

 Open access • Proceedings Article • DOI:10.1145/309847.309985

A novel VLSI layout fabric for deep sub-micron applications — [Source link](#)

Sunil P. Khatri, Amit Mehrotra, Robert K. Brayton, Alberto Sangiovanni-Vincentelli ...+1 more authors

Published on: 01 Jun 1999 - Design Automation Conference

Topics: Integrated circuit layout, Physical design, Circuit extraction, IC layout editor and Design layout record

Related papers:

- [Cross-talk immune VLSI design using a network of PLAs embedded in a regular layout fabric](#)
- [Circuits, interconnections, and packaging for VLSI](#)
- [Getting to the bottom of deep submicron](#)
- [Cross-talk noise immune VLSI design using regular layout fabrics](#)
- [Crosstalk reduction for VLSI](#)

Share this paper:    

View more about this paper here: <https://typeset.io/papers/a-novel-vlsi-layout-fabric-for-deep-sub-micron-applications-3nmokx2g50>

M7-8	W (μ)	–	2.0	1.2
	space (μ)	–	2.0	1.2
	t_{ins} (nm)	–	1400	900
M9	W (μ)	–	–	2.0
	space (μ)	–	–	2.0
	t_{ins} (nm)	–	–	1400
Via (M1-M2)	size (μ)	0.5	0.2	0.1
	R (Ω)	0.46	1.43	3.27
	ϵ_r	3.3	2	1.5

Table 1: “Strawman” process parameters

Next we study the changes in interconnect parasitics as feature size decreases in modern VLSI fabrication processes. Of the three processes we consider, the first process is used in aggressive circuit designs today, while the remaining processes are still a few years from being used.

Interconnect parasitic characterization was performed analytically as well as experimentally, using a 3-dimensional parasitic extractor called *Space3D* [3]. The input to *Space3D* is a 3-dimensional circuit layout, and the output is the value of the parasitics between different features of that layout. *Space3D* uses a boundary element method to compute interconnect capacitances.

We found that the experiments and the analysis both suggested that the capacitance of a wire to its adjacent wire was increasing as a fraction of the total capacitance. Because of this, a signal’s delay and integrity depend heavily on the switching activity of its neighboring wires. So, in order to design reliable circuits, each unique interconnect configuration would need separate characterization. The task of characterizing circuit delay would involve running a 3-dimensional parasitic extractor, which is highly compute-intensive. Additionally, the large amounts of data from these extractions would be difficult to manage.

For these reasons, we predict that DSM VLSI design using the familiar layout paradigms will not be feasible. In order to eliminate the uncertainty in the effective capacitances of a wire and the resulting uncertainty in its delay, we introduce a new layout methodology in this paper. The primary goal of our methodology is to ensure that each signal has a constant effective capacitance, regardless of the whether other signals in its neighborhood undergo any transitions.

We achieve this by alternating signal wires with power and

ground wires in the layout, on *all* metal layers. If a signal wire is labeled S , a power wire labeled V and a ground wire is labeled G , then on *every* metal layer, any sequence of wires will be labeled $\dots VSGSVSGSV \dots$. Also, metal wires on any layer run perpendicular to those in the layers above and below it. As a result, the entire a chip is maximally gridded with wires in all directions.

The advantages of this are manifold:

- First of all, the capacitance of any signal wire is entirely predictable, since the immediate neighbors of any signal wire are always one power and one ground wire. The capacitance of a signal wire to the nearest signal wires is negligible. We determined that the capacitance of a wire to its immediate neighbor is at least 10 to 15 times larger than its capacitance to its neighbor's neighbor. Hence the effect, on any signal, of its neighboring signal wires undergoing transitions is negligible. We determined it to be within $\pm 2\%$ of the nominal delay, a impressive improvement from the existing layout paradigms of today, for which the range of delays under these conditions were determined by us to be 2.47:1.

- Secondly, the routing of power and ground to the entire chip is simultaneously achieved in this way. At every point where a power (or ground) wire on metal layer i overlaps with a power (or ground) wire on metal layer $i - 1$, a via is introduced. Given the large number of such intersections, the power and ground resistance at every point is held very low, and almost constant. This gridding of power and ground gives the layout the appearance of a "fabric".

- Thirdly, each signal has a current return path which is adjacent to it, hence its inductance is very low. In the existing layout paradigms of today, the inductance of signals can vary greatly, since different wires have current return paths a different distance apart, depending on the exact layout of the circuit in the neighborhood of the signal.

In the subsequent sections, we elaborate and substantiate our claims. Section 2 describes previous attempts at achieving well-characterized layout topologies. Section 3 describes our layout scheme further, elaborating on many important practical chip design issues and how we handle them. Section 4 describes preliminary results that we have obtained using our layout topology. Finally, in Section 5, we make concluding comments and discuss further work that needs to be done in this area.

