A Framework for Insight into the Impact of Interconnect on 0.35-µm VLSI Performance

A design and learning tool called AIM (advanced interconnect modeling) provides VLSI circuit and technology designers with the capability to model, optimize, and scale total delay in the presence of interconnect.

by Prasad Raje

On-chip interconnect is having an increasing impact on the performance of VLSI chips. Previous work in this area from a technological perspective has concentrated mainly on the RC delay of the interconnect.[1] For cases in which the driving gate has been included in the analysis, there has not been an equal emphasis on accurate modeling of the resistance and capacitance of the interconnect and the interconnect's dependence on various dimensions.[2] The problem needs to be examined from the comprehensive perspective of including the gate in the delay analysis, using accurate models for the total delay, and including the dependence of delay on various parameters in the circuit and technological domains.

AIM (advanced interconnect modeling) is an efficient, accurate framework to analyze and optimize a fundamental building block of all VLSI critical paths, namely an arbitrary gate driving an arbitrary on-chip interconnect. AIM is a design and learning tool for both circuit and technology designers concerned about careful modeling, prediction, and scaling of total delay in the presence of interconnect.

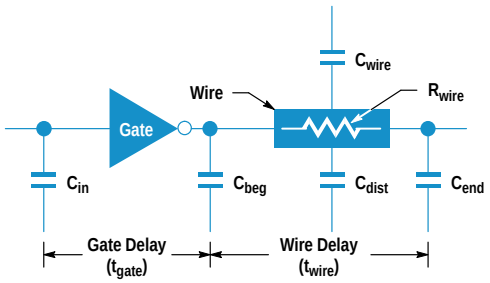


Fig. 1. Basic building block modeled by AIM.

AIM includes circuit and process technology variables while providing a framework to manage a large design space. AIM is also computationally efficient while accounting for important effects like interline capacitance and distributed RCs. It also serves as a bridge between circuit and technology designers to allow for combined optimization of interconnects in both domains. This paper describes the delay model used in AIM, its implementation and verification, and some example analyses.

System Modeled by AIM

All critical paths of CMOS/BiCMOS VLSI chips can be divided into a sequence of basic blocks, each consisting of a switched active device driving a load and an interconnect. Fig. 1 shows a typical representation of a basic building block. The switched device (logic gate) can be represented without loss of generality by a simple inverter with input capacitance C_{in}. The load consists of all nonwire capacitances, typically gate oxide and source/drain junctions. These capacitances are located at three places: at the beginning of the wire (C_{beg}), at the end of the wire (C_{end}), and distributed along the wire (C_{dist}). Note that the distributed capacitance of the wire is distinct from C_{dist} and is discussed in detail below.

The interconnect wire presents a distributed resistance (R_{wire}) and a distributed capacitance (C_{wire}). R_{wire} and C_{wire} are functions of the wire geometry shown in Fig. 2. The

Glossary

The following definitions explain terms as they are used in the context of the accompanying article.

C_{in} (input capacitance). C_{in} is proportional to the product of gate oxide capacitance per unit area, the gate length, and the gate width. The gate width is the total width of all the transistors tied to the input. C_{in} is often represented by the gate width in units of µm.

Circuit Domain. The circuit domain refers to the design realm of the circuit designer. Specifically, certain quantities are under the control of the circuit designer in the context of interconnect delay. These include the wire width, length, and space, or the gate width.

HIVE. HIVE is an internal HP software package that creates closed functions of wire capacitance components as a function of the relevant geometrical quantities. HIVE starts with the wire geometries, performs 2D numerical field simulations and arrives at closest-fit analytical functions.

Interconnect. Interconnect refers to the conducting wires on an integrated circuit chip that connect the components to each other and carry electrical signals.

Technological Domain. The technological domain refers to the design realm of the process technology designer. Certain quantities that affect gate delay in the presence of interconnect are under control of the technology designer. These quantities include wire thickness, interlayer spacing, transistor gate oxide thickness, and so on.

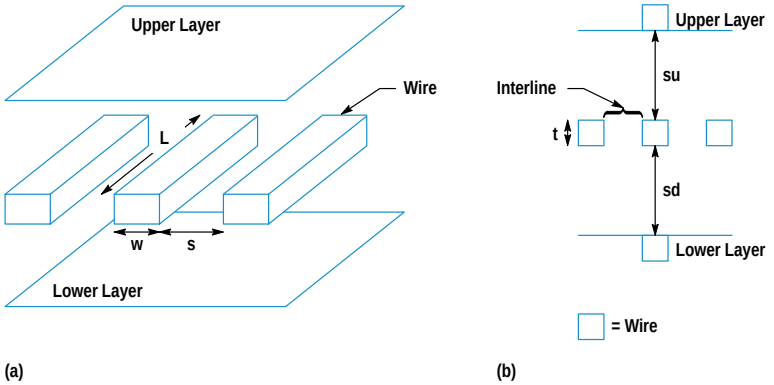


Fig. 2. Wire geometry. (a) Circuit domain variables. (b) Technology domain variables.

dimensions in Fig. 2 show the variables that represent the size (width w, length L, and thickness t) of a typical wire, and the separation of the wire from conductors that are above (su), below (sd), and adjacent (s) to it. L, w, and s are variables in the circuit domain, while t, su, and sd are variables in the technology domain.

