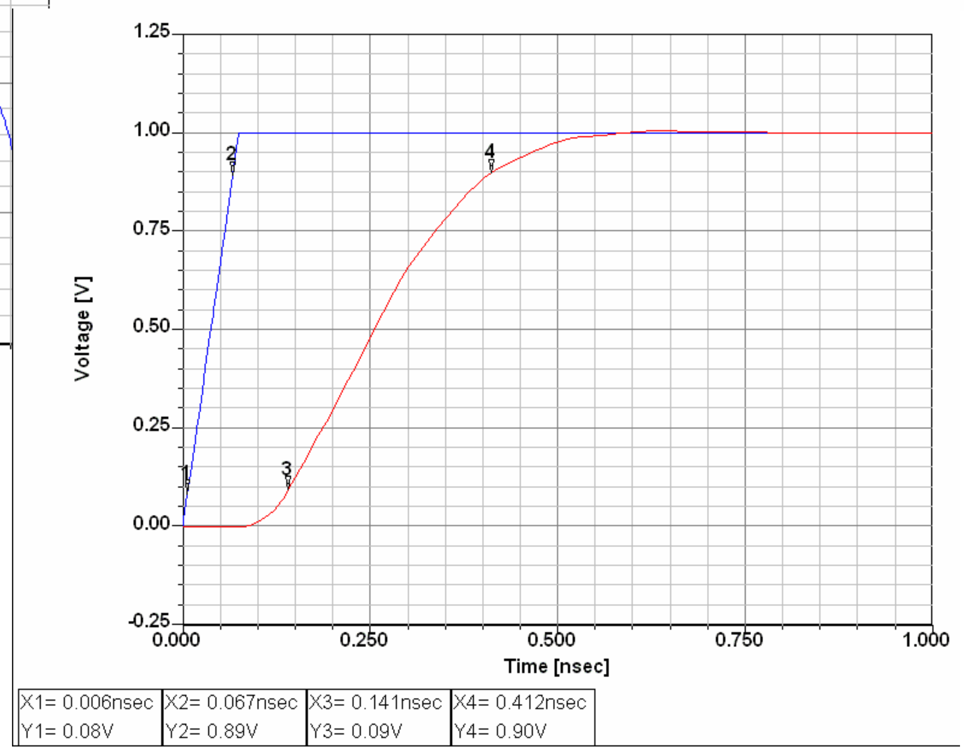


HFSS series L / shunt C
Results simulated in Designer





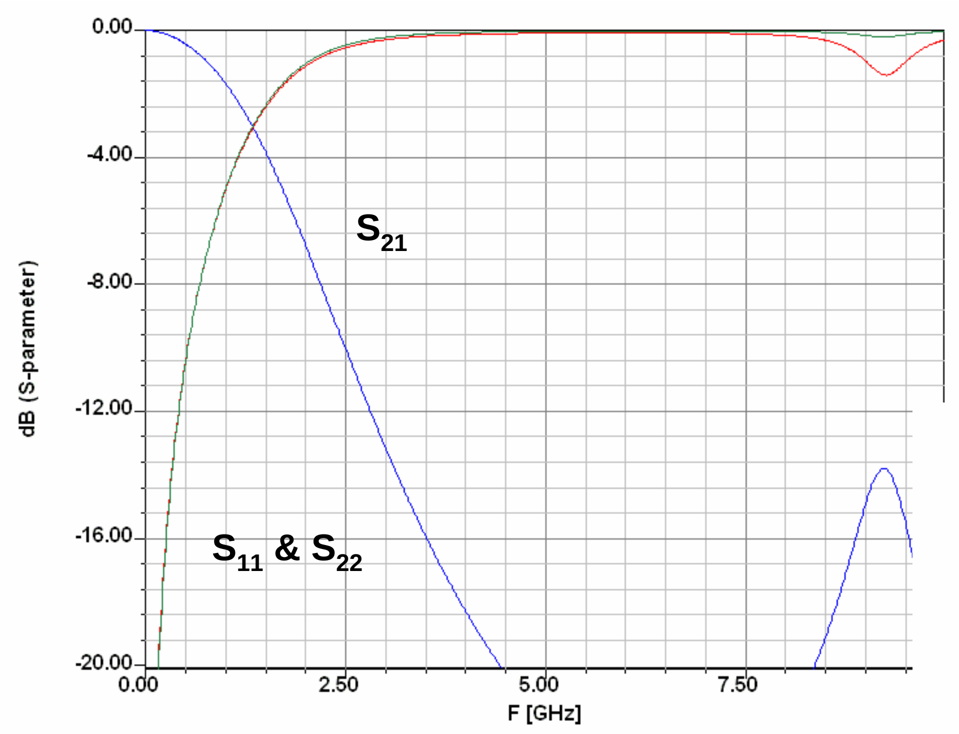
Output Rise Time = 270 ps