2 Previous Work

The problems faced in designing and manufacturing chips with DSM processes are not entirely new. Over the last few years, these problems have slowly become significant enough that academia and industry have begun to take notice.

In the past, some techniques to ensure characterizability and reliability of designs have been proposed and implemented. For example, in the DEC Alpha chip [4], metal layers 3 and 6 (in a 6 layer metal process) were exclusively dedicated to power and ground routing. In this scheme, a signal on any of the remaining layers has a constant voltage plane either above or below it, thus ensuring that a majority of its total capacitance is to a node of constant voltage. In the absence of such a scheme, signal wires on higher metal layers would have very small capacitances to a node of constant voltage, on account of their large distance from the substrate. Also, these wires would have relatively large capacitances to neighboring signal conductors. In such a situation, if the neighboring conductor would undergo a signal transition, it could couple a large noise voltage into the signal of interest, resulting in a loss of signal integrity. It could also result in altered delays when the neighboring signal conductor undergoes an signal transition while the wire of interest is transitioning.

The above solution worked well for the DEC Alpha microprocessor. However, with decreasing feature sizes, we find that the capacitance of a wire to its neighboring wire is an increasing fraction of the total capacitance of the wire. This will be shown analytically and empirically in section 3. As a consequence, a solution such as the DEC Alpha's solution is not likely work for smaller feature sizes.

In [5], the layout is analyzed for signals that are liable to exhibit

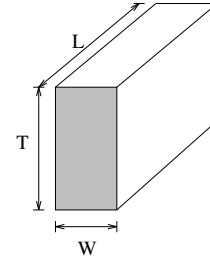


Figure 1: Resistance of a Conductor

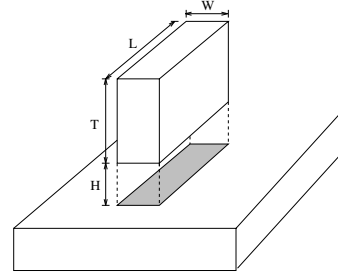


Figure 2: Plate Capacitance

cross-talk. These pairs of signals are then subjected to a logical analysis to determine if a *cross-talk fault* can be justified and propagated to the primary outputs. As cross-talk becomes a bigger problem, such post-layout methods are not expected to scale well.

An appealing solution was proposed by [6]. In this work, the authors first analyze the logic netlist for pairs of wires that are likely to have a significant cross-talk interaction, such that this interaction is visible at the primary outputs. This requires the computation of don't-cares (and for sequential circuits, image computations), both of which are computationally expensive. A specialized channel router [7] is now used to route apart these cross-talk sensitive wires. This work is appealing in that it handles cross-talk by design. However, as cross-talk becomes a significant problem in DSM IC design, it is anticipated that the number of such constraints will grow to an unacceptable level, making this approach impractical.

Other regular layout structures have been used in the past, for reasons of ease of programmability, and shorter times to market. However these structures have not been used to address DSM circuit problems.

3 Our Approach

We motivate our approach by a brief analysis on the trends of resistance and capacitance of on-chip interconnect, as feature sizes decrease.

3.1 Trends in DSM VLSI Interconnect

The resistance of the conductor in Figure 1 is given by: $R = \frac{\rho \cdot L}{A} = \frac{\rho \cdot L}{W \cdot T}$ where ρ is the resistivity of the wire material. As feature sizes decrease, the resistance of a wire increases quadratically, since both T and W scale with the minimum feature size in general. Since a quadratic increase in resistance is unacceptable, the recent trend is to increase T in relation to W .

The capacitance of a conductor consists of two parts.

- The first is called *plate capacitance*, and models the capacitance of a wire to the conductor at the lower layer.

In Figure 2, when $W \gg H$, the parallel plate model applies, where $C = k \cdot \epsilon_0 \cdot \frac{W \cdot L}{H}$. Here k is the dielectric constant of the encasing material, and ϵ_0 is the permittivity of free space. When, however, $W \leq H$, the fringing model applies and $C \propto \log(W)$. For DSM processes, the fringing model applies. At $0.6 \mu\text{m}$, $W/H \sim 2$, so the parallel plate model was applicable, but at $0.35 \mu\text{m}$ and below, $W/H \sim 1$, so the fringing model applies.

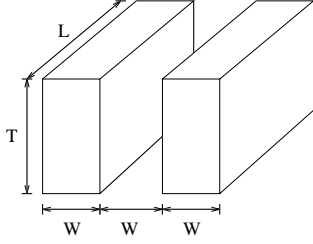


Figure 3: Edge Capacitance

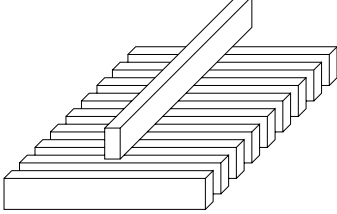


Figure 4: Single wire over a mesh

- The second is called *edge capacitance*. It models the capacitance of a wire to neighboring conductors on the same layer.

