C	ase 3:04-cv-03923-MMC	Document 177-1	Filed 06/13/2005	Page 1 of 45
1 2 3 4 5 6 7	GEORGE A. RILEY (S.B. # MARK E. MILLER (S.B. #1 PETER OBSTLER (S.B. #17 CHRISTOPHER D. CATA LUANN L. SIMMONS (S.F O'MELVENY & MYERS I Embarcadero Center West 275 Battery Street San Francisco, CA 94111-1 Telephone: (415) 984-87 Facsimile: (415) 984-87 Attorneys for Defendant MAGMA DESIGN AUTO	118304) – griley@omm.o 30200) – markmiller@om (1623) – pobstler@omm.o LANO (S.B. #208606) 3. #203526) – lsimmons@ LLP 33005 700 701 MATION, INC.	com nm.com com – ccatalano@omm.com	Tage Tor +3
8 9		UNITED STATES	DISTRICT COURT	
10	N	ORTHERN DISTRI	CT OF CALIFORNI	4
11		SAN FRANCIS	SCO DIVISION	
12				
13	SYNOPSYS, INC., a Delay	vare	Case No. C04-03923 N	ИМС
14	Corporation,	tiff	DECLARATION OF	CARL SECHEN IN GMA'S MOTIONS
15	v.		FOR SUMMARY J OPPOSITION TO S	UDGMENT AND IN SYNOPSYS'S
16 17	MAGMA DESIGN AUTO INC., a Delaware Corporati	MATION, on, AND	JUDGMENT	ARTIAL SUMMARY
18	LUKAS VAN GINNEKEN	, ndonta		
19	Dele	ndants.		
20				
21				
22				
23				
24				
25 26				
27				
28				
			DI	ECLARATION OF CARL SECHEN C04-3923 MMC

C	ase 3:04-cv-03923-MMC Document 177-1 Filed 06/13/2005 Page 2 of 45
1	I, Carl Sechen, declare:
2	1. I make this declaration based on my personal knowledge of the facts stated herein,
3	and, if called as a witness, I could and would testify to these facts.
4	Qualifications
6	2. I hold a Bachelors degree in Electrical Engineering from the University of
7	Minnesota, a Masters degree in Electrical Engineering from M.I.T., and a Ph.D. degree in
8	Electrical Engineering from the University of California, Berkeley. I was an Assistant Professor
9	of Electrical Engineering at Yale University from 1986 to 1990, and an Associate Professor of
10	Electrical Engineering at Yale University from 1990 to 1992. I was an Associate Professor of
11	Electrical Engineering at the University of Washington from 1992 to 1999. Since 1999 I have
12	been a Professor of Electrical Engineering at the University of Washington. I was named a Fellow
13	of the IEEE (Institute for Electrical and Electronic Engineers) in 2002.
14 15	3. I have graduated 17 Ph.D. students and am currently advising 10 Ph.D. students
16	studying in my VLSI Design and CAD Lab at the University of Washington. In my 19-year
17	career as a professor. I have written successfully funded research proposals for more than \$10
18	million
19	<i>A</i> I have authored or co-authored 144 papers one book, and two patents. I co
20	4. Thave authored of co-authored 144 papers, one book, and two patents. I co-
21	founded InternetCAD, Inc. in 1993, an integrated circuit placement and routing tool vendor.
22	5. A more detailed account of my work experience, qualifications and list of
23	publications is included in my Curriculum Vitae, which is attached as Exhibit CCC.
24	6. In the last nine years, I have been deposed as an expert witness in two cases: (1)
25	Cadence v. Avanti, US District Court (California); and (2) Trans Logic v. Hitachi, US District
26 27	Court (Oregon).
27 28	
_0	DECLARATION OF CARL SECHEN
	- C04-3923 MMC