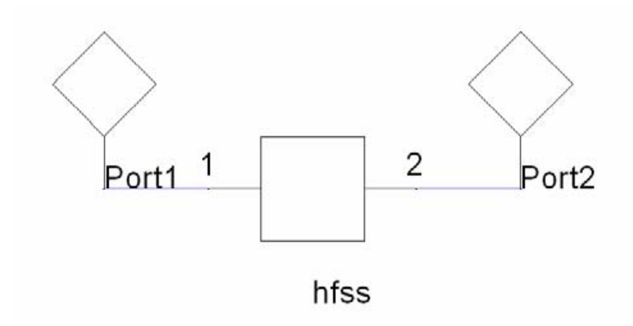


HFSS
Entire filter with Lumped RLC

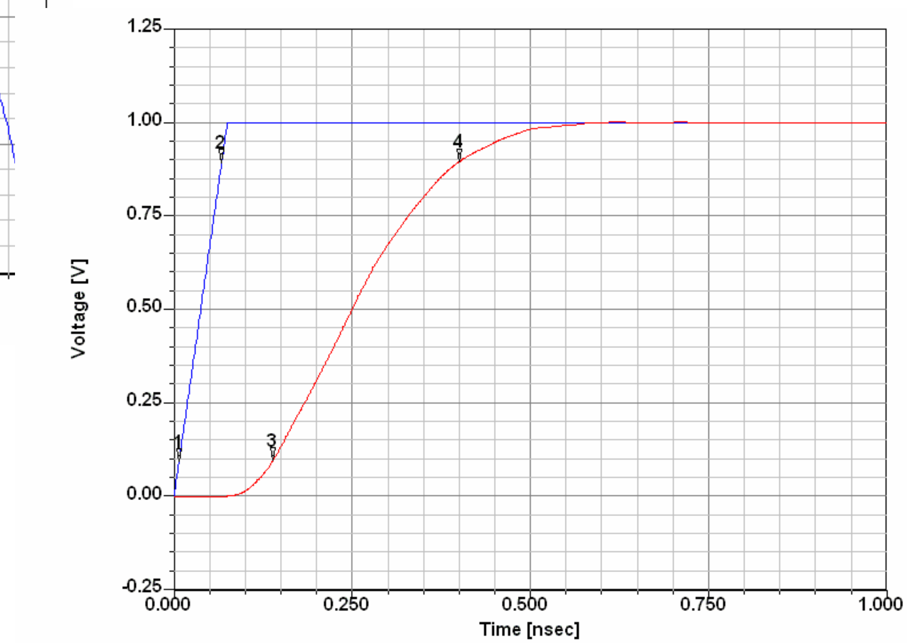
- Added LumpedRLC boundary condition inside of HFSS with a value of 12.48 nH