In Figure 3, when $T \gg W$, the parallel plate model applies, and $C = k \cdot \epsilon_0 \cdot \frac{T \cdot L}{W}$. For $T \leq W$, the fringing model applies, and $C \propto \log(T)$. For DSM processes, the parallel plate model applies. At $0.6 \mu\text{m}$, $T/W \sim 1$, so the fringing model was applicable. But for $0.25 \mu\text{m}$ and below, $T/W \sim 3$, so the parallel plate model applies.

So it is easy to see that the edge capacitance is becoming the dominant capacitance for a signal, and that the plate capacitance contributes less to the total capacitance. Our 3-dimensional parasitic extraction experiments verified this behavior.

From the above, we realize that increasing T/W decreases the resistance of a trace, but results in larger capacitances. It has been shown that increasing T/W beyond 2 does not cause a significant improvement in delay. Also, fabricating wires with T/W much more than 2 is difficult. Hence $T/W = 2$ is a practical choice of aspect ratio of the wires. In the sequel, we assume this value of aspect ratio for all metal layers.

3.2 Capacitance Extraction Methodology

Figures 4 and 5 graphically describe the interconnect configurations for which we performed *Space3D* runs. Figure 4 represents a wire of minimum width on metal layer i , which has a run over a series of wires on metal layer $i - 1$. This represents the extreme case where a wire is routed without neighboring wires on the same metal layer. Since there are no capacitances to neighboring wires, such a routing would typically be done for speed-critical signals.

The other extreme case is shown in Figure 5. Here a series of wires of minimum width, routed at minimum spacing on metal layer i have a run over a series of wires on metal layer $i - 1$. In this configuration, a wire would have higher capacitances to neighboring wires, and simultaneous transitions on neighboring wires could alter the delay of the wire of interest.

Table 2 reports the results of the *Space3D* runs on the configuration in 4. Table 3 shows the results of the *Space3D* runs for the

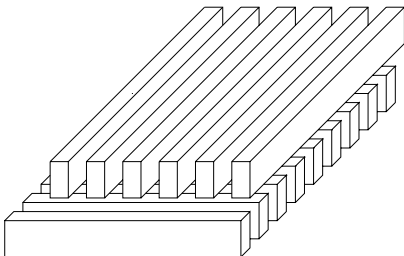


Figure 5: Mesh of wires

	t_{2f}	t_{2r}	t_{1f}	t_{1r}	t_{0f}	t_{0r}
Traditional	48.9	48.7	31.6	31.9	20.1	19.8
Our method	26.8	26.7	26.4	26.3	26.3	26.3

Table 4: Variation in delay (in ps) of inverter with environment for $0.1 \mu\text{m}$ process

configuration in 5. All *Space3D* runs were performed for all three processes under consideration, and for all metal layers. In these tables, $C_{i,0}$ refers to the capacitance of a conductor on metal layer i , to ground. $C_{i,i}$ refers to the capacitance of a conductor on metal layer i , to its immediate neighbor. $C'_{i,i}$ is the the capacitance of a conductor on metal layer i , to its neighbor's neighbor. $C_{i,i+1}$ is the capacitance of a conductor to wires of the higher layer above it. All capacitances are reported per micron of the conductor of interest, and in the units of 10^{-18} F.

We observe that for Table 2, the capacitance to ground is larger in general than for table 3. However, it is also true that for configuration of figure 5, the total capacitance of a wire is larger than for the configuration of figure 2. Hence, a signal in the configuration of 5 is more susceptible to signal integrity problems and delay variations due to the switching of its neighbors. This is because a large fraction of the wire capacitance is to its neighboring wires.

In Table 3, we notice the trend of increasing $C_{i,i}$ with decreasing feature size. Also, $C_{i,0}$ decreases in some cases. This capacitance is highly dependent on the value of t_{ins} for Metal1 in Table 1, and since this number decreases rapidly from $0.25 \mu\text{m}$ to $0.05 \mu\text{m}$, the capacitance $C_{i,0}$ actually increases in many cases with decreasing feature sizes.

In some simulations, *Space3D* did not return the values of $C'_{i,i}$, (indicated in Table 3 as an “ ϵ ” symbol). This is probably as a consequence of our setting the window-size parameter to a small value. We made this choice in order to obtain faster runs. However, whenever the value of $C'_{i,i}$ was returned, it was at least an order of magnitude less than $C_{i,i}$.