1	7. I am being compensated for my time in this matter at my customary rate of
2	\$250/hour plus expenses. My compensation is not dependent on the outcome of this action.
3	8. I expect to testify in this case on the matters set forth in this declaration as well as
4	on other matters that may arise in this litigation including, but not limited to, matters raised in
5	expert reports submitted by the defendants in this case.
0 7	Scope of Tasks and Materials Reviewed
8	9. I have been retained by O'Melveny & Myers LLP, counsel for defendant Magma
9	Design Automation. Inc. ("Magma"), to address issues relating to certain information from
10	plaintiff Synopsys Inc. ("Synopsys") and disclosures by Magma relating to Magma's technology
11	as well as other multic disclosures
12	as well as other public disclosures.
13	10. In reaching my conclusions and preparing this report, I have reviewed or
14	considered the following documents:
15	a. March 1991 Sutherland/Sproull article entitled Speed on the Back of an Envelope
16	("Sutherland et al."), a true and correct copy of which is attached hereto as Exhibit
17	N;
18	b. November 5, 1995 Grodstein et al. article entitled A Delay Model for Logic
19	Synthesis of Continuously-Sized Networks ("Grodstein, et al."), a true and correct
20	copy of which is attached hereto as Exhibit P;
21	c Two Synopsys draft patent applications entitled System and Method for Constant
22	
23	Delay Synthesis and Method for Achieving Timing Closure of Digital Networks
24	and Method for Area Optimization of Digital Networks Under Timing Closure,
25	true and correct copies of which are attached hereto as Exhibits TT and UU;
26	d. August 1996 van Ginneken ICCAD Paper Complete Draft entitled Driving on the
27	Left Hand Side of the Performance Speed-way ("DrivingLHS"), a true and correct
28	
	- 3 - DECLARATION OF CARL SECHEN C04-3923 MMC

1		copy of which is attached hereto as Exhibit U;
2	e.	September 26, 1996 ICCAD '96 Advance Program ("ICCAD'96 brochure,
3		intended tutorial abstract"), a true and correct copy of which is attached hereto as
4		Exhibit Q;
5	f.	November 13, 1996 Otten Slides Presented at ICCAD '96 entitled Tuning for
6 7		<i>Speed</i> ("OttenSlides ICCAD '96"), a true and correct copy of which is attached
/ 8		hereto as Exhibit R.
9	g	Slides entitled Gain-based synthesis ("Magma08 (Lukas 'Gain based synthesis'
10	g.	slides)") a true and correct corry of which is attached housts of Exhibit A:
11	_	sides)), a true and correct copy of which is attached hereto as Exhibit A;
12	h.	April 6, 1998 Otten paper entitled <i>Global Wires Harmful?</i> ("OttenISPD'98
13		(Global Wires Harmful?)"), a true and correct copy of which is attached hereto as
14		Exhibit S;
15	i.	June 15, 1998 Otten et al. paper entitled Planning for Performance
16		("OttenDAC'98 (Planning for Performance)"), a true and correct copy of which is
17		attached hereto as Exhibit T;
18	j.	van Ginneken paper entitled Size Independent Synthesis ("Lukas, Kudva, Shenoy
19		'98 (Size Indep. Synthesis)"), a true and correct copy of which is attached hereto
20		as Exhibit VV;
21	k.	van Ginneken slides from April 1999 ISPD Panel entitled Synthesis Driven Lavout
22 23		("Lukas Slides of 4/22/99"), a true and correct copy of which is attached hereto as
23		Exhibit B.
25	1	April 28, 1000 Magma proce release ("Magma (proce release April 28, 1000)") a
26	L.	April 20, 1999 Magina press release (Magina (press release, April 20, 1999)), a
27		true and correct copy of which is attached hereto as Exhibit K;
28	m.	November 1999 Magma white paper entitled Overview of Magma's FixedTiming
		- 4 - DECLARATION OF CARL SECHEN C04-3923 MMC

1		Methodology ("Magma (Nov. 99 white paper)"), a true and correct copy of which
2		is attached hereto as Exhibit C;
3	n.	Groeneveld slides presented at ASP-DAC 2000 Panel entitled Timing closure
4		("Magma (ASPDAC2000Groeneveld)"), a true and correct copy of which is
5		attached hereto as Exhibit D;
6 7	0.	Groeneveld slides presented at EDP 2000 Workshop entitled <i>Design closure</i>
/ 8		("Magma (EDPapril2000Groeneveld)"), a true and correct copy of which is
9		attached hereto as Exhibit E.
10		Greeneveld slides presented at a DAC 2000 Papel antitlad Design Closure, hone
11	p.	Groeneveld sides presented at a DAC 2000 Panel entitled <i>Design Closure</i> , nope
12		or hype? ("Magma (DAC2000PanelGroeneveld)"), a true and correct copy of
13		which is attached hereto as Exhibit G;
14	q.	Groeneveld slides presented at a DAC 2000 Tutorial entitled Gain-based synthesis
15		("Groeneveld DACtutorial2000") a true and correct copy of which is attached
16		hereto as Exhibit F;
17	r.	Publicly available versions of the deposition transcript of Lukas van Ginneken
18		from April 26, 2005 and April 27, 2005.
19	11.	In addition the documents listed above, I prepared this report based upon my
20	education, exp	perience and knowledge of the industry.
21 22	Analy	sis
23	12.	Logic synthesis refers to the translation of high level descriptions of the functions
24	that an integra	ated circuit must perform into an interconnected set of logic gates. (A logic gate
25	performs a sir	nple logical function, such as comparing two signals and producing a result.)
26	Physical desig	in refers to the actual physical placement and wiring of the logic gates on a silicon
27	chin Once th	e logic gates are placed and interconnected, each gate performs its specified
28	emp. Once th	to specific sale placed and interconnected, each gate performs its specified
		- 5 - DECLARATION OF CARL SECHEN C04-3923 MMC