Entire filter with Lumped RLC Results simulated in Designer





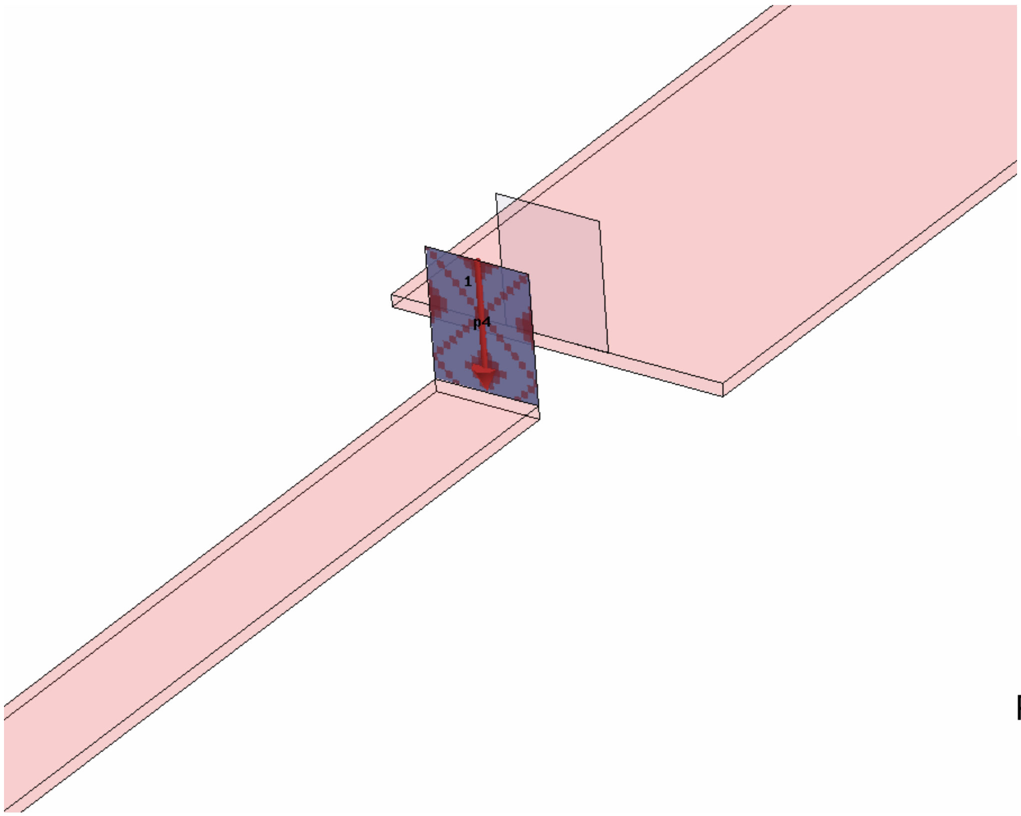
Output Rise Time = 260 ps

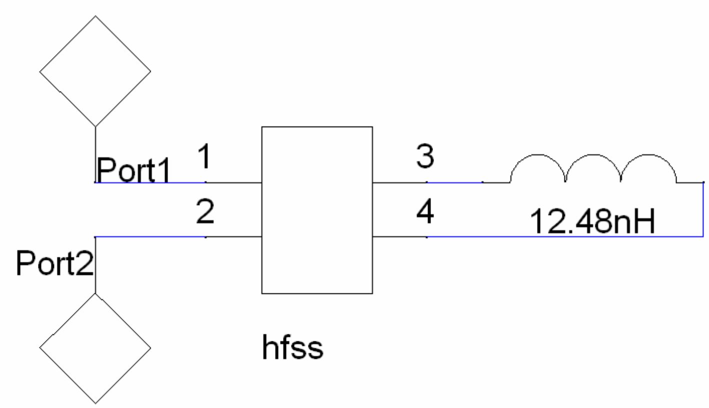


X1= 0.007nsec	X2= 0.067nsec	X3= 0.139nsec	X4= 0.400nsec
Y1= 0.09V	Y2= 0.89V	Y3= 0.10V	Y4= 0.90V