Using the parasitics determined in Table 3, we constructed a SPICE model of a $10 \times$ minimum inverter driving a Metal2 wire of length $200 \mu\text{m}$, in the $0.1 \mu\text{m}$ process. We modeled the two immediate neighbors of this wire similarly. We found the delay of the center wire to range from 19.8ps to 48.9ps, depending on whether its neighbors undergo a like or unlike transition. It is apparent that this delay range will increase with decreasing feature size since the capacitance of a wire to its neighboring wires becomes increasingly dominant with decreasing feature size.

The results of this experiment are shown in Table 4, along the row labeled “Traditional”. All delays are measured in ps, and are measured from the time the input voltage reaches $V_{dd}/2$, to the time the far end of the wire reaches $V_{dd}/2$. t_{2f} represents the delay when both neighbors switch in the opposite direction compared to the center wire which switches from high to low. t_{2r} represents the same condition, except that the center wire switches from low to high. t_{1f} and t_{1r} represent the delay when one neighbor switches in the opposite direction and the other switches in the like direction of the center wire. t_{0f} and t_{0r} represent the delay when both neighbors switch in the like direction of the center wire.

The increase in contribution of the edge capacitance to the total capacitance of a signal is a problem because adjacent wires in the same layer can in general be switching as well. These transitions on the adjacent wires can cause signal integrity problems for the wire of interest, and also significantly affect its delay. In our delay experiment (Table 4), we saw a 2.47:1 variation in a signal's delay due to its neighbors' switching activity.

3.3 Proposed Methodology

It is our opinion that DSM VLSI design using the familiar layout paradigms will not be feasible due to the problems described above. The added analysis required to design a reliable circuit using these layout paradigms will prove to be prohibitive in cost. In general, each unique interconnect topology would need to be simulated using

Process	$C_{1,0}$	$C_{2,0}$	$C_{3,0}$	$C_{4,0}$	$C_{5,0}$	$C_{6,0}$	$C_{7,0}$	$C_{8,0}$	$C_{9,0}$	Process	$C_{1,2}$	$C_{2,3}$	$C_{3,4}$	$C_{4,5}$	$C_{5,6}$	$C_{6,7}$	$C_{7,8}$	$C_{8,9}$
0.25 μ	55.47	54.02	91.80	80.61	88.21	90.13	—	—	—	0.25 μ	40.46	22.5	50.70	59.82	63.65	—	—	—
0.10 μ	53.99	58.85	102.27	67.01	95.28	84.69	92.89	89.75	—	0.10 μ	34.52	30.54	43.67	62.04	65.03	67.48	70.08	—
0.05 μ	64.32	52.48	93.39	58.52	84.28	76.71	96.08	85.44	96.47	0.05 μ	35.69	48.87	37.40	52.82	58.33	69.35	67.11	57.24

Table 2: Single Wire over mesh (10^{-18} F per μ)

Process	0.25 μ				0.10 μ				0.05 μ			
	Layer	$C_{i,0}$	$C_{i,i}$	$C'_{i,i}$	$C_{i,i+1}$	$C_{i,0}$	$C_{i,i}$	$C'_{i,i}$	$C_{i,i+1}$	$C_{i,0}$	$C_{i,i}$	$C'_{i,i}$
1	10.96	40.76	1.24	9.65	9.97	49.35	1.59	14.8	15.75	46.82	1.43	23.48
2	0.86	27.79	0.73	6.82	0.92	48.05	1.17	5.33	1.64	47.58	1.85	3.00
3	1.77	40.92	ϵ	31.81	2.58	45.84	1.46	30.14	5.38	48.28	1.11	12.45
4	0.96	41.83	ϵ	22.67	1.10	44.08	0.91	18.32	1.2	46.66	1.3	11.63
5	2.82	23.36	ϵ	41.87	2.31	39.84	ϵ	34.8	3.90	39.15	0.43	29.66
6	10.27	33.49	ϵ	—	1.07	40.59	0.55	22.04	1.51	38.33	0.15	39.2
7	—	—	—	—	3.00	23.51	0.25	38.65	3	37.61	0.355	40.27
8	—	—	—	—	10.63	30.80	2.66	—	1.57	38.95	0.14	27.15
9	—	—	—	—	—	—	—	—	13.43	31.50	1.96	—

Table 3: 3-Dimensional Parasitics for figure 5 (10^{-18} F per μ)

a 3-dimensional circuit extractor. These simulators are very compute-intensive, and hence such an approach will not be feasible.

To eliminate the uncertainty in the effective capacitance of a net and the resulting uncertainty in its delay, we introduce a new layout methodology. The primary goal of our methodology is to ensure that each net has the same immediate “electrical neighborhood”, and hence a constant effective capacitance. The other advantage of doing this is that 3-dimensional characterization needs to be done only once for the *entire* circuit. This is true regardless of the whether other signals in the signal’s neighborhood undergo any transitions.