A multilayer interconnect system, like the one shown in Fig. 2, provides different values of t, su, and sd depending on the wire layer* (polysilicon, M1, M2, M3, M4, etc.), upper conductor (M1, M2, M3, M4, or none), and lower conductor (substrate, polysilicon, M1, M2, M3, etc.). AIM allows all permutations of layer and upper and lower conductors. A layer variable in the circuit domain is defined that can take on any value selected from these permutations.

The upper and lower conductors, when present, are assumed to be continuous plates of conductors. To a first order, this is an acceptable approximation when the upper and lower layers have densely spaced wires. The adjacent wires are assumed to be equidistant on both sides. For simplicity, the R_{wire} and C_{wire} interconnect components are assumed to be uniformly distributed along the length of the wire. Any changes in layer or wire geometry that change R_{wire} or C_{wire} are important for capacitance extractions in actual layouts. There is no value to introducing this complication into AIM, where larger trends in delay on various parameters are of interest. Also, there would be no generality in choosing a particular change in R_{wire} or C_{wire} along the wire.

AIM Delay Model

A key feature of AIM is that the logic gate delay is included as a full participant in the interconnect delay analysis. Gate delay is defined as the delay from the gate input to the beginning of the wire (see Fig. 1). Wire delay is the delay from the beginning of the wire to the end, and includes the effect of C_{dist}. The total delay is the sum of gate and wire delays:

$$\text{delay} = t_{gate} + t_{wire}. \quad (1)$$

The gate delay is expressed in a delay-versus-fanout format instead of simplifying the gate as an equivalent resistance as described in reference 2. The fanout is the sum of all the capacitance seen by the gate as if it were all lumped at the output, divided by the input capacitance:

$$t_{gate} = t_0 + \text{slope}\ (C_{beg} + C_{wire} + C_{dist} + C_{end})/C_{in} - k \times t_{wire}, \quad (2)$$

where t_0 is the y-intercept of the of the delay-versus-fanout curve, slope is the slope of the delay-versus-fanout curve, and k is an empirical constant that represents a correction factor to account for "hiding" distant capacitance along a resistive wire ($0 < k < 1$ and is typically 0.5).

The wire delay is the usual RC delay including the distributed nature of both the wire and nonwire capacitance:

$$t_{wire} = R_{wire} \times (C_{wire}/2 + C_{dist}/2 + C_{end}) \quad (3)$$

C_{wire} consists of the interline capacitance of the adjacent wires on the same layer, and the interlayer capacitance of the upper and lower layers.

$$C_{wire} = C_{interlayer} + C_{interline} \quad (4)$$

The interlayer and interline components are expressed simply as the respective parallel plate capacitances. This formulation intentionally does not include fringing effects to make it easy to express the optimum width and optimum thickness formulas in the next section. Fringing effects are described in the section "Accurate Delay Modeling" on page 3.

The upper and lower layers are assumed to be quiescent but adjacent wires are allowed to have a signal switching in the opposite sense. A variable m accounts for the Miller effect and effectively doubles the value of the interline capacitance when the adjacent wires switch simultaneously in the opposite direction. If m =1 the adjacent wires are quiescent, and if m = 2 the wires are switched.

$$C_{wire} = \varepsilon wL/sd + \varepsilon wL/su + 2m \times \varepsilon tL/s \quad (5)$$

$$R_{wire} = \rho L/tw \quad (6)$$

where ε is the permittivity of the dielectric and ρ is the resistivity of the metal. The more general case of C_{wire} for upper and lower conductor switching can easily be constructed with extra Miller variables.

Optimum Wire Width and Thickness

The wire width w and thickness t appear in the numerator and denominator of the total delay expression. A larger width or thickness implies an inversely smaller R_{wire} but a larger C_{wire}. The net effect is a reduction of the wire delay (t_{wire}), but an increase in gate delay (t_{gate}). The total delay therefore is optimum at an intermediate value of w or t. The total delay is differentiated with respect to w and t to give the optimum values w_{opt} (optimal wire width) and t_{opt} (optimal wire thickness) at which the delay is a minimum.