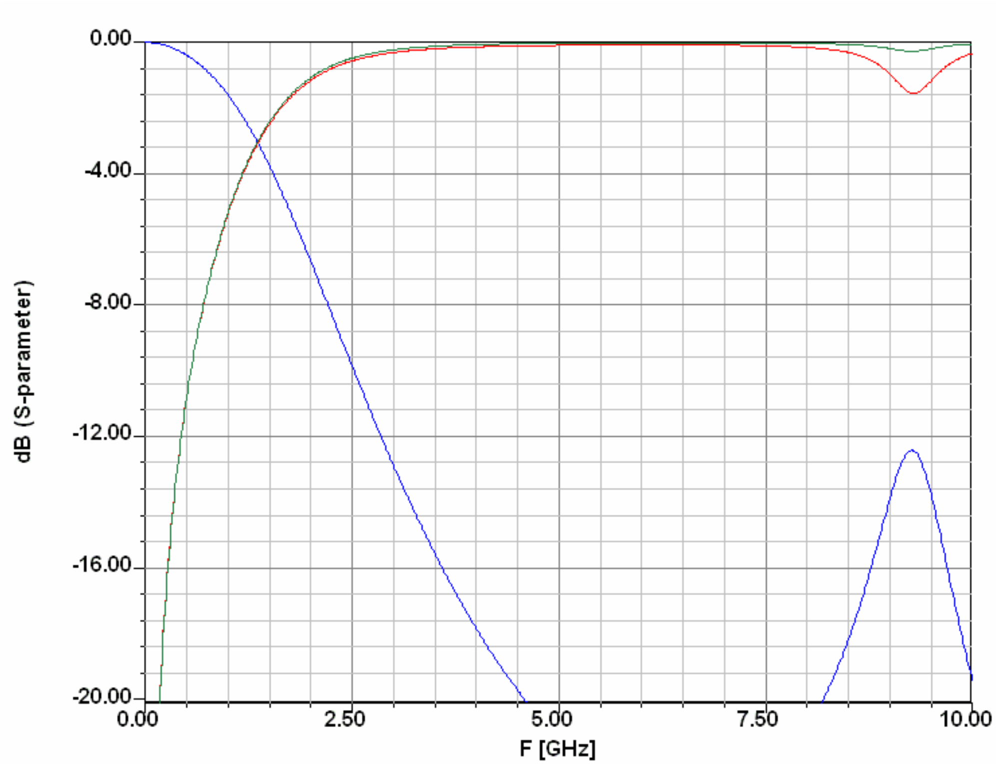
HFSS + Ansoft Designer
Entire filter with Lumped Port