This is achieved by alternating signal wires with power and ground wires in the layout, at a fixed pitch, on *all* metal layers. If a signal wires is denoted by S , a power wire is denoted by V and a ground wires is denoted by G , then on every metal layer, any sequence of wires will be labeled $\dots VSGSVSGSV \dots$. Also, all metal wires on a given metal layer are perpendicular to the wires on layers above and below it. Hence every signal wire has a power wire as one neighbor, and one ground wire as the other neighbor. This presence of constant-valued nodes on either side of a signal wire ensures that *each* wire on the chip has the same parasitic capacitances per unit length. Also, the majority of the total capacitance is to the ground or power wires, making the delays of a signal effectively independent of the transitions of any other signals in its neighborhood. To verify this, we ran the experiment to determine delay variation of a signal wire when its neighboring signal wires are transitioning. The results are reported in Table 4, along the “our method” row. Here we see a mere 2% variation in a signal’s delay using our scheme, as opposed to a 2.47:1 variation using the traditional approach.

In order to measure the area efficiency of our scheme, we laid out a series of static CMOS gates, using the fabric layout paradigm. In the process technology we used, the width of a Metal1 to Metal2 via plug is 1.33 times the minimum Metal1 width. In order to allow the place and route tools the flexibility to place vias at any location along a wire, we fixed the Metal1 pitch to be twice the width of a Metal1 to Metal2 via, or 2.66 times the minimum Metal1 width. For this configuration, we performed 3-dimensional parasitic extraction. The results are tabulated in Table 5.

Again, we observed that the capacitance of a signal to its nearest signal wire (i.e. it’s neighbor’s neighbor) was between one and two orders of magnitude smaller than the capacitance to its neighbor. Other observations were again similar to those made for Table 3, but magnitudes of capacitances were typically smaller.

If the wires in the design were spaced apart such that the pitch between wires was twice the minimum pitch, but there were no VDD/GND conductor between any adjacent signal wires, the cross-coupling capacitances were determined to be about $3 \times$ less than the traditional layout style (with no VDD/GND conductors between wires, which are routed at minimum pitch). This indicates that the presence of intervening VDD/GND wires in our proposed layout methodology has the dual benefit of reducing cross-coupling capacitances between signal wires, and also providing a power and ground

Process(μ)	$R_1(\Omega)$	$R_2(\Omega)$
0.25	0.17 - 0.24	5.5
0.10	0.39 - 0.54	10.63
0.05	0.68 - 0.86	20.1

Table 6: Power and Ground resistance

distribution network for the IC.

It may appear that the total capacitance of a given length of wire in our scheme would be much larger than that of a wire in the arrangement of Figure 4. From Table 2, we determine that the total capacitance of a metal2 wire per μm is 58.85×10^{-18} F (for a 0.1 μm process). Using our layout methodology, the total capacitance of a metal2 wire per μm is $(26.14 \times 2 + 9.83) \times 10^{-18} = 62.11 \times 10^{-18}$ F. Similarly close values are obtained for other metal layers. This shows that the total capacitance of a wire in our scheme is comparable to that of a wire routed for maximum speed. This is a reason why the delay obtained for our scheme in Table 4 is only slightly larger than the best delay obtained using the traditional routing methodology.

3.4 Advantages of the Proposed Methodology

There are many advantages of our approach.

- The capacitance of any signal wire is entirely predictable, and can be obtained simply by multiplying the corresponding entries in Table 5 by the wire length. This makes characterization of wire delays extremely easy. Since the capacitance of a signal wire to the nearest signal wire on the same metal layer is negligible, two successive signal wires on the same metal layer do not affect each others signal integrity or delay. In Table 4 we determined that the delays of a wire were within 2% of each other when its neighboring signals transitioned in all possible ways. This is an impressive improvement from the existing layout paradigms of today, which showed a 2.5:1 ratio of maximum to minimum delay in Table 4. Signal integrity will also similarly be improved by using our scheme, since signal integrity is a strong function of the capacitance of a signal to its neighboring signals.
- Routing of power and ground to the entire chip is automatically performed in our scheme. At every point where a power (or ground) wire on metal layer i overlaps with a power (or ground) conductor on metal layer $i - i$, a via is introduced. Given the large number of such intersections, the power and ground resistance at every point is held low, and almost constant. We extracted the resistive mesh corresponding to the power and ground networks, and ran this through SPICE to determine the effective power and ground resistance. Table 6 shows these results. We probed the power network resistance at different points in the power network, and report the maximum and minimum values of the power and ground resistance over the entire mesh under the heading R_1 . The absolute value of this resistance was extremely low, and its variation was within 50%. On the other hand, it is not easy to estimate the power and ground resistance of the existing routing methodologies, since they are extremely ad-hoc. For

