SEQUENCE

The SoC Design Closure Company℠

Designing Noise-free
in Deep-Submicron

March 2001

Confidential Material

Table of Contents

SEQUENCE
The SoC Design Closure Company℠

Sequence Design, Inc
469 El Camino Real, Suite 202
Santa Clara, CA 95050

Why Noise?

Noise is a particularly pernicious problem at nanometer device sizes under seven plus layers of metal. Post-layout verification (extraction followed by some analysis) continues to be the main approach to identifying and correcting noise problems before tape-out. However, full-chip verification is computationally expensive and identifies thousands of ECO's that are expensive and time consuming to correct. Why so many problem, because design for noise is an afterthought. Noise immunization has not been integrated into commercial cell-based design flows. Sequence Design now provides the first noise immunization methodology and toolset for cell-based designs.

Crowded wiring from reduced device feature sizes and coupling capacitance from increased aspect ratios impacts the timing and signal integrity in designs today. As designs move to .18µm and below potential noise sources become even more important. Directed design changes can anticipate and eliminate these problems up ahead. Such methods require physical design information, electrical analysis and directed circuit optimizations. Introduced at the right time in the flow design changes immunize the design from data dependent noise (cross-talk induced delay).

Advances in process technology with addition of copper (Cu) and low-k dielectrics will mitigate some of interconnect noise problems, but they will not remove them completely. As a result lower yields and functional failures from noise will be a bottleneck to complete successful design. Further, System-on-Chip (SOC) integration requirements will stress the usefulness of existing design methods, which don’t account for noise. Designing noise-free will be mission critical in SOC.

Sequence Design 's new approach involving avoidance, detection and correction is the best approach to eliminate noise up-front as part of a fast, automated design process. The Sequence solution works in both a pre-route and post-route mode giving the flexibility to correct noise problems both in the placement stage and in the post routing stage of the design flow.

Sources of Noise in Ultra Deep Submicron Today

Noise takes many forms in UDSM design. Noise occurs directly from special sources like clock and power grid. Clocking injects noise at specific cycle times, which then propagates through the design. Power and ground grids, because of their resistive nature, cause voltage fluctuations at the supplies and ground of the gates, which affect signal propagation. Wire density and increased wire aspect ratios result in wire coupling interactions that impact delay and functional operation. Packaging choices, resulting in bond wire inductance, introduce switching noise in high-speed applications. Substrate noise injected from digital drivers can affect the sensitive analog portions of the design. Collectively, these noise issues need to be anticipated and minimized to produce the best performance-yield against design specifications. Loss of product yield, and delays in product release are the worst result of poor management of noise issues.

Power supply, ground grids and clocking are usually pre-planned as part of design formulation. A key design problem is then accessing the trade-off of such pre-planning as well as mitigating the impact of these pre-route decisions on the rest of the design. Since these planned portions of the design inject noise into the design analysis are necessary to determine where and whether this noise affects design yield. This is best done during the design formulation phase as part of the place and route process.

One of the most difficult problems to handle in today's flows is the noise introduced from wire coupling. This noise manifests itself by impacting the delay along critical paths of the design. Not meeting the timing constraints of these paths either introduces the need for more design iterations or the hard choice of loosening product performance specifications. This type of delay is dependent on the nature and direction of the signal in the wire. Because this coupling both helps and hinders the timing it introduces ambiguity into the design process. Handling the data dependent delay ambiguity in design then is an emerging mission critical part of the design flow. Another difficult issue resulting from coupling is glitch. The peak noise injection from glitch can cause functional circuit failure if it propagates to a latch. Glitch or charge sharing noise needs to be calibrated for a given process. In a standard-cell flow the understanding the propagation characteristics of this type of noise through a cell are necessary to determining whether the noise is serious enough to cause a circuit failure.

Focus on Cross-talk

Today's flows which rely on simple assumptions like grounded 2C coupling on all nets, leads to overestimation of coupling on small nets by as much as 83%. Worst still the 2C assumption has been shown not to be an upper bound on coupling which leads to underestimation on some critical nets. It’s difficult for logic synthesis to account for coupling variability as part of the compilation stage even with an integrated placement. Directed physical optimizations based on an electrical view of the design can anticipate and eliminate data noise problems as part of the physical design process. It is precisely at this point in the flow that all the available information is available to make the right choices to eliminate this source of noise.

Timing and Noise

Timing convergence is considered the major problem in design flows today. Convergence occurs when the pre-physical timing falls within a window of post-route and extracted timing, or when all paths meet the timing constraints. Static timers are used as the measure of validated timing.

Static timers look at timing using an expression made up of three elements, gate delay, gate loading delay and wire delay in form:

Delay = $G_i+G_L+RC_w$

Gate delay (G_i), often called intrinsic delay, is the time it takes the signal to traverse from input to output of the gate. This will depend on the supply voltage, the input slew, the temperature, and the process mix. This delay is relatively small now compared to the other elements of the static delay expression. The second portion of the static timing view is the Gate loading (G_L). This is the delay added to the intrinsic delay the account for the fanout loading and the loading that each gate drives. The third portion of static view of delay is the time-of-flight delay of the wire (RC_w). This is the delay due to the resistive and capacitive nature of the wire. The problem with timing convergence is that the physical partitions, placement and optimizations of today's P&R tools do not understand the network nature of the wires and how they are affected by each change in placement. The best design will match cell drive strength to the load for each placement choice, accounting for the capacitive or resistive characteristics of the wire.

