Using CAD for RF Architecture Engineering and Optimization

By Rulon VanDyke

The RF architecture phase is the foundation for the entire RF design. However, designers are still using a hodgepodge of software tools during this phase, which completely eliminates all of the design verification steps along the process. Many design iterations are the result of oversights in the architecture. Furthermore, these hodgepodge software tools cannot identify root causes of architecture problems. It seems so ironic that during this day and age when we have had so many advances in computer and software technology that the RF foundation work is still being done in spreadsheets, math packages, and custom tools.

During the course of this paper we will present a completely new way of thinking about RF simulation that starts at the foundation of the problem, which is RF architecture. We will then establish the need for RF architecture tools and show through an actual software example how we can use this tool to improve the design process by reducing unnecessary costs and design iterations allowing designers time to improve the quality of the product.

Design Process Overview

"A communication-system design from conception through production involves a sequence of phased steps. If you can't achieve satisfactory results in one step, you may need to go back to the previous one and restructure it as part of an iterative back-and-forth process, but you don't want to go back more than one step in the process" (courtesy *Wireless Design and Development,* published by Cahners Business Information).

"You can first simulate system architecture to efficiently allocate and distribute performance parameters among the various stages, balancing and trading off factors such as input signal strength, internal and external noise, bandwidth, distortion, and channel dynamics to achieve system goals at lowest cost. This simulation isn't necessarily a one-pass process, either. You may decide on what initially appears to be optimum balance among the stages, only to find that you can't meet the design goals for one stage. When this situation happens, the iterative process begins: You go back to your original plan, re-evaluate and reallocate performance goals for each stage or modify your algorithms, and then try to design a suitable circuit" (Bill Schweber, "Communication simulation software smoothes system design", EDN Magazine, August 3, 1998, pgs 87 – 108).

The typical "conception through production" design flow is as follows and illustrated in Figure 1.

1. Architecture Phase
2. Design / Purchase Phase
3. Implementation Phase
4. Integration Phase
5. Pre-production Phase
6. Production Phase

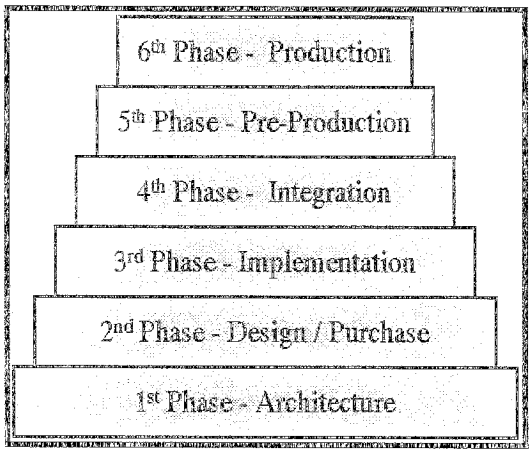

Figure 1 - RF Design Process

An ***ideal*** RF design cycle using state-of-the-art technology is as follows:

Architecture Phase

1. Propose a number of architectures
2. Perform a quick high level static analysis of each architecture
3. Perform an advantage / disadvantage trade-off study between proposed architectures
4. Select the desired architecture
5. Perform an in-depth dynamic analysis of the desired architecture
6. Partition block specifications
7. Re-evaluate and reallocate block specifications to meet performance and cost goals
8. Document the architecture and block specifications

Typical tools used to complete this phase:

- Spreadsheets (architecture & link budget)
- Math packages (architecture & link budget)
- Custom designed software (architecture & link budget)
- Word processor (documentation & block diagrams)
- DSP tools (link budget)
- Graphic packages (block diagrams)
- Schematic tools (block diagrams)

Major Obstacles with this Phase: Companies are not using current RF simulation tools for this phase! Apparently, linear, SPICE, and harmonic balance simulators are not being used for this design phase. Only very rough customer requirements can be verified during this stage using a hodgepodge of software tools.