Process Layer	0.25 μ				0.10 μ				0.05 μ			
	$C_{i,0}$	$C_{i,i}$	$C'_{i,i}$	$C_{i,i+1}$	$C_{i,0}$	$C_{i,i}$	$C'_{i,i}$	$C_{i,i+1}$	$C_{i,0}$	$C_{i,i}$	$C'_{i,i}$	$C_{i,i+1}$
1	17.82	17.38	ϵ	9.65	13.31	23.87	ϵ	17.11	22.67	26.01	ϵ	39.51
2	1.46	17.26	0.69	16.03	1.35	26.14	ϵ	9.83	2.05	27.91	1.71	3.84
3	8.91	21.18	ϵ	48.93	2.05	32.61	ϵ	23.57	5.08	29.73	1.23	4.0
4	1.67	19.68	ϵ	31.76	1.24	26.76	ϵ	22.94	1.03	29.20	1.03	14.35
5	0.88	11.05	ϵ	49.47	1.44	21.70	ϵ	42.73	3.35	20.84	0.29	34.89
6	17.24	17.85	ϵ	–	0.94	23.18	ϵ	27.71	2.64	20.65	1.08	32.69
7	–	–	–	–	3.44	11.39	0.33	48.42	2.62	19.26	0.22	45.07
8	–	–	–	–	10.82	19.69	ϵ	–	1.55	23.35	0.128	29.86
9	–	–	–	–	–	–	–	–	13.37	18.96	1.74	–

Table 5: 3-Dimensional Parasitics for new layout scheme (in $10^{-18}\text{F per } \mu$)

a comparison, we assume a standard cell methodology, with rows of length 1000 times minimum Metal1 width, and a width of 8 times minimum Metal1 width. The rows are powered from one end. The power or ground resistance at the end of such a row is listed in column 3, and is about 20 times larger than that obtained using our routing methodology.

- Thirdly, each signal has a current return path which is adjacent to it, thus keeping its inductance very low. As DSM circuits move to gigahertz speeds, on-chip signal inductances start affecting wire delays. This is more severe in top layer metal wires.

We use [8] to determine the inductance of a Metal8 conductor routed in our scheme versus the scheme described in Figure 4. We get a uniform inductance of $2.68 \times 10^{-4}nH/\mu$ of wire, whereas the scheme of Figure 4 gives rise to an inductance of $4.155 \times 10^{-4}nH/\mu$ of wire. These numbers were experimentally verified using [9]. This number is very dependent on the exact local layout topology, and is in general unpredictable using the existing layout paradigms of today. This is because different wires have current return paths a different distance apart. The current return path is also dependent on the input vector. This gives rise to input vector dependent delays which is very undesirable. However, in our scheme the ground plane is a small distance away from any wire, and the circuit topologies are fixed. Hence wires in our scheme have a *predictably* uniform inductance value.

- It has been empirically observed that for large chips with varying local densities of metalization, t_{ins} varies locally as well, since the changing metalization density makes it difficult to obtain a constant t_{ins} during the chemical-mechanical polish (CMP) phase of processing. This in turn causes local changes in capacitance which is undesirable.

Our scheme results in a constant density of wires in any region of the chip, and on every metal layer. This has the added advantage that it results in a much tighter control of t_{ins} , which in turn results in more predictable capacitances across the die.

- Another advantage of our scheme is that it is easy for CAD tools to make use of the regularity and predictability of parasitics to their advantage. For instance, it can be shown [10] that the regular layout structures give rise to predictable delays and a notion of “critical length” of a wire segment. A “critical length” is the maximum length of wire that can be driven unbuffered before its delay begins to increase quadratically. Wires of critical length on any layer have the same “critical delay”. More efficient CAD algorithms can be devised taking these into account.

- Further, by modifying the minimum width and spacing of each layer, the delays of different metal layers can be tuned to the designer’s specifications.

- This special layout style does not require a major modification to the existing routing tools. For our routing experiments with YACR [11], the routing was achieved by using a routing pitch which was twice the normal value. Power and ground routing can be achieved trivially after the signal wires have been routed.

- Specialized circuits like memories can be handled easily within our layout scheme. Given the regular nature of memory structures, it is not difficult to see that our scheme lends itself naturally to such structures. We created the layout for an SRAM and a DRAM cell in our layout scheme, and found that this could be done without an area penalty. Similarly, we expect that other regular structures like datapaths would also map cleanly into our layout scheme.

- Global clocking of a chip using our layout scheme would be easily achieved. We would reserve a series of grid wires for the clock signal. Since metal layers i and j run in perpendicular directions, it is easy to construct a clock H-tree structure, and to ensure equal skew at each endpoint of the clock tree.

4 Layout Experimental Results

We performed a variety of experiments to compare the area utilization characteristics of our layout scheme. Intuitively, it would appear that since signals can now only use every other routing track, there would be a big penalty of usage. It would seem that since the routing grid has twice the pitch of the traditional grid, our routes would take 4 times the area of designs routed using existing routing methodologies.