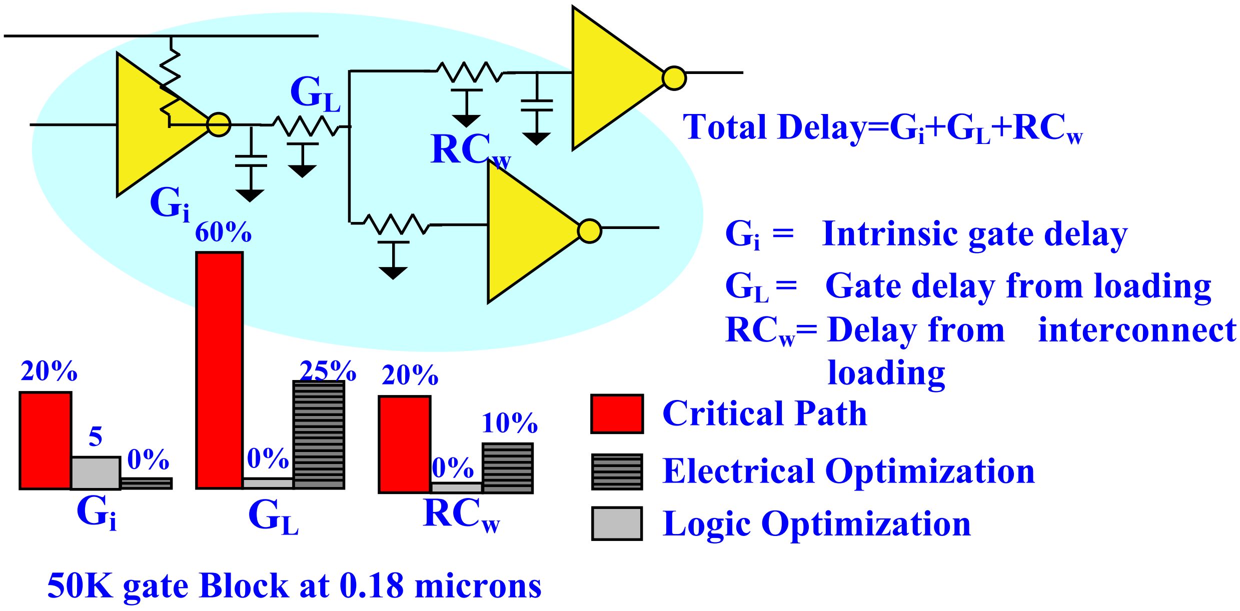


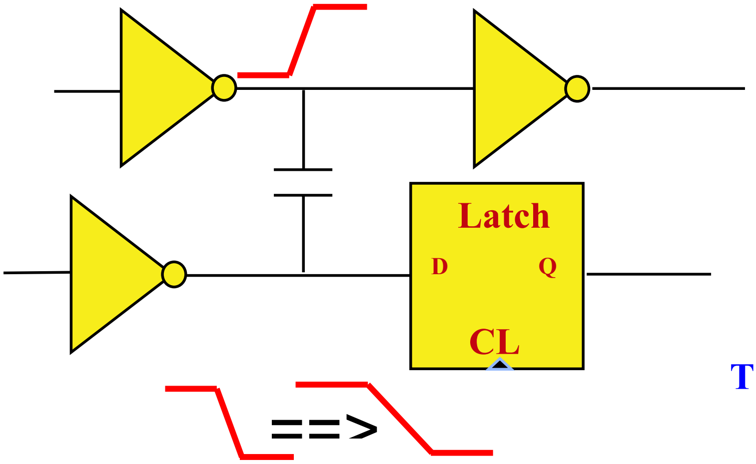
Figure 1

Figure 1 illustrates graphically the importance of understanding the nature of delay as designs move below 180 nanometers. The first histogram shows the total overall contribution to delay from each element of the static timing (G_i,G_L,RC_w) on a fifty thousand gate block. Gate loading and wire delay (both strong functions of interconnect capacitance) contribute nearly 80% of the total path delay. Intrinsic gate delay contributes to a shrinking percentage of the overall critical path delays (20%). Focusing on eliminating a gate delay or two along the critical path only contributes to gains in the range of 5% to 7% depending on the logic depth (histogram 2). However, adjusting the placement based on an electrical view of the design contributes up to 35% improvement in overall timing (histogram 3). The simple conclusion, spend more time optimizing electrically for the greatest return in performance. This is the Sequence approach.

As designs become more wire congested the coupling capacitance becomes more significant contributor to the total capacitance along the path. This coupling capacitance impacts the delay equation by modifying both the load and the wire delay. However, the impact of coupling is based on signal relationships that change based on changes in the data driving the circuit. Thus the interaction depends not only on physical capacitance between wires but also on the Miller relationship. The Miller relationship can either increase the relative impact of capacitance or in some cases decrease it. Regardless, this ambiguity left undetected will result in performance-yield problems or excessive iterations to assure proper timing.