Desired Output: *Architecture document that include a block diagram and specifications for each block*

Design / Purchase Phase

1. Determine blocks to be purchased or designed
2. For purchased blocks create specifications and procure blocks or components
3. For designed blocks synthesize the circuit as much as possible
4. Design linear circuits using linear simulation tools
5. Design non-linear circuits using non-linear simulation tools
6. Sensitivity analysis
7. Yield predictions
8. Temperature sensitivity
9. Collect simulation data from designed circuits
10. Collect measured data from purchased circuits / blocks
11. Substitute data back into top level architecture for static design verification
12. Simulate dynamic requirements such as digital modulation effects of BER, eye diagrams, and constellation plots
13. Repartition block / component specifications if necessary
14. Documentation of the design

Typical tools used to complete this phase:

- Spreadsheets (cascaded components)
- Math packages (cascaded components)
- Custom designed software (cascaded components)
- Linear circuit simulator (linear circuit design)
- Harmonic Balance or SPICE (non-linear circuit design)
- Test automation software (data collection)
- Schematic tools (schematic for designed blocks)
- Word processor (documentation)

Major Obstacles with this Phase: Most RF designs contain a mix of purchased and designed blocks / components. Software to collect data for purchased components is very limited and typically ends up being custom software. Once again verification of customer requirements is difficult at best because current RF simulation tools are geared toward RF circuit design and not RF architecture. This random approach does not facilitate thorough exploration and optimization of the chosen architecture, and often does not expose design or performance limitations lurking within.

Desired Output: *Design documentation and schematic ready for layout*

Implementation Phase

1. Prototype layout for designed circuits
2. Electromagnetic simulation of layout
3. Collect measured data from purchased circuits / blocks
4. Substitute EM data back into top level architecture for design verification
5. Repartition block / component specifications if necessary

Typical tools used to complete this phase:

- RF simulator layout tools (good for RF prototype work)
- Linear circuit simulator (linear circuit design)
- Harmonic Balance or SPICE (non-linear circuit design)
- Electromagnetic simulator
- Lab equipment
- Data collection software

Major Obstacles with this Phase: Once again customer requirement verification is difficult because simulation and measurement tools are not integrated with RF architecture tools.

Desired Output: *Commitment to a complete prototype unit*

Integration Phase

1. Integrate firmware, hardware, and software
2. Collect measured data from prototype unit
3. Verify static customer requirements such as frequency response, power levels, etc.
4. Verify customer dynamic requirements such as BER, eye diagrams, and constellation plots
5. Repartition block / component specifications ONLY IF NECESSARY

Typical tools used to complete this phase:

- Lab equipment
- Data collection software
- Other firmware and software tools

Major Obstacles with this Phase: Unfortunately, this is typically the stage for the current design process where architecture weaknesses can be identified. Identified architecture weaknesses are very costly at this stage an undetected architecture problems are even costlier.

Desired Output: *Commitment to pre-production layout and models*

Pre-production Phase

1. Collect measured data for entire system or RF architecture
2. Verify all static and dynamic requirements have been met through measurements

3. Repartition block / component specifications ONLY IF ABSOLUTELY NECESSARY
4. Yield optimization and tuning

Typical tools used to complete this phase:

- Lab equipment
- Data collection software

Major Obstacles with this Phase: Time to market and manpower resources are typically the limiting factor in collecting complete static and dynamic design performance information.

Desired Output: *Design ready for manufacture*

NOTE: In practice many of the preceding design steps are skipped because of time-to-market and resource constraints. Obviously, the solution to this problem is to develop user-friendly tools that can be used to integrate all software and data collection tools used from 'conception to production'.

Identifying Problems Early Saves Time and Money

Customers are increasing pressure on companies to reduce the time to market and to drive down the costs while improving the quality and reliability of their products. RF designers need tools that are easier to use, require a shorter learning curve, and are more integrated, thus allowing designers the ability to do top-down design. This top-down design must provide constant feedback to the designer of requirement compliance in order to eliminate costly design turns near the end of the development cycle.