* M1, M2, M3, and so on represent different types of metal layers.

$$w_{opt} = \sqrt{\frac{(1-k)}{2}\frac{r}{\varepsilon}\frac{C_{in}}{slope}\frac{sd \times su}{(sd+su)}\left(\frac{2m\varepsilon L}{s}+\frac{C_{dist}+2C_{end}}{t}\right)} \quad (7)$$

$$t_{opt} = \sqrt{\frac{(1-k)}{2}\frac{r}{\varepsilon}\frac{C_{in}}{slope}\frac{s}{2m}\left(\frac{\varepsilon L}{sd}+\frac{\varepsilon L}{su}+\frac{C_{dist}+2C_{end}}{w}\right)} \quad (8)$$

For the circuit designer, w_{opt} is an important quantity and one that can potentially be changed for each different net in a circuit to achieve the lowest delay. Wire widths must be increased:

- when driving longer wires (larger L)
- in the presence of an extra load along or at the end of the wire (greater C_{dist} and C_{end})
- when driving with stronger drivers (larger C_{in} or smaller slope)
- when adjacent wires are switching ($m > 1$).

For the technology designer, optimal wire thickness is the important quantity. The difficulty here is that it is not possible to change the wire thickness for each different net. Therefore, estimations must be made of parameters such as the expected range of wire length, driver size, wire spacing, and nonwire loads in the chips that are expected to be designed in the technology. Once these parameters are known, then one can state that the wire should be designed to be thicker when it is expected to be longer, driven by bigger drivers, or in the presence of a significant nonwire load. Equation 8 provides an analytical basis for the well-known interconnect design guideline that upper layers of metal that go over longer distances should be made thicker.

There is a subtle interaction between the optimum width and the optimum thickness of the wire. The variable w_{opt} depends on the thickness of the wire and vice versa. Thus, a wider wire may lead to a smaller optimum thickness according to equation 8. If the nonwire loads C_{dist} and C_{end} are small compared to both the interline and interlayer capacitances, then w_{opt} has no dependence on wire thickness, and t_{opt} has no dependence on wire width. In this case the technology designer can optimize the wire thickness from a delay standpoint without any consideration for the width of the wire.

Accurate Delay Modeling

The analytical delay model of the previous section provides important insight into the various parameters affecting wire delay. To make specific predictions about interconnect behavior in a technology, it is necessary to use accurate numerical values of the different components of the delay. These components are provided in AIM by HIVE[3] and Spice. The analytical expressions from HIVE are modified so that they can be used with Mathematica. Mathematica is an interactive, interpreted programming environment that allows one to do such things as express analytical equations, perform analysis, and create 2D and 3D plots.

HIVE for Wires and Spice for Gates

HIVE provides for some second-order effects not included in the C_{wire} expression (equation 5). The interlayer capacitances are still linearly proportional to width (w), but the second-order dependence on interline space (s) is included. This is the fringing effect which reduces the interlayer capacitances as interline space is reduced. The interlayer capacitance has the form:

$$C_{interlayer} = F_6(s) + \frac{w - w_{min}}{w_{max} - w_{min}}\left(G_6(s) - F_6(s)\right) \quad (9)$$

where $F_6(s)$ and $G_6(s)$ are sixth-order polynomial functions of s and $w_{min} < w < w_{max}$ is the range over which the fitting function applies.

The dependence of interline capacitance on s is modeled as a sixth-order polynomial rather than the simple linear s term in the denominator. The interline capacitance has the form:

$$C_{interline} = 1/H_6(s) + \frac{w - w_{min}}{w_{max} - w_{min}}\left(1/J_6(s) - 1/H_6(s)\right) \quad (10)$$

where $H_6(s)$ and $J_6(s)$ are sixth-order polynomials.

HIVE uses two-dimensional finite element simulation of actual geometries in an IC technology to obtain the coefficients of the sixth-order polynomials given above. Further, accurate values of these components are available for all values of the layer variable (M4 over substrate or M3 over M2 under M4, etc.).

Spice simulations on various basic gates are performed to obtain accurate t_0 and slope values in the technology of interest. For the 0.35-μm CMOS technology (CMOSA) used in this paper, t_0 = 40 ps and slope = 23 ps/fanout. Empirical studies are also carried out to estimate the value of k in equation 2. The value of k lies between 0.4 and 0.6 in CMOSA technology. The dependence of gate delay on input slope could be included with the addition of one or two more fitting parameters. (For simplicity this is not introduced in the first implementation of the AIM model.) Also, the emphasis in the delay analysis is on the trends in delay as a function of various wire parameters. These dependencies are, to a first-order approximation, independent of the input waveform slope at the gate.

The delay predicted by the AIM delay model is compared to a Spice simulated delay of the same gate with the wire represented by a HIVE subcircuit. Delay calculations for 336 data points of various wire widths, lengths, and gate sizes are obtained and they show that the margin of error between the AIM and Spice results is <3% for 60% of the samples, <5% for 75% of the samples, and <10% for 93% of the samples. This provides confidence in the predictions made with the AIM delay model.