function and communicates the result to the next gate. The time that it takes for the gate to carry out its function and communicate the result is referred to as the delay. As the demand or "load" on a gate increases, the delay increases. Under the concept of constant delay, however, the delay for each gate is determined at the beginning of the design process and held constant throughout the remainder of the process. Increases in a gate's load imposed by changes in the design are accommodated by increasing the size of the gate to provide more electrical current so that the delay remains constant.

9 13. In preparing this declaration, I reviewed the two Synopsys patent applications and
10 the paper entitled "Driving on the Left-Hand Side of the Performance Speedway" (collectively
11 "the Synopsys Materials") to determine what information they disclose. I have found that all of
12 the information disclosed in the Synopsys Materials can be decomposed into the following 11
13 concepts and techniques.

15 14. Constant Delay. The concept of holding the delay associated with each gate
16 constant during logic synthesis and physical design.

17 15. Constant Delay Synthesis. The concept of applying constant delay to the
18 synthesis of digital circuits (including structuring and mapping).

19 16. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
20 16. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
21 16. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
22 16. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
23 24. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
24. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
25. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
26. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
27. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
28. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
29. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
20. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
21. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
22. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
23. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain
24. Constant Delay Set Via Optimal Gain. The concept of selecting the best gain

17. Buffer Insertion For Area Minimization. The concept of inserting a buffer (a
 gate that performs no logical function but boosts signal strength) for area minimization in the
 constant delay paradigm (to achieve the optimal stage effort and therefore the optimal gain for a
 stage).

27

1 18. Sizing Driven Placement. The concept of changing cell sizes during iterative 2 placement with the objective of holding the delays of each cell constant. 3 19. **Net Weight Placement.** The concept of computing a weight for each net that 4 reflects the degree to which additional load impacts overall circuit area, and applying those net 5 weights during placement. (A net refers to the wiring between the output of one gate and an input 6 of one or more other gates.) 7 20. **Continuous Gate Sizing.** The concept of employing continuous sizing of a gate 8 9 to maintain a constant delay for that gate during logic synthesis and physical design. 10 21. **Discrete Gate Sizing.** The concept of employing discrete gate sizes with the 11 objective of maintaining a constant delay for that gate during logic synthesis and physical design. 12 22. **Area Minimization.** The concept of formulating an equation that calculates the 13 area of a circuit and using that equation to minimize the area while maintaining constant delay. 14 23. **Area Estimation.** The concept of computing a weight for each net and using 15 16 those net weights to estimate circuit area in the constant delay paradigm. 17 24. **Stretching Constant Delays.** The concept of adjusting (or stretching, 18 compressing, trimming, etc.) the constant delay for gates during the logic synthesis and physical 19 design. 20 **Comparison Of Concepts To Magma Disclosures** 21 25. I next compared the concepts that I had identified in the Synopsys Materials with 22 information in Exhibits A, B, C, D, E, F, G, and K, (collectively "the Magma Materials"). I 23 24 understand that the Magma Materials were disclosed in publications and presentations made by 25 Magma. 26 26. **Constant Delay.** The following chart lists the disclosures in the Synopsys 27 Materials (left column) and in the Magma Materials (right column) of the concept of holding the 28 DECLARATION OF CARL SECHEN - 7 -C04-3923 MMC