Cross-talk Induced Delay

Figure 2 illustrates the relationship that Miller capacitance (m) has on the deep submicron static timing equation. The effect appears both in the gate-loading portion of total delay and wire delay. Accounting for the variation the Miller effect has on delay, and set-up & hold time, is essential for correct timing and higher yielding designs.



$$\text{Total Delay}=G_i+G_L+RC_{int}$$

$$G_L = f(C_L) = f(C_{gate}, C_{int})$$
$$C_{int} = f(C_{gnd}, mC_c)$$

Figure 2

Comparison of Critical Nets

The data dependent noise that results from coupling slows and speeds signals along the paths of synchronous designs. Typical timing-driven methods look to constrain or fix path delays as the design moves from the logical stage to the physical stage. The critical paths are identified using static timing analysis. As the design reaches physical layout, timing along the identified critical paths is used to constrain wire minimization approaches. After placement & routing the typical design flow takes extracted capacitance from physical layout and doubles it, setting the Miller effect at 2. This 2C view is then grounded and used in static re-timing to account for coupling noise.

The problem with this existing approach is that the simple 2C lumped and grounded view of coupling along the critical path is no longer valid. Assuming that all nets are coupled in this manner leads to both pessimistic and optimistic estimates of timing. Table 1 illustrates the potential magnitude of the error as a function of wire length. As shown, relatively short wires, up to 500 microns, show significant variation. Based on this variability, the question then becomes, which wires are critical from a timing perspective. Identifying the wrong nets because of a pessimistic estimate or ignoring nets because of an optimistic estimate both results in an unacceptable risk at tape-out.

Wire Length	**Over Pessimistic** $2C_c$ (to ground) to lower bound delay along path	**Over Optimistic** $2C_c$ (to ground) to worst delay along path
125 µ	7%	4%
250 µ	26%	11%
500 µ	83%	22%
1000 µ	217%	14%
2000 µ	463%	9%
4000 µ	815%	7%

Table 1 maximum percentage error using 2C approximation[1]

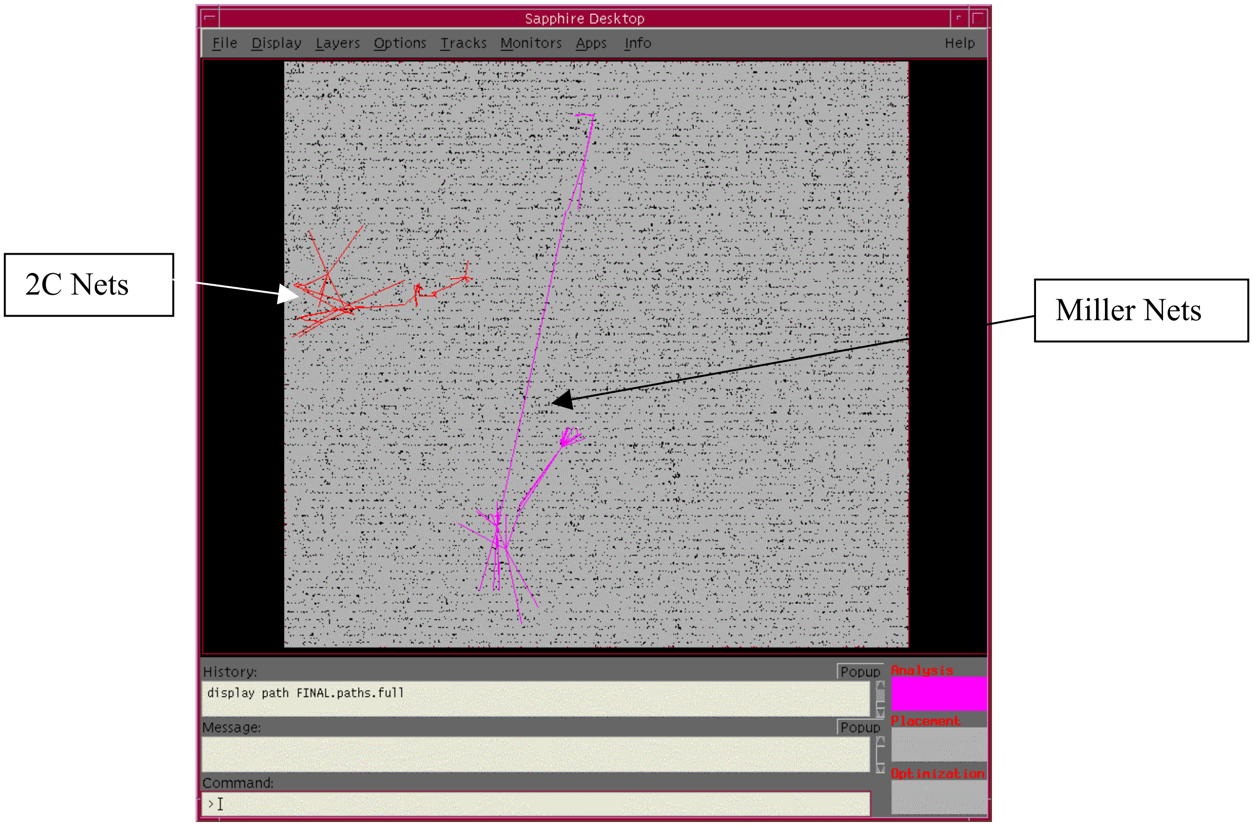
Timing inaccuracies due to simple 2C approximations propagate into timing-driven placement algorithms and physical optimization algorithms that rely on net timing numbers. Excessive weights placed on the wrong nets in quadratic placement will cause timing iterations. Isolating on the wrong nets for physical optimizations will also cause timing iterations.