Implementation in Mathematica

With the more complicated wire capacitance expressions from HIVE, the delay model is no longer tractable by hand, but it is still in an analytical form that can be coded into Mathematica expressions. The basic delay expressions and subexpressions are in a single file. The technology dependent coefficients in the wire capacitance expressions are in a separate technology file. This allows different interconnect technologies to be analyzed by simply changing the technology file.

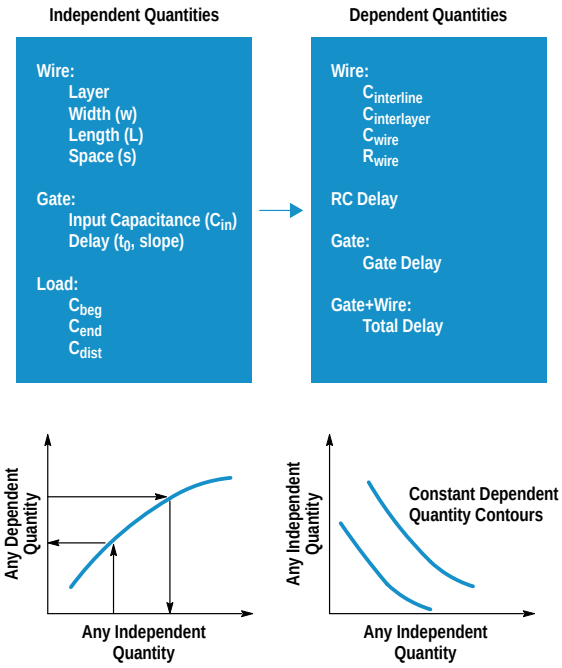


Fig. 3. AIM implementation showing the different types of analyses that are possible.

The analytical implementation in Mathematica allows powerful, fast, and accurate analyses of various dependent observable quantities as functions of various independent quantities (see Fig. 3). Independent quantities can be in numerical or symbolic form and include properties of the wire, gate, or load. These properties are typically specified by values in an input file. However, one or more of these properties may be left in symbolic form. The dependent quantities are expressions or formulas defined in terms of the independent quantities. Each piece of the wire capacitance, such as $C_{interline}$ and $C_{interlayer}$, is available separately. The most important quantity of interest is of course the total delay.

AIM provides standard routines to plot any dependent quantity as a function of any independent quantity. Similarly, given a dependent quantity and all but one independent quantity, the unknown independent quantity can be obtained. More complex analyses consist of plotting a dependent quantity in 3D versus two independent quantities, or plotting a contour plot of a constant dependent quantity with two independent quantities on the x- and y-axes. Examples of these analyses are discussed below.

Mathematica versus a Spreadsheet

The AIM model as it is implemented in Mathematica is highly customizable and many more types of analyses are possible. However, there is a barrier to using this implementation for designers not familiar with Mathematica. The model could be implemented in a spreadsheet, and although the graphical analyses would not be as easy, obtaining quick numerical results would be easier.

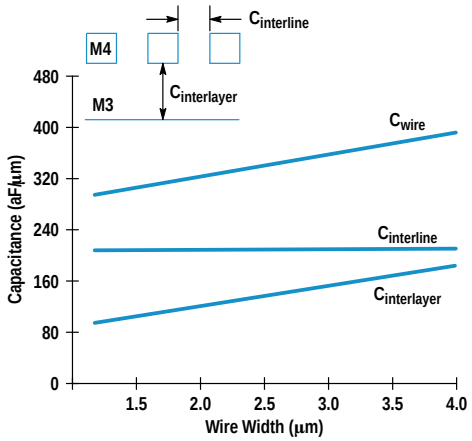


Fig. 4. Wire capacitance components for minimum-spaced M4 over M3.

Insights Provided by AIM

The following examples describe some special insights that are possible with the AIM model.

Interline Capacitance and Fringing

A simple but insightful analysis with AIM is the M4 wire capacitance versus the width of the wire. Fig. 4 shows the interline, interlayer, and total wire capacitance for minimum-spaced (1.6 μm) M4 wires over M3. Adjacent wires are assumed to be switching so that m = 2 (equation 5). The first observation is that the interline capacitance is larger than the interlayer capacitance. The interlayer capacitance increases linearly with width as expressed in equation 5. However, at zero width the extrapolated $C_{interlayer}$ line is not zero because of the fringing component. This behavior is included in the HIVE expressions.