Early identification of potential issues alerts designers to weaknesses in the design and architecture before they become a problem. If these weaknesses can be examined and addressed early on during the architecture phase, the cost of design changes will be at a minimum. Typically, the longer an architecture problem lurks the costlier it is to fix. Resolving these issues early is paramount to helping designers improve time to market while lowering costs and improving the quality. See Figure 2.

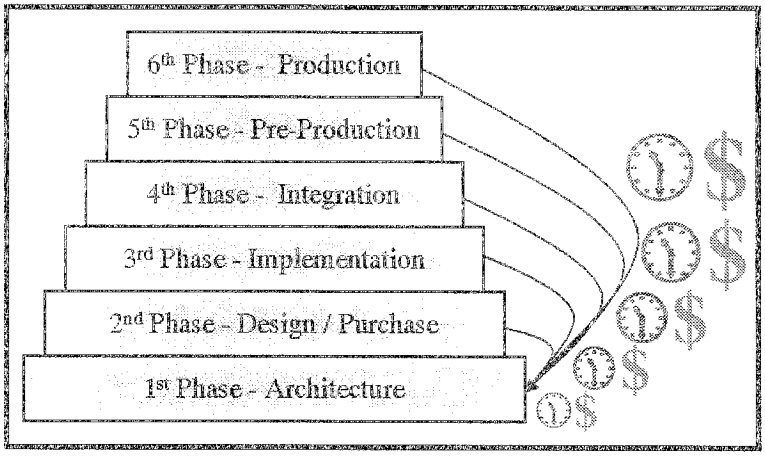

Figure 2 - Cost of Rework vs Phase of Identified Problem

Design Verification Good in Theory But Is It Practical?

One of the biggest challenges in the RF CAD arena is the ability to verify that customer requirements are met during every step of the design process. Time to market pressures compound the problem and short cuts are usually taken eliminating important design verifications along the way. Many times the first design verification step may actually be when the first piece

of hardware shows up in the lab. This is a risky and costly approach. Doesn't it seem reasonable in today's advances of software and computer technology that design verification should be done quickly in software before committing to hardware? Why is this a big challenge?

To understand why design verification is currently so difficult let's examine the current types of software tools used during the RF design process. During the architecture phase RF designers have been accustomed to creating their own spreadsheets, custom software tools, or using math packages. This is a quick and dirty way of getting the job started but these tools are very lacking when it comes to supporting the design through the entire process. Design and purchase is the next development phase. Currently, CAD tools have focused on this particular design step. Techniques such as linear S-parameter type simulation, harmonic balance, SPICE, and electromagnetic simulation can be used during this phase. During the next phase, which is the integration phase, we begin to have physical realizations so the useful tools are once again electromagnetic simulation and measured data.

Ideally as the design proceeds from "conception through production" RF designers would like to verify customer requirements all along the way. However, looking at the current software tools on the market we see that the best we can do is circuit verification because the data from simulation tools such as a linear simulator or harmonic balance doesn't flow back into the tools used for RF architecture. Once the initial architecture design work is done, the current architecture tools provide little value during the rest of the design process. Using this approach, design verification is nice in theory but not very practical. See Figure 3.

In order to identify problems early, a good RF architecture tool needs to be developed and integrated fully with linear, non-linear, electromagnetic, and data collection tools.