If the 2C approximation is not adequate then what technique is appropriate to apply? The actual coupling effect, adding to and degrading the signal, is related to the actual net-to-net interaction (Miller effect). Calculating a Miller view of the coupling then will provide a range of capacitance's, from 0C to 3C, depending on what is actually

---

[1] Wire Delay in the Presence of Crosstalk, Yee, Cahndra, Gaseasan & Sechen p174

happening along the wire. With this new view it's possible to reassess the timing along the paths to determine which ones are really critical. Figure 2 illustrates the results of such an operation on one particular design. As shown the path identified using the 2C approximation and that identified when the Miller view is used are different. Exploring further, the 2C path is gate delay dominated, whereas the Miller path is fewer gates but spaced further apart. The longer routes increase the coupling interaction.



```
display path FINAL.paths.full
```

```
>I
```

Figure 2
Different Critical Nets Identified due to Data Dependent Noise

The validity of using the Miller approach instead of the 2C approach has been confirmed on a number of circuits. Consistently, the post route timing critical nets match the nets identified with the Miller view. Sequence is the first company to apply this Miller approach successfully in the standard-cell design flow. But even with this better electrical view of the design the number of potential victim-aggressor nets that need to be tested becomes prohibitive large. The next step then is to isolate the nets that are susceptible to noise.

Glitch Noise

Glitches also arise due to coupling effects between nets. They can lead to functional failure of the circuit if they propagate to a latch. Sequence's glitch solution is used in pre-layout mode for noise avoidance and in a post-layout mode for noise correction. Glitch is isolated and corrected using the following methodology.

First, physical and electrical pruning narrows the search space for glitch analysis. This focuses the analysis on a subset of nets. The sensitivity of a net to glitches is computed based on the electrical parameters of driver, net and load. Nets with low sensitivity are ignored in further analyses. This is called victim pruning. Nets that are physically apart cannot inject noise into the victim net and are ignored for further analyses. This is called aggressor pruning.

Next, glitches get injected into the victim node when the aggressors are switching in the opposite direction to the polarity of the victim node. Worst-case glitches arise when the aggressor and victim are switching simultaneously in the opposite direction. Glitches due to all the aggressors are computed at all the victim nodes. The resultant glitch at the victim node is computed by integrating all of these glitches along with the propagated glitch (figure 3). Glitch injection depends upon the drive resistances of the victim and aggressor drivers, interconnect resistances, interconnect capacitances and coupling capacitance.

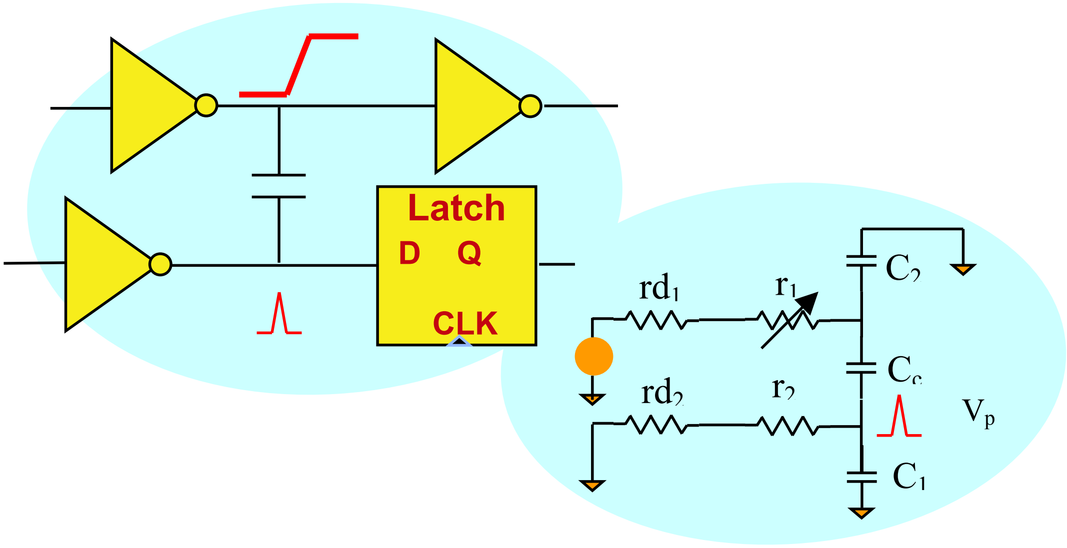


Figure 3

Sequence's comprehensive electrical event driven timing analysis provides all the timing windows required for the analysis.

In the pre-layout mode, the total coupling capacitance of the victim net is distributed in the aggressor nets.

Then the glitches above a glitch threshold get propagated thru a cell. Glitch threshold values are set at a global level or cell-by-cell after a comprehensive characterization. The magnitude of the output glitch depends upon the magnitude of the input glitch.