We first created a group of 14 static CMOS cells in our layout methodology, using the MAGIC [12] layout editor. We will henceforth refer to these as “fabric cells”. These cells followed the gridding conventions described in this paper, and did not use Metal2 at all. The transistor level design of these cells matched that of a control set of 14 standard cells that were part of an existing standard cell library we had access to. This standard cell library used Metal1 for internal wiring and Metal2 to contact the cell pins. However, our fabric cell library used Metal1 to contact the cell pins, and left *Metal2 free for over-the-cell routes*.

Once we had a fabric cell library and a corresponding standard cell library with 14 identical cells, we proceeded to perform placement and routing tests to compare the area utilization of the fabric cell based design to that of a standard-cell based design.

Our design flow consisted of choosing a *blif* version of a benchmark circuit, performing some simple logic optimizations on it using SIS [13], and then mapping the circuit using our library of 14 gates. After this we use the OCT [14] toolset to perform a placement and routing of this mapped design, using both the standard cell and fabric cell methods. We use the *wolfe* tool within OCT to do the placement and routing. *Wolfe* in turn calls *TimberWolfSC-4.2* [15] to do the placement, and YACR [11] to do the routing. Note that even though the fabric cell concept does not require us to use a channel-based place and route technique, we were constrained to use one. This is because even though we had access to a macro-cell placement tool, we did not have access to a reliable area router. Once the routing was complete, we compared the total area of both resulting designs.

We ran a series of example designs through both flows. In all examples, and in both the fabric cell and standard cell methodologies, we performed numerous runs and reported the best result in Table 7. The examples we used were some of the combinational circuits from the MCNC91 benchmark suite, and some additional examples as well.

Table 7 compares the result of the fabric cell based place and route with the standard cell based place and route. The areas and the corresponding number of rows are reported for both styles of layout. The “Ratio” column represents the ratio of the size of the resulting block using fabric cells to that using standard cells.

We observe that the average real area penalty while using the fabric cell concept is within 65%. This is much less than the expected $4\times$ area penalty. Also, for some designs, the fabric cell implementation has lower area than its standard cell based counterpart.

Circuit	Traditional		Our Method		Ratio
	Area ($10^6\lambda^2$)	Rows	Area ($10^6\lambda^2$)	Rows	
C432	1.25	8	1.80	5	1.44
C499	2.60	10	4.49	9	1.72
C880	2.30	10	4.86	6	2.11
C1355	3.49	13	4.60	10	1.32
C1908	2.53	13	4.87	8	1.92
C2670	5.54	13	10.67	6	1.93
C3540	8.86	19	17.80	8	2.01
alu2	1.77	9	3.711	8	2.10
apex6	6.24	13	12.8	7	2.06
count	0.64	6	1.09	6	1.69
decod	0.19	4	0.23	4	1.17
pcl	0.23	4	0.41	5	1.80
rot	6.36	13	13.29	7	2.09
pair	16.31	20	31.97	11	1.96

Table 7: Area penalty

One of the techniques being suggested [16] to minimize the delay variations due to crosstalk is to upsize transistors in the design. In [16] it was found that $W/L = 23$ for transistors was optimal in this sense. Such an increase in device sizes would certainly lead to a large increase in layout area. Our increase of 65% would be comparable, if not smaller than that the approach of [16].

There are many reasons why we believe that the fabric cell idea can be implemented with a yet lower area penalties than we observed.

- First of all, the natural router for the fabric cell concept is an area router. We believe that an area router will reduce the area of the fabric cell based designs. However, we did not have a reliable area router at our disposal. We used a channel router for our experiments since it was available in the public domain, but this kind of router is not the preferred choice for our scheme. This is because our fabric cells are designed to have a *fixed horizontal pitch* (so that two cells, when placed side by side, will obey the gridding restrictions imposed by our scheme. As a result, fabric cells have *variable heights*. When such cells are placed in a row, the channel router assumes the height of the row to be that of the tallest cell in the row. This results in a significant wastage of routable area.

- Secondly, even though our fabric cells do not utilize Metal2, the channel router approaches our cells in Metal1 and does not use the Metal2 area available over the fabric cells. In the case of the standard cells, the router approaches the cell in Metal2 and there is therefore a full utilization of the available routing resources. However, in the case of the fabric cell based layout, the Metal2 area above the cells is not utilized at all. This problem would not occur if we had an area router.

- Recent advances in combinational logic optimization [17], [18] can be used to eliminate unnecessary wires in combinational designs. Applying such optimizations to the fabric scheme would give rise to smaller layouts.

- Using a richer fabric cell library will result in better technology mapping result, and hence a more efficient design. Currently our library only had 14 cells.

- A better placement tool, which allows rotation of a cell along the horizontal axis, could improve our results.

5 Conclusions and Future Work

We have presented a layout methodology for use in DSM circuits. The salient features of this methodology are

- Exactly and simply characterized parasitics for all signals in the design.