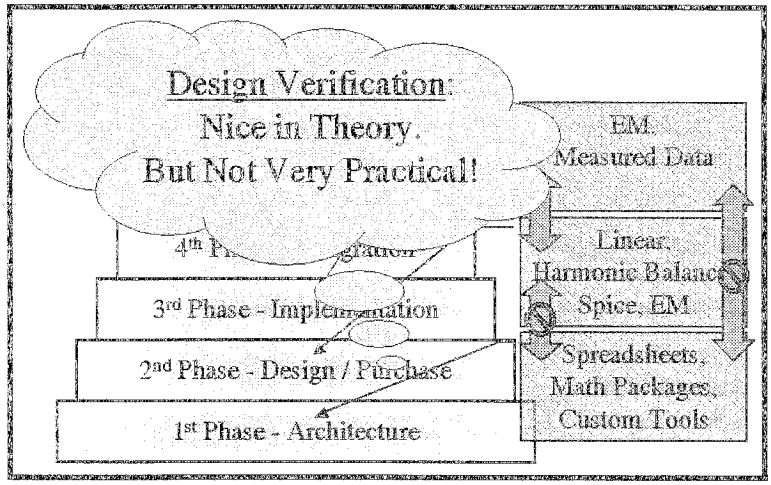

Figure 3 - Design Verification with Current Software Tools

Static and Dynamic Design

There are two categories of RF design, ***static*** and ***dynamic***. Static design is where the designer looks at design from static performance conditions (such as frequency response, gain, and power levels etc.) where all design parameters are budgeted to meet the performance criteria, minimize cost, improve the quality, and reduce the time to market. This is the first step where the designers declare that customer objectives are completely met in the static case. Dynamic design is where the designer examines the design that has already been completed under dynamic conditions such as time domain. Examples of dynamic design are looking at the design performance under modulation conditions such as BER, eye diagrams, and constellation plots. Examining the current RF simulation tools on today's market gives the perception that dynamic

analysis is all that is required since only these types of tools exist. Even though dynamic design is very important, static design has been greatly overlooked by the RF design community.

Can you imagine trying to dynamically balance the tire of your car when you have spent little time or effort doing a static balance first? Wasted time it the result of this balancing technique! This same principle applies to RF design. Investing huge amounts of time doing dynamic time domain simulations on a poor RF architecture is nothing more than a waste of time and money. As an RF design community we need to better utilize our time and money by establishing a solid RF architecture foundation and then when it comes time to do the dynamic design this process will be much easier and produce more fruitful results.

RF Architecture Solution

Eagleware has attacked this problem head-on and has developed a new simulation engine needed for RF architecture work that provides the platform to integrate synthesis tools, linear circuit simulation, non-linear circuit simulation, electromagnetic simulation, and measurement data collection. RF design from 'conception through production' is now possible in a single software tool.

New Simulation Engine

The simulation engine that Eagleware has developed is very unique. The simulation technique maps very closely into the physical world. Basically, the way it works is that every source plus intermods, harmonics, and noise propagates to every node in the system. Unlike a harmonic balance technique, all signals have bandwidth and spectral density. Each node contains spectrums from all signal sources and the products that they have created all along the way, traveling in all directions through the node. This is a continuous frequency simulator just as signals and noise appear in nature. The user has to tell the simulator which frequencies to ignore otherwise all frequencies in the entire spectrum would be processed.

Users can examine full node spectrums at any node or view channelized measurements along a user specified path. All channelized measurements integrate the full node spectrums at every node along the path. The user can define arbitrary paths and many RF types of measurements can be examined along these paths. All spectrums along the path are categorized before integration, allowing measurements the ability to integrate only specific types of spectrums. For example, some intermod measurements only act on intermod spectrums regardless of other signals that may be present.

Level diagrams and tables can be used to determine the RF performance along user-defined paths. Furthermore, each spectrum is uniquely identified with respect to how it was created and the path that it took to arrive at the viewing destination. These insights give the users the power to identify architecture weaknesses and problems long before they are built into the product. In other words, this is like having a super spectrum analyzer in the software that can look at signals, intermods, or noise only, show phase and amplitude of individual frequencies, identify direction of signal flow, and completely characterize who created the signal and the path it took to get there.

Documentation has been made much easier by using operating systems such as Microsoft Windows where schematics, tables, and graphs can be copied and pasted into documents and spreadsheets. However, this is not sufficient if schematics become cluttered with measurement icons and text. Typically, the designers don't want this detailed simulation-only information in high-level documentation. Once again Eagleware has tried hard to keep the schematics clean so they are of printable quality.