Fig. 5 shows the same M4 wire with the only change being that it is over substrate instead of M3. There is a dramatic difference in the capacitance curves. The reduction in

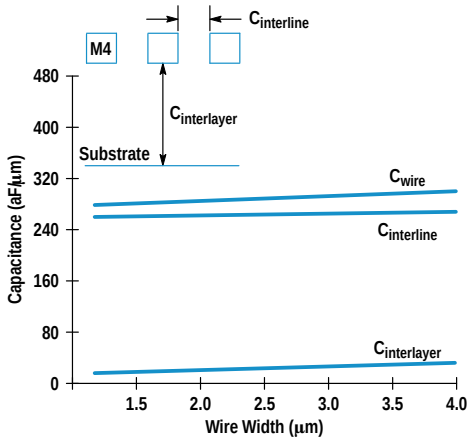


Fig. 5. Wire capacitance components for minimum-spaced M4 over substrate.

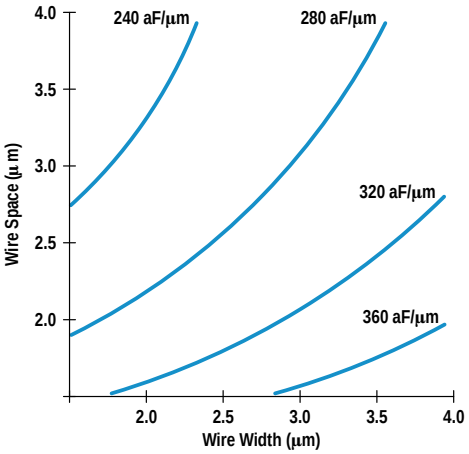


Fig. 6. Constant-capacitance contours for M4 over M3.

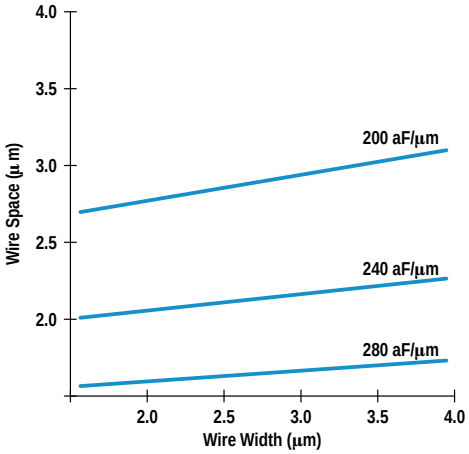


Fig. 7. Constant-capacitance contours for M4 over substrate.

$C_{interlayer}$ might be expected from the larger distance of the wire to the substrate, but the significant feature is that the total capacitance is not significantly reduced despite the larger distance from the substrate. This is because $C_{interline}$ has increased. This increase is because more lines of flux from the lower surface of the wire terminate on the adjacent wire instead of the lower conductor. In other words, the fringing component has increased. Another result of fringing is that the interlayer capacitance now has a very weak dependence on wire width.

Thus, the conventional wisdom that upper layers of metal enjoy much reduced capacitance because of their distance from the substrate does not hold. For one thing, the upper layers may run over wires in the immediate lower layer and even when they do not, the total capacitance is not much lower.

Capacitance versus Width and Space

A visual representation of the relative importance of width and space is obtained from a contour plot of wire capacitance in a 2D space of wire width and interline spacing. Fig. 6 shows constant-capacitance contours for M4 over M3 lines. The data in Fig. 6 is a superset of the information in Fig. 4. Along any horizontal line in Fig. 6 several contours are cut, indicating a rapid increase in C_{wire} (interline and interlayer capacitance) with width. The dependence on space is also significant. Fig. 7 shows the contours for M4 wire over substrate. The contours appear more horizontal indicating that there is a weak dependence on wire width and that a reduction in wire capacitance is easier to achieve with an increase in (interline) spacing. The contours have reduced in value but the reduction is substantial only when the wire spacing is large and the width is large. Such contours can be made for all the metal levels to provide a quick ready reference of wire capacitance for a range of geometries.

Optimizing the Width of Wires

RC delay is an important factor that causes a circuit designer to choose wider wires when they are long. However, it is important to realize that larger width comes with an increase in the total capacitance of the wire and therefore a possible increase in the total delay. There is an optimum width of the wire at which the total delay is a minimum. Equation 7 expresses the dependence of the optimum width on various parameters. Fig. 8 shows an example plot of the gate and wire delay components as a function of wire width. The optimum width is only 1.5 µm, which is relatively small for a 5000-µm long wire. There is a built-in function in AIM that provides the optimum width when the remaining variables in the system are specified.