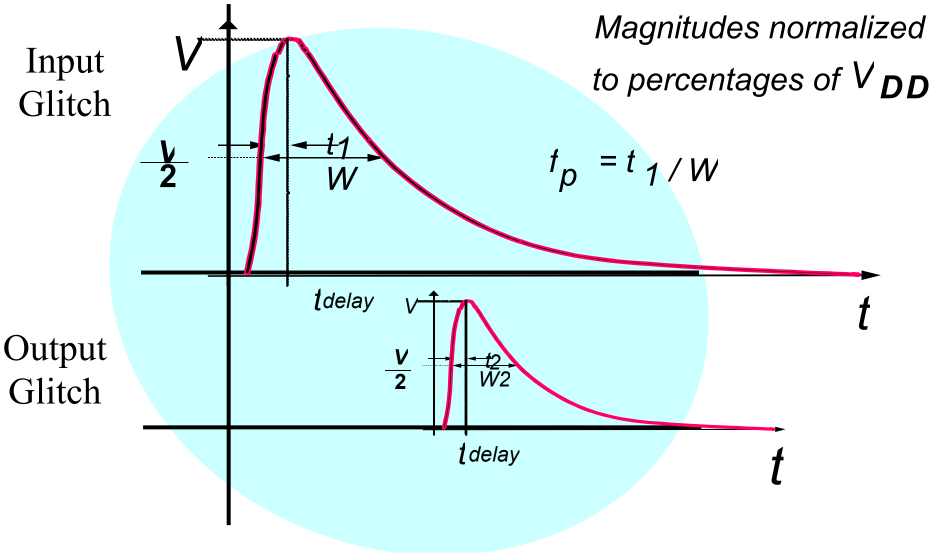


Figure 4

Once models and isolated the following optimization techniques are used for glitch noise avoidance (pre-layout mode) and correction (post-layout mode):

- Buffer, Inverter Insertion: Reduces the magnitude of glitch by reducing resistance and capacitance. Inverters help in annulling glitches.
- Gate Sizing: Victims can be up-sized and (or) aggressors can be down-sized.
- Placement: R,C can be reduced by changing placement.
- Routing Directives: Wire width, spacing and shielding directives can be provided to the routers.

Isolating Noise

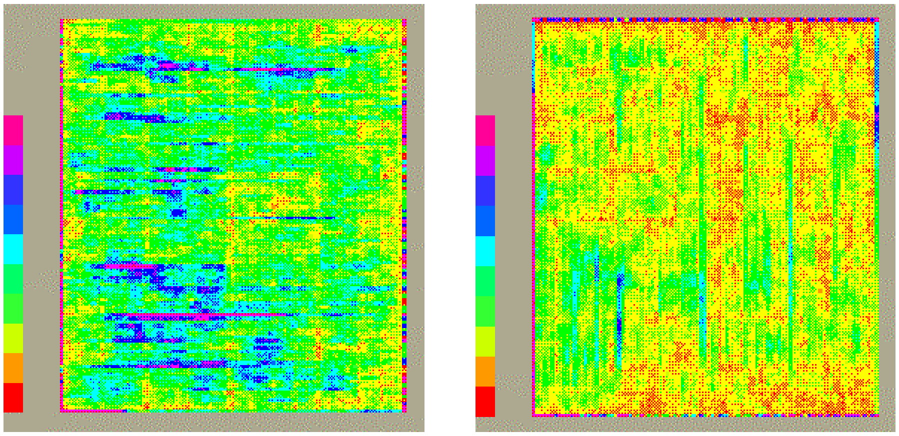
The most difficult problem in isolating the potential sources of noise involves determining which nets are most noise sensitive in the current systems-level chips being developed. In UDSM designs today the potential net-to-net interactions approach the order of $10^{12}/2$ for designs of one hundred thousand nets. It is virtually impossible to maintain a list this large and include it in any optimization process. Noise avoidance hinges on identifying the nets with noise susceptibility from a very large set of victim-aggressor pairs and immunizing them by taking corrective action as the standard-cells are placed.

Actual noise occurs only on a small set of nets (and paths) not all nets. Any effective noise solution, whether applied during design or verification, requires a methodology to isolate and focus on this set of nets where noise can cause damage.

There are two critical elements to noise isolation. The first involves construction of a noise modeling approach that in effect makes each net interaction as accurate as possible. This model depends on the technology of the process, the physical nature of the design, and the data dependent nature of the nets. The second element involves a set of isolation methods, which are applied during the physical design phase, prune the large number of victim-aggressors pairs to a manageable number.

Noise modeling using a Route Prototype

Sequence constructs a noise model, which is an extension to the Route Prototype Model (RPM) used in the Physical Design Studio product. The RPM model is used to prototype routing to predict the electrical impact of specific routes. The RPM model adapts based on changes with respect to placement density, routing congestion & pin out



Example Congestion Map (horizontal and vertical) used in route estimation characteristics.

The RPM model is a unique combination of placement-based and global route-based estimations that work during the physical layout stage to match pre to post route timing and isolate potential sources of noise in UDSM. In order to achieve these timing and noise objectives the RPM model adapts to provide analysis and optimization with the best estimates of capacitance possible. For example, figure 3 represents a placement-based model view of estimated capacitance vs 3D extracted capacitance after routing. As shown here the average error is nearly 30%–not nearly good enough to drive effective optimizations. However, if you isolate the only the critical nets (shown in green) and prioritize the routes for these nets, the placement approach you can bring the pre to post route error down to within 8%.