RF Architecture Design Example

This is a simple example of a 3-sector 5.8 GHz receiver that can be used as a TX power meter or VSWR tester. Three coupled antennas have been connected through a virtual node that represents the antenna-to-antenna isolation. Consequently, multiple carriers can be driven into each antenna where the 5.8 GHz receiver will be able to see all transmitted carriers through all

paths. Equations have been written to determine the complex impedance of the antenna based on VSWR and the reflection coefficient angle. This impedance has been substituted into all antennas. A six-position switch is used to select the antenna and the direction of power flow to be examined by the receiver. Two IF outputs have been used for this design. The 1st IF is at 450 MHz and has no AGC. On the other hand, the 2nd IF is at 70 MHz and has an AGC amplifier. Perhaps the 2nd IF could be used for demodulation of some other receiver functions and the 1st IF output could be used to drive a power detector. Cost is an issue for this design and so a resistive splitter has been used as well as several low pass filters. See Figure 4.

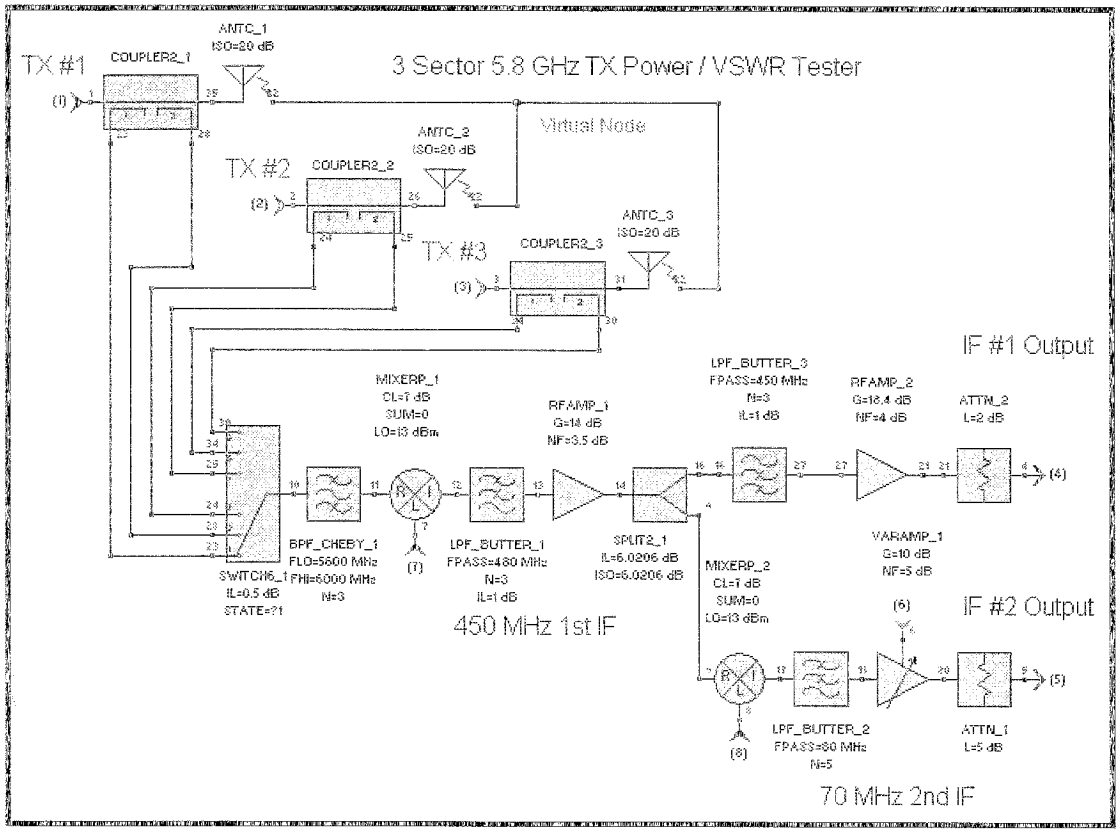

Figure 4 - Schematic of 3 Sector 5.8 GHz TX Power / VSWR Tester

At the maximum transmit power which is +30 dBm, we can plot the dynamic range of each stage along the path for IF #1 output. From Figure 5 we can quickly see that the amplifier between nodes 27 and 21 is in compression. As a matter of fact an orange colored schematic symbol means that the component has a warning. This will quickly alert users to potential issues. We could double click on the amplifier symbol right in the level diagram and change its requirements or we can change the parameters of any of the other stages if needed.