G	ase 3:04-cv-03923-MMC Document 177	7-1 Filed 06/13/2005 Page 8 of 45
1	delay associated with each gate constant duri	ng logic synthesis and physical design
1	delay associated with each gate constant duri	ng logic synthesis and physical design.
2		
2	Synopsys Patent Application #1,	➢ Lukas Slides of 4/22/99, slide 4
3	Claim 1, Page 7:	"Freeze delays" prior to placement;
4	" a) logic synthesis of the digital	"physical design must maintain timing"
	network, using network delay as an	
5	optimization goal, where the delay of each gate is presumed constant b) Placement	➢ Groeneveld DACtutorial2000 slides
6	of the gates in a two dimensional plane. c)	29. 30. 41
0	Sizing of the gates in the network such	29: "fixed timing methodology"
7	that the network meets the cycle delay as	30: "fixed timing"
	computed by step a)."	41: a clear approach for how to set target
8		design
9	> DrivingLHS, page 1, left column and	41: "Translate delay budgets into gains"
-	right column:	
10	"In the ICCAD proceedings of 1995,	N Marrie (no. 1 4 120 1000)
11	authors from Digital Equipment	Magma (press release, April 28, 1999),
11	which they propose a synthesis method	page 1 "Magma's notant panding
12	based on a delay model for continuously	Magina's patent-pending FixedTimingTM methodology "
	sizable libraries. In the first paper [1] Joel	"system that achieves optimum timing
13	Grodstein et al. observe that if continuous	sign-off (timing closure) at the <i>beginning</i>
14	the gates can be held constant. A gate is	of the physical design process"
11	sized by multiplying all transistor widths	
15	by the same <i>gate size</i> The delay of a	
16	gate can be held constant by increasing	Magma (Nov. 99 white paper), pages 1,
10	to the capacitive load that needs to be	3, 6, 12
17	driven.	Page 1: "This backgrounder puts the
10	"In [1] Grodstein et al. propose to	technology and ideas behind Magma's
18	keep the delay constant, while expressing	Fixed I iming approach Page 1: " Fixed Timing which provides
19	the size as a function of the load.	timing sign-off at the beginning of the
	"In a companion paper [2] Eric Lehman et	physical design flow "
20	al. use the constant delay model in a	Page 3: "FixedTiming methodology
21	mapping algorithm."	freezes the delays before physical design
<u>~1</u>	"In an earlier paper [5] the constant delay	final cell sizes will be determined
22	model is applied to fanout optimization."	during a sizing driven placement fixed
a a		delays can be met based on the real wire
23	"The constant delay model is not really different than conventional models. The	load."
24	difference merely is which variable is held	Page 6: "In a nutshell, the FixedTiming
- ·	constant, when the load on a gate	methodology consists of the following
25	changes."	major steps 1 1 arget libraries are
26	"The assence of all these penets is that	The design is mapped onto
20	they reverse the causality between the	SuperCells All delays in the circuit are
27	delay and gate size. Normally, the <i>delay</i>	determined and frozen. 3 The cells are
20	of a gate is expressed as a function of its	placed and simultaneously sized to meet
28	capacitive <i>load</i> and the <i>size</i> of the gate is	

1	assumed to be a given constant	the timing based on the actual loads."
2		Page 6: "SuperCells are functional place holder cells that have a fixed delay but
3	DrivingLHS, page 2, left column: "In the new formulation, the delay	variable size. Blast Fusion automatically
4	becomes a parameter of the design, which is adjusted by an optimization algorithm	library SuperCell that has a fixed delay,
5	and its effect on gate size is observed using load analysis. Since delay is now	but variable area." Page 12: "timing is constant"
6	constant with respect to size and load, this	"Sizing Driven Placement' technology is
7	model. Note that the delay of a gate does	paradigm. It combines placement, gain
, 8	synthesis, but a change in delay is a	driven sizing, re-optimization, buffer
0	conscious' design decision."	all in one step."
9	DrivingLHS, page 3, left column:	
10	"It is interesting though that under these assumptions the delays on a path should	 Magma (DAC2000PanelGroeneveld), pages 3 5
11	be the same, apart from the intrinsic delay. It also should be noted that in the constant	3 "Fixed delay up-front, fix gate size later"
12	delay model, the <i>delay is constant by</i>	5: Magma Fixedtiming Actively managing wire delay: Through automatic
13		sizing (sizing-driven placement); Through buffer insertion"
14		
15		Magma (ASPDAC2000Groeneveld), pages 2 12 16 19 20 28
16		2: "Fix timing up-front" "True integration
17		"Integrated solution, covering the entire
18		12: "Timing is fixed, As a result, cell sizes
19		change. But large cells and small cells cancel out: some get bigger, others
20		smaller" 16: "We know the optimal size-ratio for
21		delay without knowing the <i>exact</i> values" "We can use this to fix delay
22		before the parasitics are known!" 19: "'Sizing-driven' placement" "The gain
23		ratio (=Cout/Cin) is maintained is placement: Sizes change <i>during</i> placement:
24		As a result, delay is (almost) constant" 20: "Summary FixedTiming" "Delay fixed:
27 25		Cell Area unknown; Sum of areas
25 76		28: "Magma FixedTiming sizes right" "All
20 27		capacitance they drive"
21		
28		Magma (EDP2000Groeneveld), pages
		- 9 - DECLARATION OF CARL SE