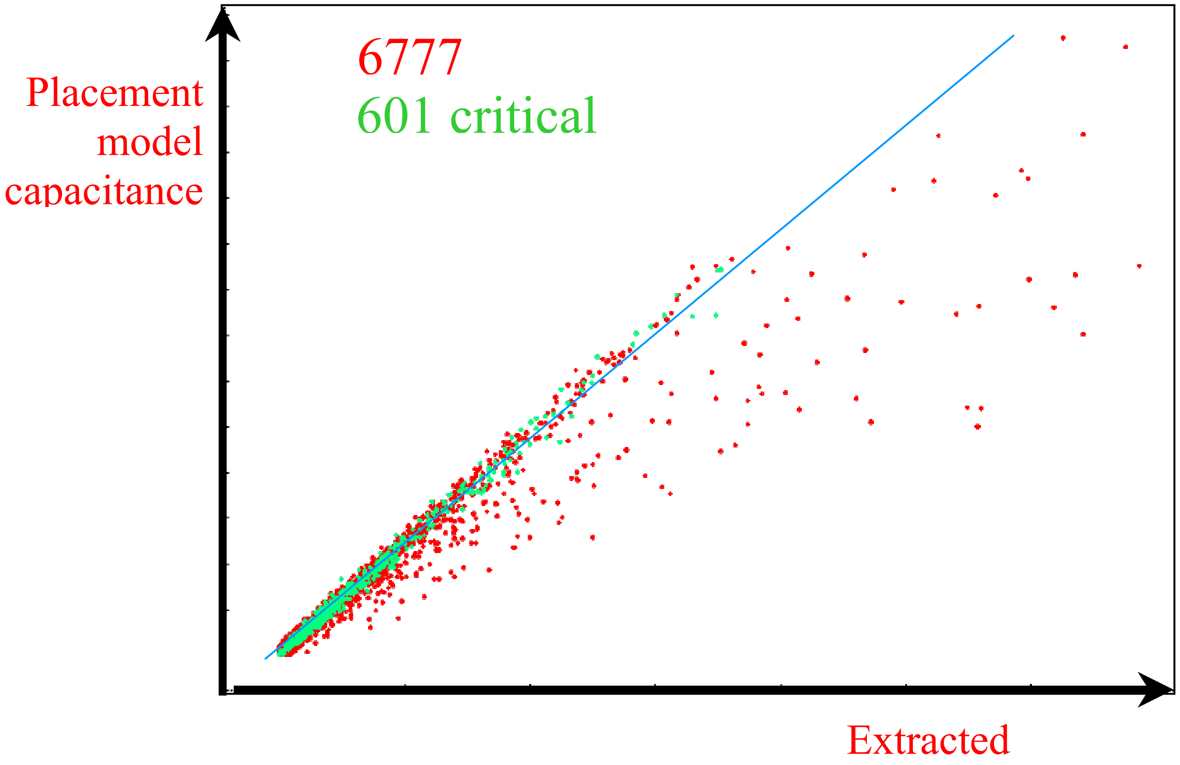
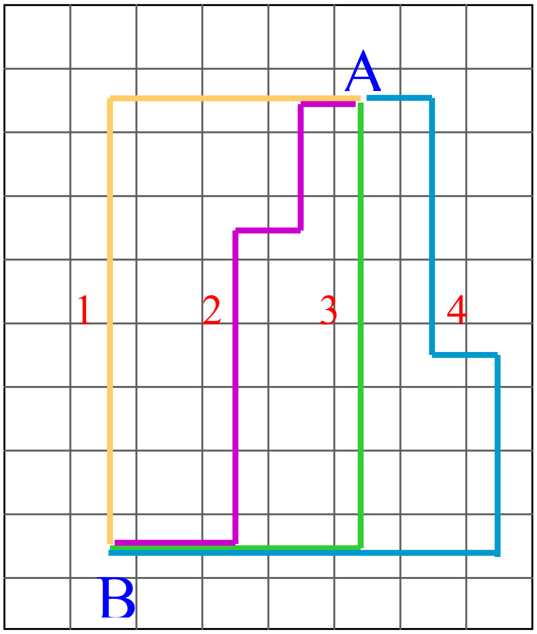


Figure 3 Estimated vs Extracted capacitance
Average error is 27 % on all nets
Average error 8% on critical nets

Next, the placement approach is extended to anticipate the route in the global route-based adaptation of the RPM. Based on the placement and the layer obstruction information the route can be anticipated as illustrated in figure 4. In this figure the model is able to predict the nature of the route between point A and point B before it actually occurs. This method of route estimation provides a measurably better estimate of the



route. With this technique the placement-based model is enhanced to provide better information to the analysis and optimization algorithms. As seen in figure 5, with the edition of routing the model now shows an average 12% error on <u>all</u> nets. Again if the critical nets are routed first these errors drop to between 3% and 5%. The adaptation of the model depends on the available information during the design and the type of optimizations required.