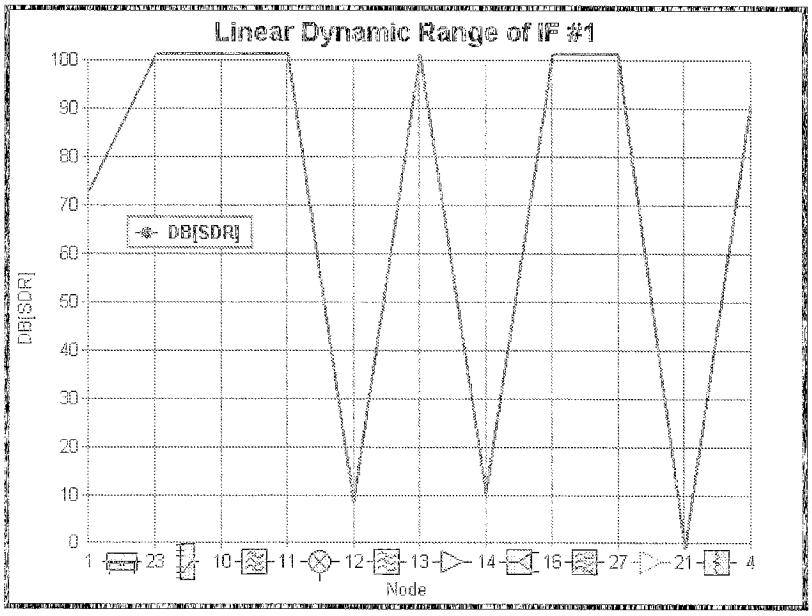

Figure 5 - Headroom at Maximum Power for IF Output #1

We can also look at the spectrum at the minimum transmit power, which is +10 dBm. From Figure 6 we can see that we definitely have some spectrum problems. Our desired modulated signal at 450 MHz is not the most powerful signal in the spectrum. We would now want to ask ourselves if the prior amplifier compression problem was only due to our desired signal or from other sources. Looking at Figure 7 we can compare the channel power to the total node power along the path. We can definitely see we have a problem with total node power since it is much larger than the channel power at many of the nodes. In this case the better architecture solution would be to get rid of the extra spectral junk than just raise the compression point of the amplifier.