- Power and ground routing done implicitly, and not in a separate step in the design methodology.

- Elimination of cross-talk and signal integrity problems that are common in DSM designs.

- Power and ground resistances are very low and vary much less compared to previous schemes of power and ground distribution.

- Variations in delay of a signal wire due to switching activity on its neighboring signal wires is less than 2%, compared to a 2.5:1 variation using conventional layout techniques.

- Smaller and uniform inductances for all wires on the chip, com-

pared to larger and unpredictable values using the existing layout styles.

- Total switching capacitance for a signal node comparable with the total switching capacitance of a wire routed using the configuration in Figure 4.

We believe that this technique will significantly simplify the design of chips with minimum feature sizes in the DSM range.

In the future, we plan to try out other regular structures and estimate their usefulness. We also plan to try out better place and route tools that better exploit the regularity of a design which uses our technique. We are exploring the idea of relaxing the gridding on certain metal layers, so as to allow for better logic densities. Finally, the regularity of geometry, parasitics and delays that is the core of our scheme opens up many new CAD and synthesis problems, which we plan to motivate and tackle.

6 Acknowledgements

We are grateful to the SRC, which provided financial support for this research under contract 98-DC-324.

References

- [1] "The National Technology Roadmap for Semiconductors." <http://notes.sematech.org/97melec.htm>, 1997.
- [2] P. D. Fisher, "Clock Cycle Estimation for Future Microprocessor Generations," tech. rep., SEMATECH, 1997.
- [3] "Physical Design Modelling and Verification Project (SPACE Project)." <http://cas.et.tudelft.nl/research/space/html>.
- [4] B. A. Gieseke et al., "A 600MHz Superscalar RISC Microprocessor with Out-of-Order Execution," in *Digest of Technical Papers, International Solid State Circuits Conference*, 1997.
- [5] A. Rubio, N. Itazaki, and K. Kinoshita, "An approach to the analysis and detection of cross-talk faults in digital VLSI circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 13, pp. 387–95, March 1994.
- [6] D. Kirkpatrick and A. Sangiovanni-Vincentelli, "Digital Sensitivity: Predicting signal interaction using functional analysis," in *Proceedings of the International Conference on Computer-Aided Design*, pp. 536–41, Nov 1996.
- [7] D. Kirkpatrick and A. Sangiovanni-Vincentelli, "Techniques for cross-talk avoidance in the physical design of high-performance digital systems," in *Proceedings of the International Conference on Computer-Aided Design*, pp. 616–9, Nov 1994.
- [8] S. Y. Liao, *Microwave Devices and Circuits*. Prentice-Hall, 1980.
- [9] "Analysis of Silicon Inductors and Transformers for ICs." <http://kabuki.eecs.berkeley.edu/~niknejad/doc/asitic.doc.html>.
- [10] R. K. Brayton, "Logic Synthesis for Ultra Deep Sub-Micron (UDSM)," in *Proceedings of the 35th Design Automation Conference*, 1998.
- [11] J. Reed, M. Santomauro, and A. Sangiovanni-Vincentelli, "A new gridless channel router: Yet another channel router the second (YACR-II)," in *Digest of Technical Papers International Conference on Computer-Aided Design*, 1984.
- [12] G. T. Hamachi, R. N. Mayo, and J. K. Ousterhout, "Magic: A VLSI Layout system," in *21st Design Automation Conference Proceedings*, 1984.
- [13] E. M. Sentovich, K. J. Singh, L. Lavagno, C. Moon, R. Murgai, A. Saldanha, H. Savoj, P. R. Stephan, R. K. Brayton, and A. L. Sangiovanni-Vincentelli, "SIS: A System for Sequential Circuit Synthesis," Tech. Rep. UCB/ERL M92/41, Electronics Research Laboratory, Univ. of California, Berkeley, CA 94720, May 1992.
- [14] A. Casotto, ed., *Octtools-5.1 Manuals*, (Electronics Research Laboratory, College of Engineering, University of California, Berkeley, CA 94720), University of California at Berkeley, Sept. 1991.
- [15] C. Sechen and A. Sangiovanni-Vincentelli, "The TimberWolf Placement and Routing Package," *IEEE Journal of Solid-State Circuits*, 1985.
- [16] D. Sylvester and K. Keutzer, "Getting to the bottom of deep submicron," in *Proceedings of the International Conference on Computer-Aided Design*, 1998. To Appear.
- [17] S. Yamashita, H. Sawada, and A. Nagoya, "A new method to express functional permissibilities for LUT based FPGAs and its applications," in *Proceedings of the International Conference on Computer-Aided Design*, 1996.
- [18] R. Brayton, "Understanding SPFDs: A new method for specifying flexibility," in *Workshop Notes, International Workshop on Logic Synthesis*, 1997.