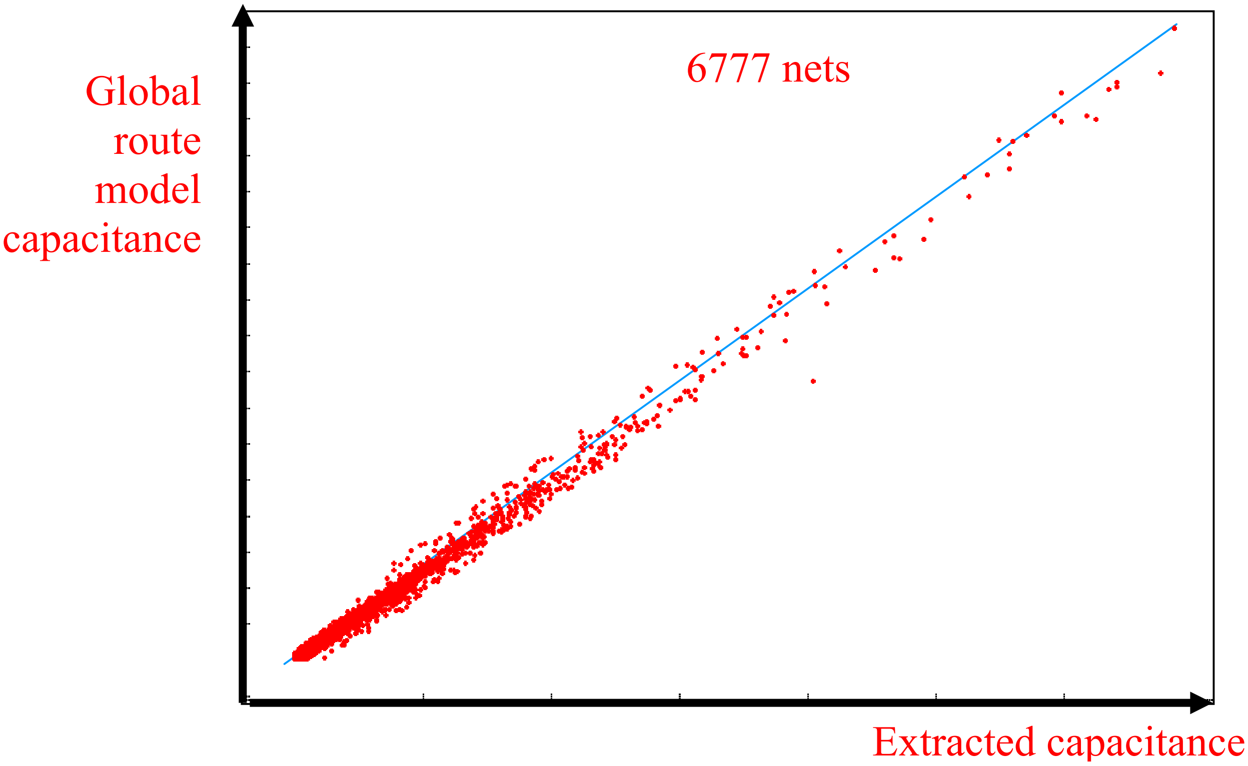


Figure 5 Estimated vs Extracted Capacitance
Average error is 12% on all nets
Average error is 5% on critical nets

Net Isolation techniques

Sequence's approach to immunizing at the design stage uses four distinct types of net pruning or net noise isolation. The four techniques include physical, temporal, electrical and critical. The objective of this type of isolation is to limit the potential victim-aggressor interactions to a "significant" but manageable number. The techniques applied are describe as follows:

Physical:	Look at nets that will interact based on their proximity.
Temporal:	Look at nets where switching events will occur within a time window that can cause noise issues.
Electrical:	Look at nets where the pin-cap to wire-cap and strength make coupling significant.
Critical:	Focus on nets where the delay impact of noise creates critical timing paths.

Physical net isolation is needed in the narrowing process because the location of each net determines the degree to which it interacts with surrounding nets. Physical pruning involves looking in detail at the placement and doing route estimates base on the pin locations. During this process a bounding box approach is used to determine whether estimated routes will cause two nets to interact. Using a conservative estimate it is possible to determine the interactions that will occur. Of course this technique then eliminates a large number of nets in the design that will not interact because they are not anywhere close to the aggressor net being analyzed. Of course the Route Prototype Model plays and important roll in the process since it varies based on placement density and routing congestion.

Temporal isolation is the most important isolation technique for data dependent noise. The degree of damaging interaction between even highly couple nets is determined by whether a particular event occurs at a time when it would cause problems. Temporal pruning involves constructing windows of events as they propagate through the design. Understanding what these switching events are and where they might occur allows the construction of conservative timing windows, which determine whether two nets will interact. Using this approach a number of nets that may be close to one another but don't interact temporally are pruned out.

Electrical isolation looks at how the capacitance of the wire and the pin and the drive strength of the gate interact. If the net is driven appropriately the coupling capacitance will not impact the delay. Likewise when the pin capacitance dominates the wire cap as in very short nets the contribution of the coupling capacitance to the overall wire capacitance although significant is not sufficient to be of concern in terms of noise coupling.

Critical path monitoring is the final type of net isolation. Those nets that have significant coupling but are not critical within a timing margin are pruned away. Continual monitoring of all nets, when they become critical and when they are not, allows optimization process to anticipate when at the result of a change will introduce other noise problems. This look-ahead capability prevents one correction from introducing another problem.

During the process of physical layout these net isolation techniques are able to trim the sensitivity list down to a manageable number. Further, as the design is placed the list of susceptible nets is continually updated giving a fresh view of noise at every point in the optimization loop.

Identifying potential noise susceptible paths and continually updating these paths as placement and circuit optimizations take place is essential to maintaining the integrity of the noise isolation technique. The types of net isolation listed focus the placement and circuit optimizations on the most important nets. Understanding the noise delay implications of the set of corrections on non-critical nets prevents costly external iterations from layout through extraction to timing and back. Changes are only made when the overall timing improves.

By taking these steps the optimizations needed to make the design noise immune Describe the physical, temporal, electrical and critical net isolation techniques.