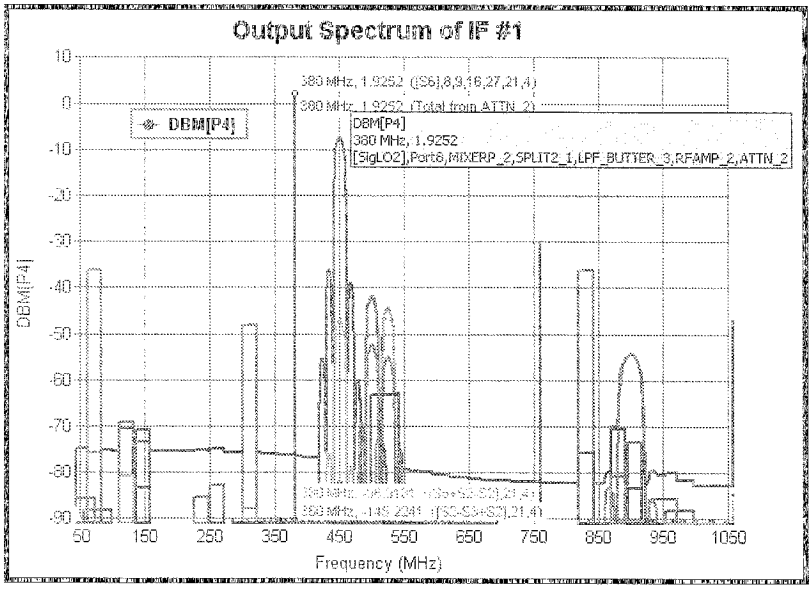

Figure 6 - Spectrum Output of IF #1 at Minimum TX Power

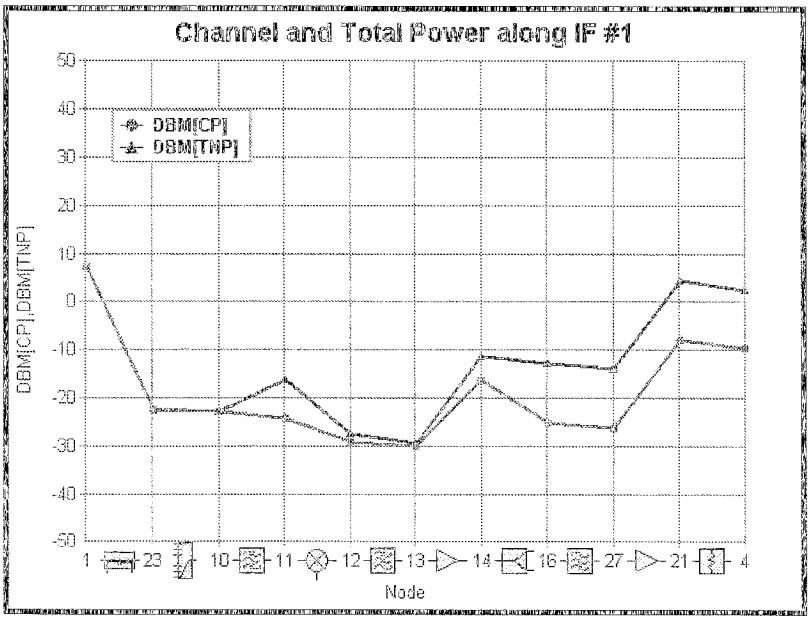

Figure 7 - Level Diagram of Channel and Total Node Power along Path

As can be seen we need to make an architecture change to get our TX receiver to work correctly. By placing the mouse on this strongest signal we can see that the worst offender is 'SigLO2' which is the 2nd LO signal at 380 MHz. Furthermore, we can also identify the path that this offender took to arrive at IF #1 output. This information shows us that the signal came through the LO leakage of the 2nd mixer then went through the passive splitter, filter, amplifier, attenuator, and to the output. Since we now know the exact problem we can select the best architecture changes to fix this problem. In this case one of these three ways could be used:

1) Better LO to RF isolation on the 2nd mixer
2) Better isolation in the splitter (i.e. use of a Wilkinson instead of a resistive)
3) Change low pass filters to a bandpass filters

Through the RF architecture software the user can evaluate the performance tradeoffs of all the potential solutions. After replacing the last low pass filter in the 1st IF with a band pass filter we achieve the following performance as shown in Figure 8. Notice the remarkable improvement in the spectrum. The hodgepodge of current simulation tools could not give the user this type of insight into these architecture problems. Having complete identification of the problems available to the designer gives them the power to provide the best architectural solution.