1		8, 10, 14, 15
2		8: "Magma Fixed Liming" timing + parasitics = size
3		10: "Timing is fixed, As a result, cell sizes change. But large cells and small cells
4		cancel out: some get bigger, others
		14: "'Sizing-driven' placement'' "The gain
5		ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i>
6		placement; As a result, delay is (almost) constant"
7		15: "Summary FixedTiming" "Delay fixed;
8		determines chip size. (Additive)"
9		
10		Magma98 (Lukas 'Gain based
11		2: "Sutherlands Theory of Logical Effort"
12		2: "Constant Delay and Timing Closure"4: "The variable part of the delay is the
12		same for all gates regardless of function" 6: "Constant Delay: Pick delays unfront:
13		Use a gain dependent delay model;
14		initial delay gives the best gain without
15		inserting buffers"
16	27. Constant Delay Synthesis.	The following chart lists the disclosures in the
17	Sumanaya Matariala (laft ashurun) and in the	Maama Matariala (right as huma) of the sequent of
18	Synopsys Materials (left column) and in the	Magina Materials (right column) of the concept of
19	applying constant delay to the synthesis of di	gital circuits (including structuring and mapping).
20		
21	Synopsys Patent Application #2, Claim 1 Page 20	Lukas Slides of 4/22/99, slide 6 "Layout Driven Synthesis" "Technology
21	Constant delay synthesis consisting of "a	dependent synthesis: sizing, cloning,
22	an unmapped digital network." " using	burrering, resynthesis, remapping
23	network slack as an optimization goal, where network slack is calculated assuming	 Groeneveld DACtutorial2000, slides
24	that the delay of the cells of the network is constant with respect to load " "c)	53, 54 70
25	Estimation of the area of the network based	54: overall Magma constant delay flow
26		70: Cloning and restructuring
27	Synopsys Patent Application #2, Claim	Magma (press release. April 28, 1999).
28	13, Page 22 "(globally optimal mapping)" b1) a	page 4
-		DECLARATION OF CARL SECHEN
		- 10 - C04-3923 MM

1	traversal of the network from primary	"Magma physical optimization techniques	
2	choosing at each cell the fastest matching	optimization functions that are performed	
3	using the constant delays of the books and	structuring, collapsing, re-mapping and	ļ
4	the fastest arrival times of the fanins of the matching book. b2) a traversal of the	such as SuperCell TM continuous dynamic	
5	network from right to left, while choosing at each cell the fastest matching book from	sizing and active load management."	ļ
6	the candidates selected during the previous traversal."	Magma (Nov. 99 white paper), pages 8,	
7		12 Page 8: "Logic re-mapping and	
8	Synopsys Patent Application #2, Claim 20, Page 23	restructuring at various stages during the design." "Advanced 'sizing driven'	
9	"A method for the placement and sizing of cells of a mapped digital network d)	placement tool" "FixedTiming takes care of automatic buffer insertion and cell	
10	Choosing a target delay for each cell; e) Computing the network slack using the	resizing." Page 8: "Automatic library analysis tools	
11	target delays; f) Placement of the cells of the network; g) Sizing of the cells of the	for qualifying individual library cells and building SuperCell."	
12	network such that the network meets the network slack as computed by step b)."	Page 12: " timing is constant"	
13		"Sizing Driven Placement' technology is the first to address this entirely new paradigm. It combines placement, gain	ļ
14		driven sizing, re-optimization, buffer	
15		all in one step."	
16			
17		Magma (DAC2000PanelGroeneveld), page 9	
18		"Innovative Gain-based synthesis provides early feasibility feedback"	
19		carry reasionity recuback	
20		Magma (ASPDAC2000Groeneveld), page 2	
21		"Fix timing up-front" "True integration of	ļ
22		solution, covering the entire flow"	
23			
24		Magma (EDP2000Groeneveld), page 3 "True integration of Synthesis with	
25		Place&Route"	
26		Magma98 (Lukas 'Gain based	
27		synthesis' slides), Page 16	
27 28		"Fast mapping, sizing, placement algorithms"	
20			F
		- 11 - DECLARATION OF CARL S	đ