Minimum Metal Width Design Rules

The optimum wire width illustrated in the previous section depends on a number of parameters, the most important being length L, gate width C_{in}, wire spacing s, and wire layer. AIM can rapidly generate the optimum width for a large range of these parameters. Fig. 9 shows the optimum width versus length for a few example cases with a 200-fF fixed load at the end of the wire. The curves are not smooth because of the limited number of data points generated and the slow variation of delay with length. The approximately square root dependence on length predicted in equation 7 is illustrated in Fig. 9. If the gate width is increased to 200 µm, the curve moves to a higher w_{opt} (optimum wire width) as predicted in equation 7. A larger interline spacing reduces w_{opt} as illustrated by curve C, but this dependence is not strong. Curve D shows the optimum wire width for an M1 wire and is surprisingly close to A for the same conditions. This is because the interlayer spacings are similar when M1

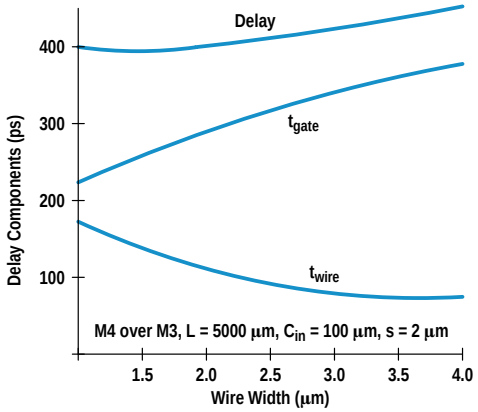
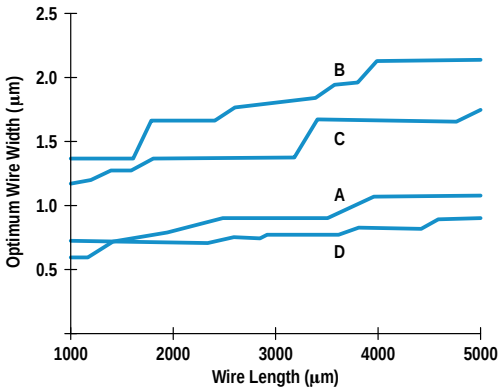


Fig. 8. Total delay versus width. RC delay is reduced but wire gate delay is increased with larger wire width.



Legend

Curve	Layer	Spacing	Gate Width
A	M4 over M3	1.6 μm	50 μm
B	M4 over M3	1.6 μm	200 μm
C	M4 over M3	10 μm	200 μm
D	M1 over Substrate	1.6 μm	50 μm

Fig. 9. Optimum wire width versus wire length under different layer, spacing, and gate width conditions.

and M4 are both surrounded by upper and lower conductors. Also, the dependence of w_{opt} on wire thickness is very weak.

If the minimum width design rule for M4 is 1.3 μm, then Fig. 9 shows that the optimum width can be smaller than the design rule width for many reasonable conditions. A full range of high and low values for the wire lengths, spaces, and inverter sizes can be simulated to determine the range of optimum widths of the wire. This can then provide guidelines for technology designers in setting minimum design rules for wires.

Delay versus Wire Length and Driver Size

An important analysis that encapsulates a lot of useful information for a circuit designer in a single figure is a plot of delay contours with gate width along the x axis and wire length along the y axis. Fig. 10 shows such a plot for M4 over M3 with 1.5-μm wire width, 2-μm spacing, and a 100-fF

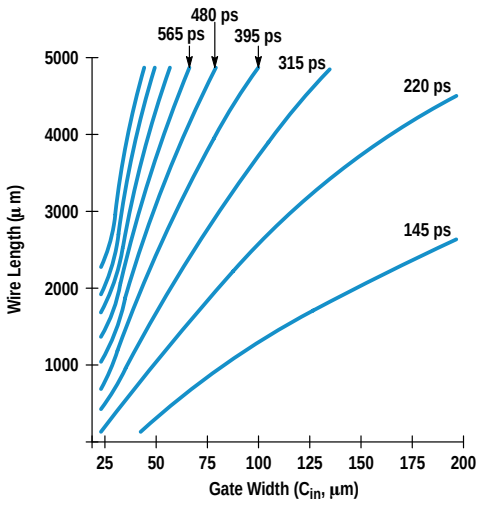


Fig. 10. Constant delay contours (ps) in a space of gate width (μm) and wire length (μm).

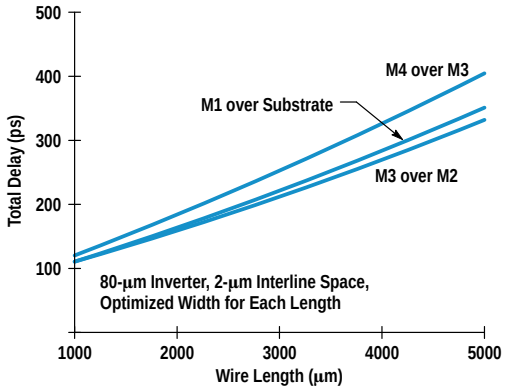


Fig. 11. Total delay versus wire length for various levels.

load at the end. This allows a ready reference for quickly looking up the gate width that would be needed to drive a certain wire length with a desired delay. Alternatively, the dependence of delay on wire length for a given gate width can be seen. For example, a 3000-μm long wire would require an 80-μm gate width to achieve less than 300-ps total delay. A 50-μm gate width would see a very rapid increase in its delay beyond a 2000-μm length as seen by the bunching of the contours. Similar plots for other wires or other conditions can easily be generated using built-in functions provided in AIM.