Generating Tight Upper Bounds

Verifying noise

One of the very difficult issues regarding noise is the particular difficulty dealing with it in a post-routing context, especially if a full-chip approach is taken. Detailed analysis at a transistor abstraction level is virtually impossible. So although the accuracy of such an approach is desirable the simulation time necessary makes such an approach intractable.

Other verification approaches are also possible. These involve abstracting to a gate level and applying static analysis techniques to generate a post-routing view of noise. These approaches are both faster and operate on a larger design. But even this approach works better if a selected set of noise susceptible nets is targeted for such an analysis.

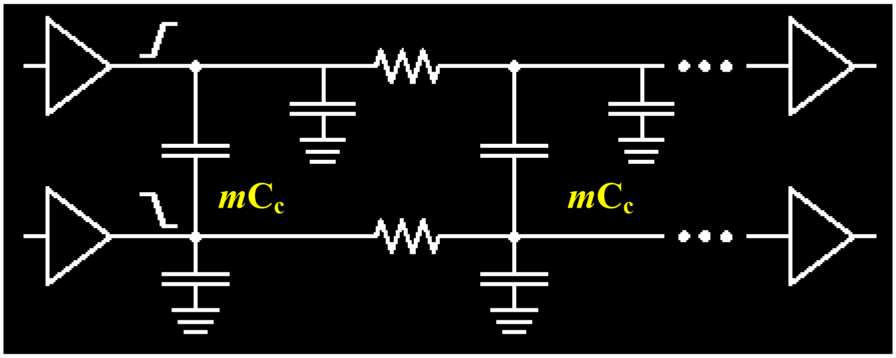
Upper Bound Approach

In the limit, if resistance is ignored the net-to-net voltage coupling interaction is approximated by:

$\Delta V_c = \Delta V_s(2C_c/C_{gnd}+2C_c)$

In this equation the number multiplying the coupling capacitance by two represents a worst-case assumption of the impact between victim and aggressor nets. As has already been shown, this does not necessarily represent an upper bound on the net-to-net interaction. In some cases this 2C approximation is to conservative, in others it is not the upper bound on the coupling interaction.

Sequence's approach is to construct a tight upper bound on the net-to-net noise interaction and use this *m* (Miller factor) to adjust the timing. Using this approach the Miller number is a calculated switching coupling transmission factor. This factor depends on events and transition times, the ratio of the slew times, and the cross coupling capacitance between the victim and aggressor nets.



Opposite direction switching: $1 \le m \le \sim 3$

Same direction switching: $\sim -1 \le m \le 1$

The typical noise waveform is positively skewed and demonstrates a set of characteristics that allow for efficient processing for on-chip interconnects. The peak voltage for net-to-net noise can be bounded and calibrated for a given RC network and process. In the case of multiple aggressor nets linear superposition holds and the total noise effect is computed by multiplying each nets respective capacitive coupling by the input slope of the aggressor net. The entire noise contribution on all nets can be computed in linear time. On chip noise is data dependent. Arrival time windows on different nets aggressor nets are different. With additional timing information only the aggressor nets with overlapping arrival time windows need to be considered when computing the maximum noise on the victim net. Experimental results have shown that the computation of peak noise predicted with this methodology is always more than the actual obtained by detailed simulation.[2]

Noise Immunization

Optimizations

Once noise is detected and its effects factored into the overall timing a number of specific circuit optimizations are possible to immunize the design. The gate sizing is adjusted up on victim nets and down on aggressor nets. Placement changes remove wires from the bounding box area where they interfere with other nets. Nets, which already demonstrate stronger resistive shielding, are broken with buffers. The optimizer looks at the situation and derives a choice. One choice along with a number of other corrective actions across the circuit is presented for trial analysis. The choices include placement adjustment, buffer insertion, cell sizing and routing directives. The trial analysis looks at the local logic area to determine whether the changes actually accomplish the required noise adjustments. After the trial analysis an incremental analysis determines the impact of changes on the complete circuit. Only after both analyses indicate that not other paths have become noise susceptible are the changes hardened in the design.

If the placement adjustment changes are still not adequate to remove the noise from the design a set of routing directives is generated. These directives include routing priorities, pitch constraints, shielding and layer recommendations. Routes that will see excessive congestion are directed to avoid interaction by placing buffers that force routes to avoid areas where anticipated noise cannot be corrected by any other means.

The final stage in design noise free is taking the post-route physical database back into Studio SI for a final set of optimizations. This step then no longer relies on estimated parasitics but the actual parasitics extracted from the 3D extractor in the flow. . It receives input from parasitic extraction in the form of either a DSPF or SPEF file. At this stage, full 3-D coupling capacitance effects and loading of the interconnect can be determined accurately and corrections for all noise violations are attempted.

One goal for making corrections at this stage of the design flow is to minimize the perturbations to the detailed routing. Optimizations with minor routing disruptions are possible only if detailed extracted information is known, including <*x,y*> coordinates. This mechanism is also referred to as in-context optimization. At this point however, because of the accurate estimation that takes place in the first stage of the noise flow, the number of specific nets that require adjustments is small. The local optimization is made without significantly impacting the design placement or the routing netlist.

---

[2] Anriudh Devgan, Efficient Coupled Noise Estimation for On-chip Interconnects, 1997

Sequence's Studio SI Option

Sequence's Studio SI product uniquely addresses the noise issues in UDSM design. When deployed in either a Cadence or Avant! flow Studio SI gives you the confidence that your designs will perform reliably. Studio SI fully utilizes the techniques already described.