- Model looks the same as LumpedRLC, but with shunt lumped ports instead

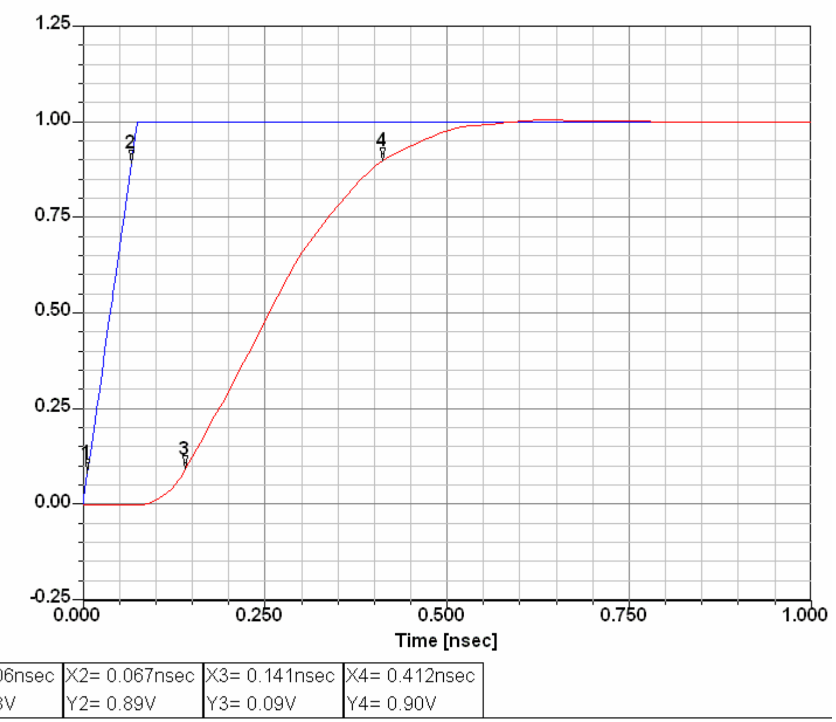




Entire filter with Lumped RLC Results simulated in Designer

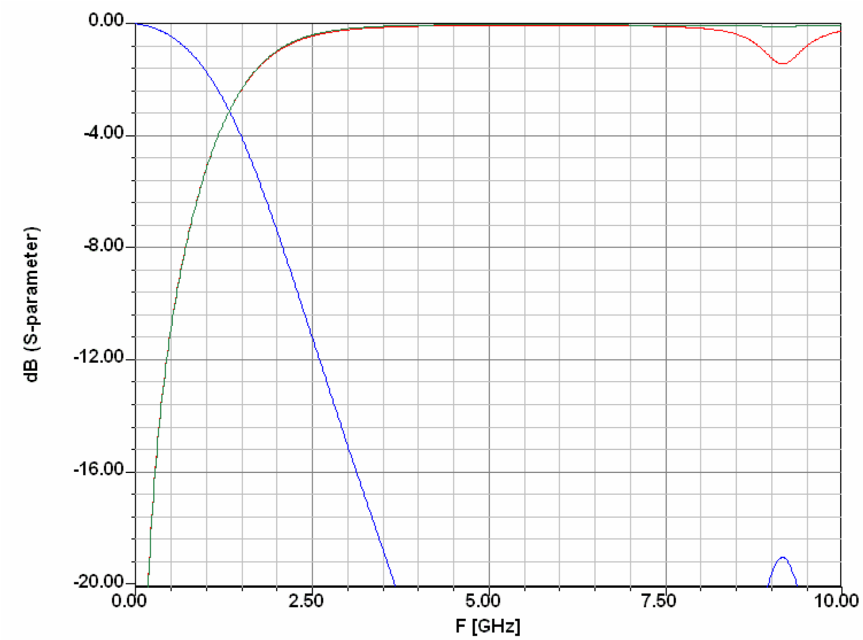


Output Rise Time = 270 ps



Entire filter with Lumped RLC Results simulated in Designer

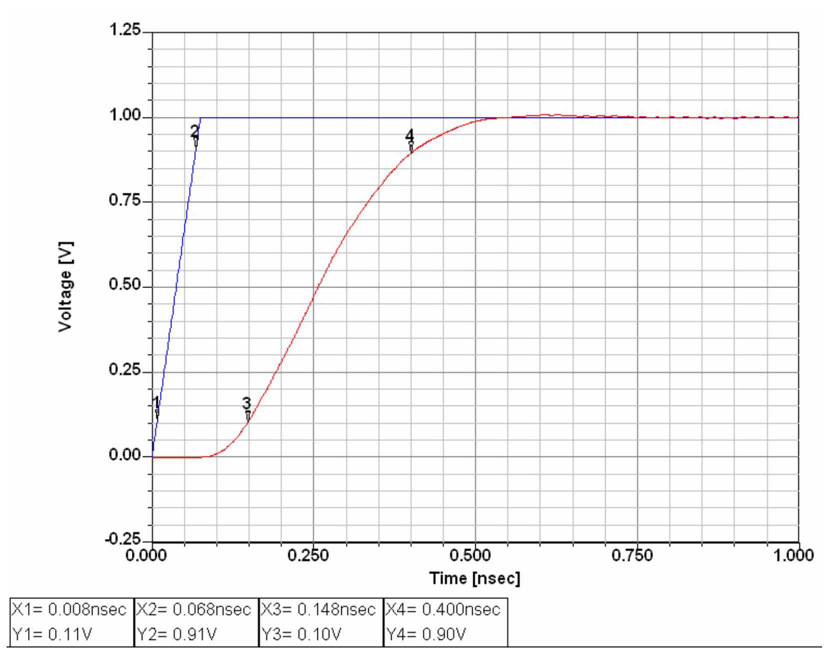
- Now let's try a real inductor
  - Coilcraft – 0402CS – 12 nH