Delay versus Wire Layer

A common misconception is that an upper level of metal is always faster when driving long distances on the order of a few thousand micrometers. To analyze this, the built-in routines in AIM are used to plot total delay (a dependent quantity) versus wire length (an independent quantity). Fig. 11 shows this plot for M4, M3, and M1 wires, all with minimum interlayer spaces and a 2-μm interline space. The wire width for each level is optimized for each length as discussed earlier. A large 80-μm inverter size is chosen to emphasize the RC delay over the gate delay. The total delay is larger for the M4 wire than the M1 wire! Even though the wire delay of the M4 wire is lower, its higher capacitance leads to a larger gate delay. Also, the M4 wire suffers from higher interline capacitance than the M3 wire because of the passivating nitride over the M4 wire. While the M3 wire has the lowest delay, the M1 delay is remarkably close. The optimum width for a 5000-μm long M4 wire is 1.4 μm and that for an M1 wire with the same length is 1.2 μm. The fact that the M4 wire is slower than the M1 wire is not an artifact of AIM, but has been confirmed by Spice simulation with HIVE subcircuits.

Pitfalls in Algorithmic Shrinking

Many VLSI chips are shrunk from one generation of technology to the next by algorithmically scaling all the layers in the design to match the design rules of the new technology. The result on the wires in the circuit domain is a reduction in widths, spaces, and lengths. There may also be scaling of wire dimensions (thickness, interlayer spacings) in the technology domain. The result on the FETs is a reduction in gate width and length and also source/drain areas. It is relatively easy to predict the performance scaling of the delay for logic circuits that have a relatively small amount of interconnect. It is much harder to predict delay scaling in critical paths that have a large amount of interconnect. This is because the

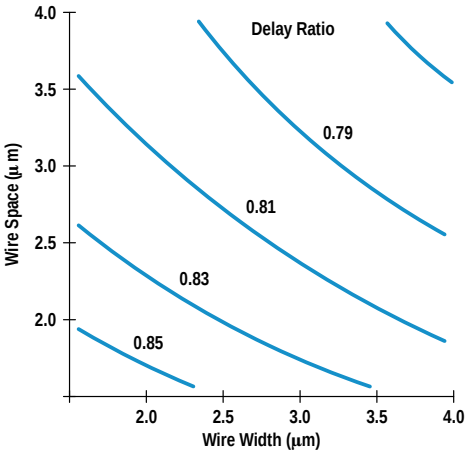


Fig. 12. Delay ratio CMOSA/CMOSB with 0.75× wire and FET shrink. Unscaled gate width = 100 μm, wire length = 5000 μm.

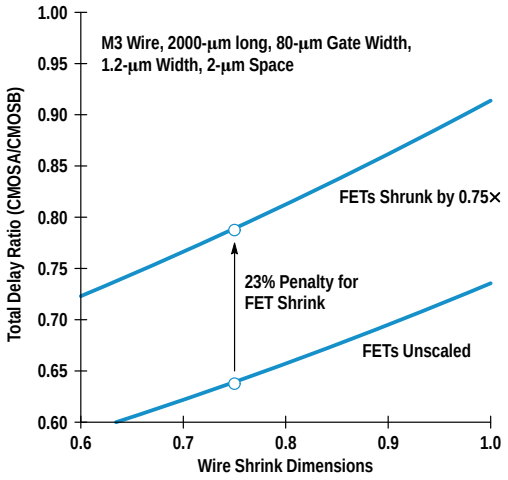


Fig. 14. Total delay scaling in going from CMOSB to CMOSA.

scaling factor depends on the interplay of a large number of parameters. AIM allows a rapid exploration of the design space and can pinpoint scenarios in which the delay improvement could be compromised.

To illustrate this capability, the scaling of delay from a 0.5-μm CMOS technology (CMOSB) to the 0.35-μm CMOSA technology is observed for a range of values of different variables. The gate width in each technology is characterized by the t_0 and slope values (see equation 2). The values for a CMOSA inverter are t_0 = 40 ps and slope = 23 ps/fanout. The values for a CMOSB inverter are 40% higher. This accounts for the change in the FETs in the technology domain. The change in the interconnects in the technology domain is accounted for by a new set of HIVE coefficients for capacitances, which get translated into a new AIM technology file. In the circuit domain, a shrink factor of 0.75 is applied to all wire dimensions (width, spacing, and length) and to the gate width and nonwire loads.