28. Constant Delay Set Via Opt	imal Gain. The following chart lists the disclosures
in the Synopsys Materials (left column) and	in the Magma Materials (right column) of the
concept of selecting the best gain for each ga	te and using that gain to determine the constant dela
associated with that gate.	
Synopsys Patent Application #1, Claim	Lukas Slides of 4/22/99, slide 5 5. theory of logical effort (Sutherland et al.)
"(Choice of constant delay)	5: theory of logical effort (Sutherland et al.)
Claim 2, Page 7: "The method of claim 1, where the parameter C/S for each type of	Groeneveld DACtutorial2000, slide 56
gate is chosen before step 1) and is used during step a) to calculate the cycle delay,	"library analysis" shows how to determine initial intended delay
said parameter being chosen to have the largest possible value such that a long	
chain of identical gates of this type, each gate in the chain having identical value of	Magma (press release, April 28, 1999), page 7
parameter C/S, said chain cannot have simultaneously improved delay and	Page 7: "Timing sign-off is determined at the beginning of the physical design flow
improved gain by adding a buffer at some point to the same chain, even when the	using a combination of optimization techniques and logical effort abstractions."
parameter C/S is chosen optimally after adding the buffer."	Page 7: "Blast Fusion assigns cell drive strengths to every cell in the design (not
	just cells on selected critical paths) after determining the actual wire loads from the
 Synopsys Patent Application #2, Claim 3, Page 20 	layout."
"the delay of the book being chosen by choosing a load size ratio C/S for each	Magma (Nov. 99 white paper), pages 3,
book, which is independent of the network."	5, 6, 8, 14 Page 3: "FixedTiming methodology freezes
	the delays before physical design final
 Synopsys Patent Application #2, Claim 5, Page 20 	sizing driven placement fixed delays can
" a parameter C/S for each book is chosen to have the largest possible value	be met based on the real wire load." Page 3: "FixedTiming employs the
such that a long chain of identical books each cell in the chain having identical value	concepts of logical effort and gain that Magma has applied for the first time over
of parameter C/S, said chain cannot have	to a comprehensive EDA tool."
improved gain by adding a buffer at some	Page 5: "The 'electrical effort' h is called the gain of the gate: it is the ratio of the
point to the same chain, even when the parameter C/S is chosen optimally after	output and the input capacitances This
adding the buffer."	limits beforehand, and can be kept constant
	by gate sizing throughout the following