Series resistance of real inductor dampened

Higher frequency resonance

Output Rise Time = 250 ps

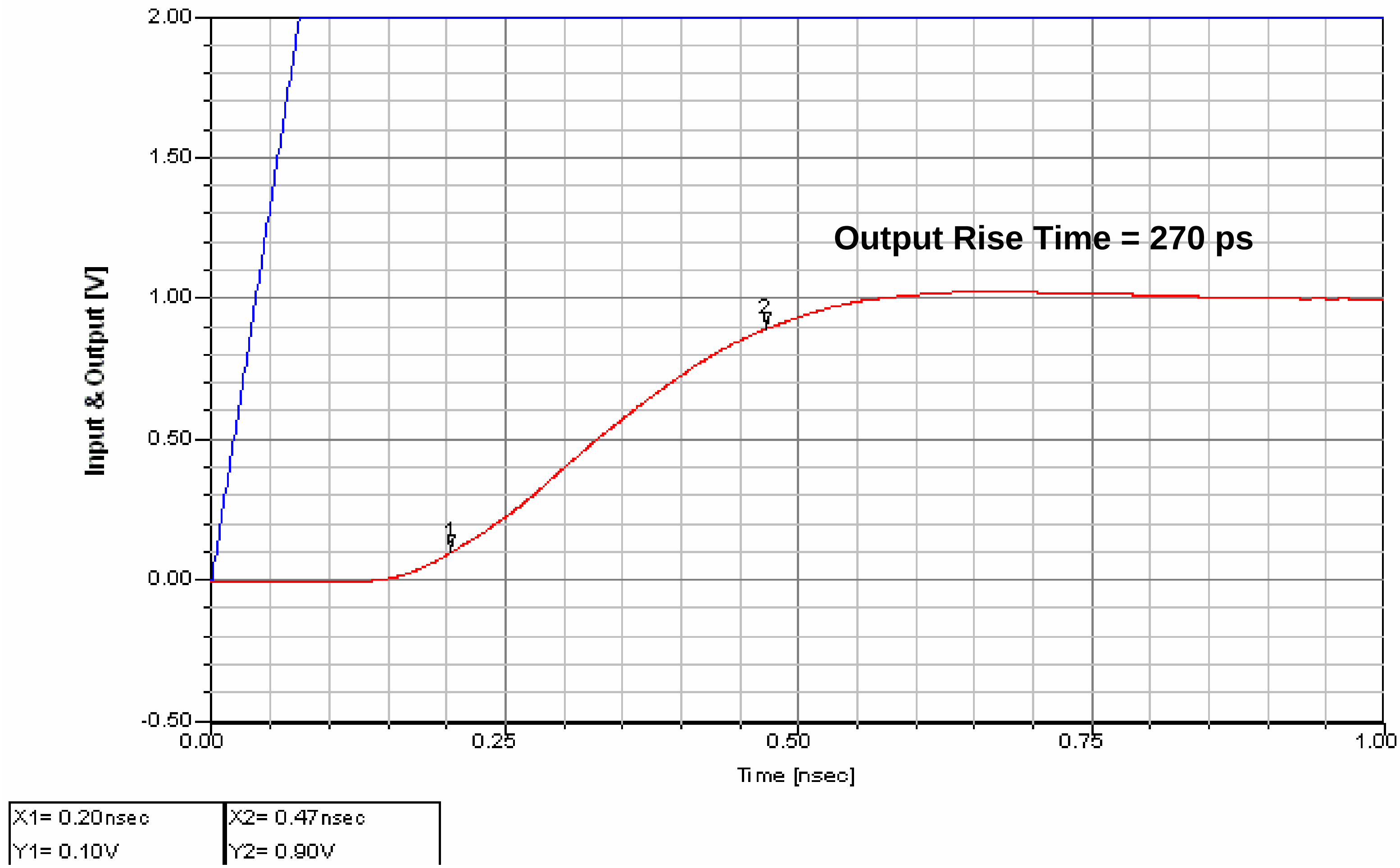


Entire filter in HFSS
Add dual via sections

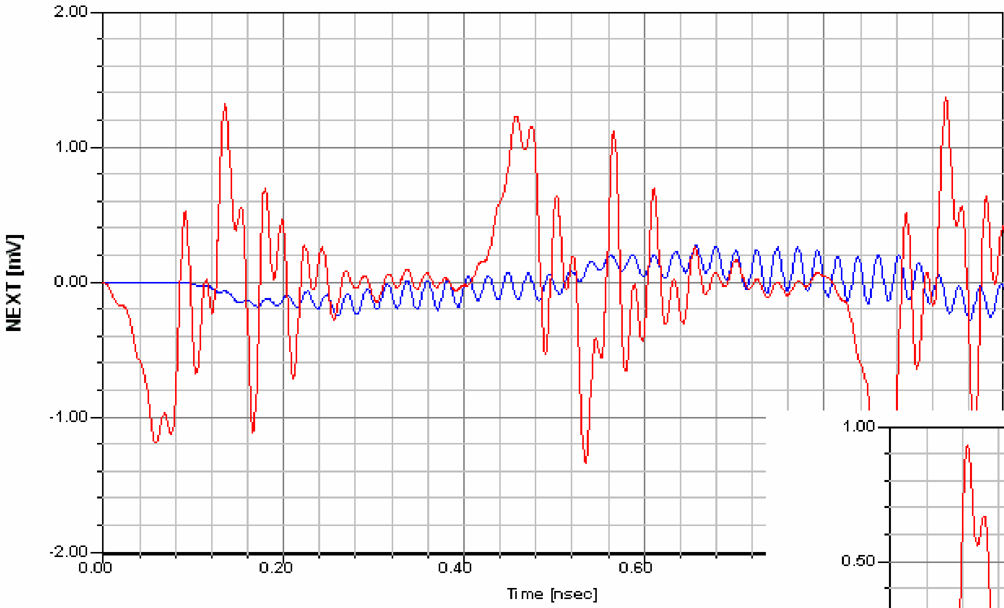
+1 V pulse

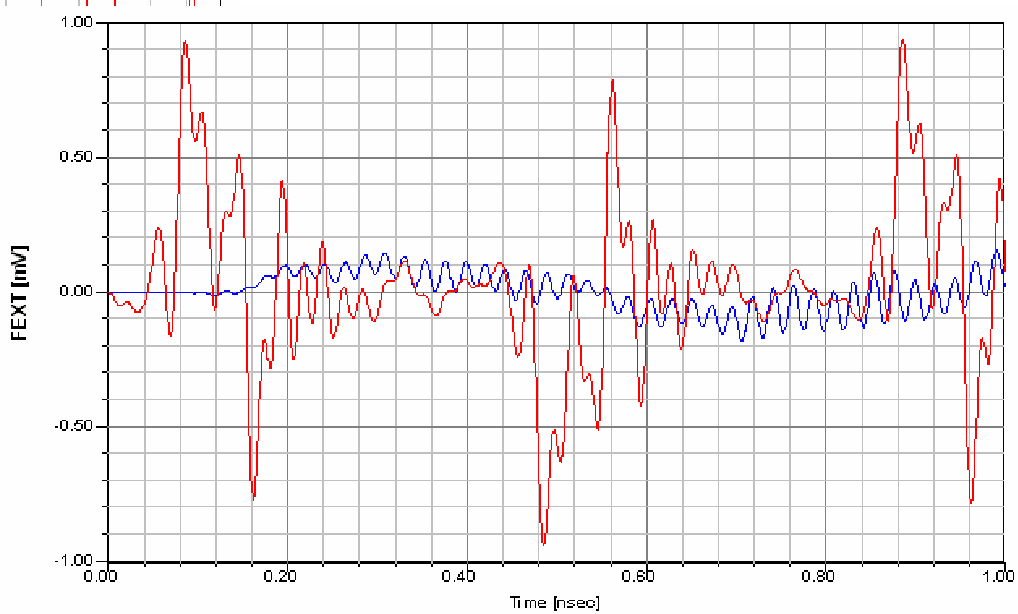
-1 V pulse

Entire filter in HFSS
Add dual via sections