The resultant scaling of wire capacitance, RC delay, and so on is taken care of in AIM and only the circuit-domain scaling parameters are supplied as inputs. The data in Figs. 12 and 13 shows the delay ratio (CMOSA/CMOSB) as a function of wire space, wire width, wire length, and gate width. Fig. 12 shows that delay ratios are better for large wire space and wire width. Large wire space is often not available because of pitch limitations, while large wire widths can increase total delay. Large wire lengths are detrimental to delay scaling as are small FETs. AIM can generate these concise reference charts showing the dependence of delay scaling on various important parameters.

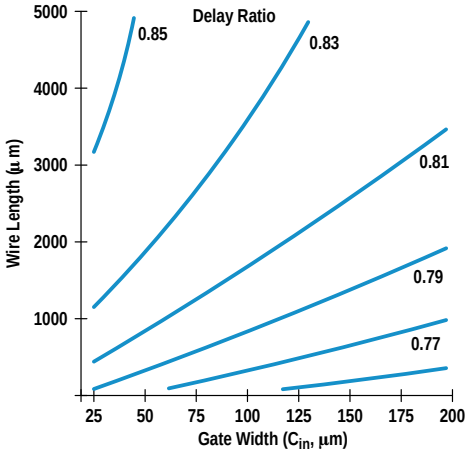


Fig. 13. Delay ratio in going from CMOSB to CMOSA with 0.75× wire and FET shrink, space = 2.4 μm, width = 2 μm.

The improvement in the delay is not significant for the wire dominated basic blocks considered in this paper. However, it is incorrect to assume that it is only the wire RC delay that is the cause of the problem. The reason is that when the wire dimensions scale down, the FET widths also scale down, reducing the drive to the wires. Further, even though the wire length reduction is beneficial to capacitance, wire spacing reduction increases interline capacitance. Also, fringing effects undermine the linear capacitance reduction expected from simplistic scaling of wire width. The resistance of the scaled wire is constant if the thickness stays the same. The net result is that both the gate delay and the wire delay do not scale well.

AIM allows one to examine the delay ratio of a typical basic block with independently varied scaling factors for FET and wire scaling. Fig. 14 shows the ratio of CMOSA to CMOSB delay as a function of wire shrink dimensions. If the FETs are kept unscaled and only the wires are shrunk, the delay ratio is 0.63. This substantial improvement would also be obtained for basic blocks that do not have significant wire loading. However, it is incorrect to expect this number when a whole chip is shrunk and the critical path consists of many wire-dominated basic blocks. Fig. 14 illustrates this scenario when the FETs are shrunk by 0.75×. The delay ratio is now only 0.78, a 23% increase over the previous case. This illustrates the importance of the capacitive load of the wire. AIM can be used to examine each net of a chip design to flag those nets that are susceptible to poor delay scaling if they are shrunk. These nets could either be redesigned or special cases made to keep the selected FET widths unscaled in a shrink. This can lead to guidelines for "design for shrinkability." In the meantime, the statement can be made that the ultimate technologically capable delay improvement is not possible in a pure shrink strategy.

Summary

AIM has been presented as a comprehensive framework to understand and optimize the performance of basic blocks in VLSI critical paths. The interconnect is modeled with highly accurate expressions that account for many second-order effects, and the gate driving the interconnect has been included as a full participant in the analyses. The design space is large because of the many variables in both the technology and circuit domains. This has been managed with a simple but accurate analytical delay model. The implementation in Mathematica provides quick and efficient analyses of many different types of technology and circuit variables.

The examples shown have illustrated only some of the capabilities of AIM. The myth of much lower capacitance for upper levels of metal has been shown to be unfounded. A visual insight into the relative influence of wire width and spacing on wire capacitance has been provided. The importance of the optimization of wire width has been demonstrated and its dependence on various parameters has been correlated with simple analytical equations. It has been shown that metal widths are often made larger than necessary and some minimum width rules may preclude optimal delay. A reference chart for circuit designers showing delay versus wire length and gate size has been demonstrated. It has been shown that upper levels of metal are not necessarily the best choice even for long wires. Algorithmic shrinking of chips from one technology to the next has been shown to suffer a substantial penalty in wire dominated basic blocks. The gate capacitive delay scales as poorly as does the wire RC delay.

Acknowledgments

The author would like to thank Gene Emerson for his encouragement, support, and guidance. Thanks also to K.J. Chang and Soo Young Oh for developing HIVE.

References

1. K.C. Saraswat and F. Mohammadi, "Effect of Scaling of Interconnects on the Time Delay of VLSI Circuits," *IEEE Transactions on Electron Devices,* Vol. ED-29, no. 4, April 1982, p. 645.

2. T. Sakurai, "Approximation of Wiring Delay in MOSFET LSI," *IEEE Journal of Solid-State Circuits,* Vol. SC-18, no. 4, August 1983, p. 418.

3. K.J. Chang, et al, "Parametrized Spice Subcircuits for Multilevel Interconnect Modeling and Simulation," *IEEE Transactions on Circuits and Systems,* Vol. 39, no. 11, November 1992.