1> Synopsys Patent Application #2, Claim 20, Page 23design steps maintain that timing fixed through final layout."2"A method for the placement and sizing of closoing a target delay for each cell; e) Computing the network slack using the target delays; f) Placement of the cells of the network slack as computed by step b)."design steps maintain that timing fixed through final layout."7> Driving LHS, page 6, right column, Section 8 ("Library analysis"): "A method for the network used that the network whech statter method is the continuous buffering assumption, based on the ideas of 13." The delay of an amplifier in the technology is chosen as follows. Consider a technology is chosen as follows. Consider a fain (N Amplifiers, and a given gain requirment $H = h^{10}$ for them. For which stage h is the delay $D = N(p + gh)$ minimize the delay by choosing h: $D = \ln(H)/\ln(h)$ ($p + gh$) Now lets minimize the delay by choosing h: $D = \ln(H)/\ln(h)$ ($p + gh$) Now lets minimize the delay by choosing h: $D = \ln(H) [\left(\frac{1}{\ln(h)} - \frac{1}{n^2(h)}\right)g - \frac{1}{\ln^2(h) \ln p} = 0$ > Magma (ASPDAC2000Groeneveld), pages 16, 18, 1918"Which simplifies to $g \ln(h) - g - p/h = 0$. For a CMOS inverter, $p = aga$, which gives us $a = h \ln(h) - A$. In a barve on 8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be e=2.718 this analysis ignored the parasitic delay p of the inverters."> Magma (ASPDAC2000Groeneveld), pages 16, 18, 1922 23"Which simplifies to g ln(h) - g - p/h = 0. For a CMOS inverter, $p = aga,$ which give arasitic delay p of the inverters."> Magma (LePD2000Groeneveld), pages 16, 18, 1924 25 26 26 27 27"Wi			
20. Page 2.5"A method for the placement and sizing of clist of a mapped digial network d) Computing the network slack using the target delays, f) Placement of the cells of the network; g) Sizing of the cells of the network; g) Sizing of the cells of the network such that the network meets the design "1.4 Advanced size and the actual loads." Page 6: "In a nutshell, the fixed flag, but variable area." Page 8: "Logic re-mapping and restructing at various stages during the design." 1 Target library SuperCell that has a fixed delay. but variable area." Page 8: "Automatic buffer insertion and cell resizing." Page 8: "Automatic buffer insertion and cell resizing." Page 8: "Automatic buffer insertion and cell resizing." Page 16, 18, 1911 12 14 15 15 16 17 16 18 19 19 10 10 10 10 <th>1</th> <th>Synopsys Patent Application #2, Claim</th> <th>design steps maintain that timing fixed</th>	1	Synopsys Patent Application #2, Claim	design steps maintain that timing fixed
3Cells of a mapped digital network d) Choosing a target delay for each cell; e) Computing the network slack using the target delays; f) Placement of the cells of the 	2	20, Page 23 "A method for the placement and sizing of	through final layout." Page 6: "In a nutshell, the FixedTiming
Choosing a larger delay for each cell; e^{-1}_{0} Computing the network slack using the target delays; f) Placement of the cells of the network such that the network meets the network slack as computed by step b)." The delay for a mappifier in the technology is chosen as follows. Consider a fain of N amplifiers, and a given gain requirement $H = h^{2}$ for them. For which tage h is the delay $D = N(P + gh)$ minimal? Since $N = \ln(H)/\ln(h) (p + gh)$. Now lets minimize the delay by choosing h: $\frac{g}{D} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^{2}(h)} \right) g^{-1} \right]$ The delay of in saboul 34 to 44. The answer given by [8] is that the stage gain should be e=2.718 this analysis ignored the parasitic delay p of the inverters." Magma (ASPDAC2000Groeneveld), pages 16, 18, 19 10. 11. 12. 13. 14. 15. 15. 15. 15. 15. 16. 16. 16. 17. 16. 17. 16. 17. 17. 16. 19. 17. 16. 19. 17. 10. 17. 16. 19. 17. 10. 10. 10. 10. 10. 10. 10. 10	3	cells of a mapped digital network d)	methodology consists of the following
target delays: f) Placement of the cells of the network such that the network meets the network slack as computed by step b)." DrivingLHS, page 6, right column, Section 8 ("Library analysis"): * DrivingLHS, page 6, right column, Section 8 ("Library analysis"): * A better method is the continuous buffering assumption, based on the ideas of [3]." The delay of an amplifier in the technology is chosen as follows. Consider tain of N amplifiers, and a given gain requirement $H = h^n$ for them. For which stage h is the delay $D = N(p + gh)$ minimal? Since $N = \ln(H)/\ln(h)$ the delay is $D = \ln(H)/\ln(h) (p + gh)$. Now lets imminize the delay by choosing h: $\frac{fD}{fh} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} g - \frac$		Computing the network slack using the	major steps 1 Target libraries are