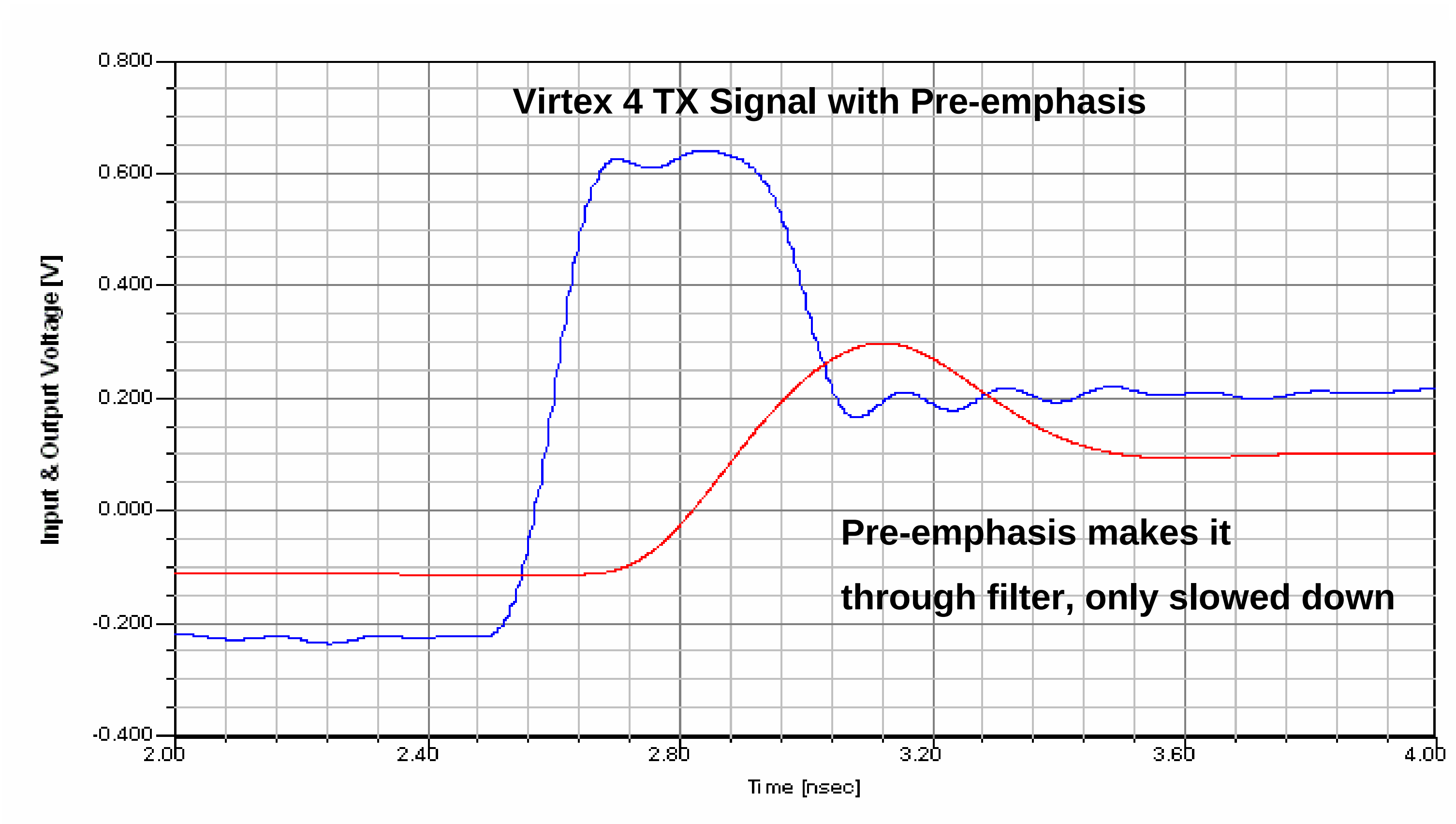
Entire filter in HFSS
Crosstalk Noise





- Blue trace – With filter
- Red trace – Without filter
- Both – 60 ps edge rate input

But what about ***real*** signals ?



Summary

- Illustrated that sometimes, faster is not always better.
- We can compensate for faster edge rates by adding a simple Bessel filter on the outputs of the transmitters.
- This filter can be printed and may require the use of external passive components.

Thank You

- The author would like to thank the contributions of:
  - Jory McKinley – Ansoft
  - Paul Back & Abe Hartman – Juniper networks
  - Paul Galloway & Mike Degerstrom - Xilinx