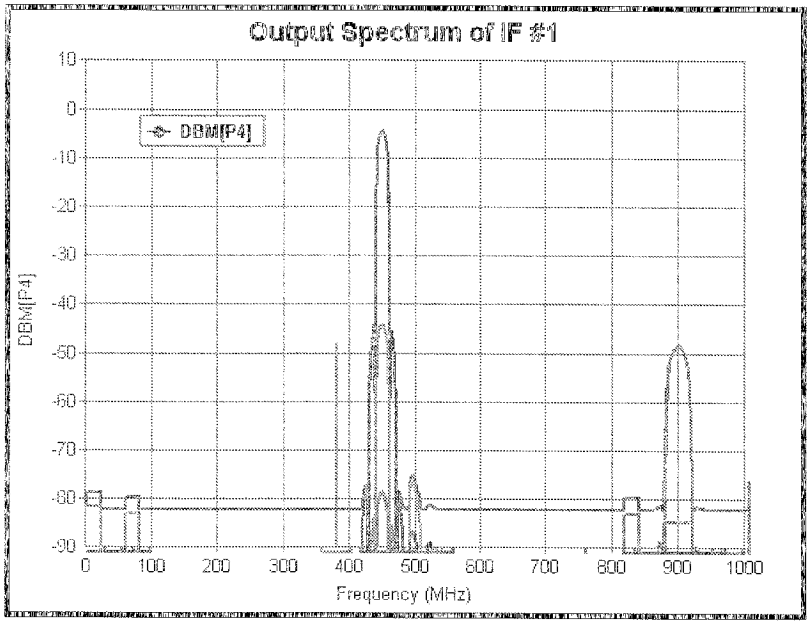

Figure 8 - Output Spectrum of IF #1 after replacing a Low Pass Filter with a Band Pass

We can also look at the conducted emissions at the antenna as shown in Figure 9. Here we can also identify issues and determine where the root problems are so we can address architecture issues early on in the design process.

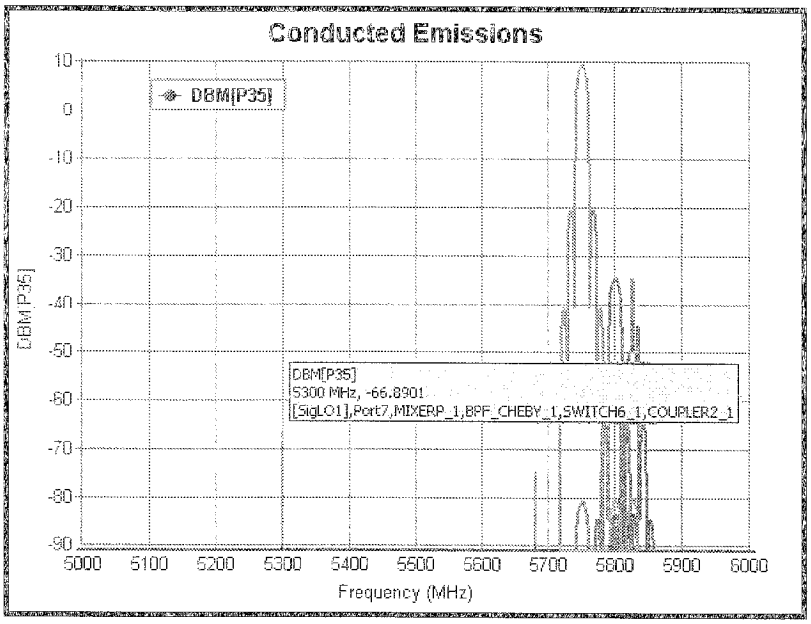

Figure 9 - Conducted Emissions at the Antenna #1

Summary

Due to the limitations in printed space we have only identified a few of the features that the current RF architecture tool provides. A feature that allows the user to integrate any portion of the spectrum over any bandwidth is available as well as optimization of any RF measurement at any node along any path. Included in this tool are also yield, sensitivity, and monte carlo analyses.

Through this brief example it can be seen that the early identification of many problems can be caught and the architecture optimized before committing to any type of hardware or requirement specifications. Once specifications have been sent to hardware vendors and hardware is being built any design changes cause time-to-market to slip and cost increases. Having an integrated RF architecture tool is the key that allows designers the ability to closely look at the performance of the entire RF chain for both static and dynamic cases before committing to any type of hardware.

Author Information

Rulon VanDyke is the lead engineer in systems simulation at Eagleware Corporation. He received both a Bachelor of Science degree and Master of Science degree in electrical engineering from Brigham Young University in 1990. For 10 years, he designed first-, second-, and third-generation digital cellular transceivers and base stations for AT&T Bell Labs and Lucent Technologies. In 2001, he joined Eagleware to develop SPECTRASYS. He may be reached via E-mail: rulon@eagleware.com

For more information, contact:

Eagleware Corporation
635 Pinnacle Court
Norcross, GA 30071
Tel: 678-291-0995
Fax: 678-291-0971
Email: sales@eagleware.com
Internet: www.eagleware.com