5The delay of the cells of the entwork such that the network meets the network slack as computed by step b)."SuperCells. All delays in the circuit are determined and frozen. 3 The cells are placed and simultaneously sized to meet the timing based on the actual loads." Page 6: "SuperCells are functional place holder cells that have a fixed delay but variable area."7> DrivingLHS, page 6, right column, Section 8 ("Library analysis"): "A better method is the continuous buffering assumption, based on the ideas of [3]."9"The delay of an amplifier in the technology is chosen as follows. Consider a hain 2 D = ln(H)/ln(h) (p + gh). Now lets minimal? Since $N = \ln(H)/\ln(h)$ ($p + gh$). Now lets minimize the delay by choosing h: $\frac{1}{ph} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{n^2(h)} \right) g - \frac{1}{n^2(h)} n p \right] = 0$ 18"Which simplifies to g ln(h) - g - p/h = 0. For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$, $n a better en 0.8 and 2.the gain h is about 3.4 to 4.4. The answergiven by [8] is that the stage gain should bee=2-2.18 this analysis ignored theparasitic delay p of the inverters."22"Which simplifies to g ln(h) - g - p/h = 0.For a CMOS inverter, p = ag, which givesus a = h \ln(h) - h, n a better meet 0.8 and 2.the gain h is about 3.4 to 4.4. The answergiven by [8] is that the stage gain should bee=2-2.18 this analysis ignored theparasitic delay p of the inverters."23Magma (ASPDAC2000Groeneveld), pages13, 1424Magma (EDP$	4	target delays; f) Placement of the cells of	The design is manned onto
6network slack as computed by step b)."determined and frozen. 3 The cells are placed and simultaneously sized to meet the timing based on the actual loads."7> DrivingLHS, page 6, right column, Section 8 ("Library analysis"): "A better method is the continuous buffering assumption, based on the ideas of [3]."determined and frozen. 3 The cells are placed and simultaneously sized to meet the timing based on the actual loads." Page 6: "SuperCells are functional place holder cells that have a fixed delay but variable size. Blast Fusion automatically abstracts SuperCells out of the target library SuperCell that has a fixed delay, but variable area." Page 8: "Logic re-mapping and restructuring at various stages during the design." "Advanced 'sizing driven' placement tool" "FixedTiming takes care of automatic buffer insertion and cell resizing." Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell." Page 8: "Automatic library analysis tools for values" we can use this to fix delay be care of automatic buffer insertion and cell resizing." Page 8: "Automatic library analysis tools for values" we can use this to fix delay be for a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$, for a busy by hich gives us $a = h \ln(h) - h$, for a busy by hich gives us $a = h \ln(h) - h$ for a busy by the singe in should be e=2.718 this analysis ignored the parasitic delay p of the inverters."> Magma (ASPDAC2000Groeneveld), pages 16, 18, 192216161916"We know the optimal size-ratio for delay without knowing the exact values" 'We can use this to fix delay before the parasitics are known!" 18: Sutherland (1991) logical effort 13, 14 <th>5</th> <th>the network; g) Sizing of the cells of the network such that the network meets the</th> <th>SuperCells. All delays in the circuit are</th>	5	the network; g) Sizing of the cells of the network such that the network meets the	SuperCells. All delays in the circuit are
7 > DrivingLHS, page 6, right column, Section 8 ("Library analysis"): "A better method is the continuous buffering assumption, based on the ideas of [3]."page 6: "SuperCells are functional place holder cells that have a fixed delay but variable size. Blast Fusion automatically abstracts SuperCells out of the target library SuperCell that has a fixed delay, but variable area." Page 8: "Logic re-mapping and restructuring at various stages during the design." "Advanced 'sizing driven' placement tool" "FixedTiming takes stage h is the delay $D = N(p + gh)$ minimiz? Since $N = \ln(H)/\ln(h)$ the delay is $D = \ln(H)/\ln(h) (p + gh)$. Now lets minimize the delay by choosing h: $\frac{fD}{fh} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} \ln p \right] = 0$ Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell." Page 14: " Magma's patent pending effort based delay models."10"Which simplifies to g $\ln(h) - g - p/h = 0$. For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$, for a between 0.8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be e=2.718 this analysis ignored the parasitic delay p of the inverters."> Magma (ASPDAC2000Groeneveld), pages 16, 18, 1922 23 24Magma (EDP2000Groeneveld), pages 13, 14 13: Sutherland (1991) logical effort 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change during placement; As a result, delay is (almost) constant"23 24Magma (EDP2000Groeneveld), pages 13, 1425 26 27 28Magma (EDP2000Groeneveld), pages 13, 1429 29Magma (EDP2000Groeneveld), pages 13, 14<	6	network slack as computed by step b)."	determined and frozen. 3 The cells are
7> DrivingLHS, page 6, right column, Section 89> DrivingLHS, page 6, right column, Section 89> Magma (SuperCells are functional place holder cells that have a fixed delay but variable size. Blast Fusion automatically abstracts SuperCells out of the target library SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCells out of the target library SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCells out of the target library SuperCell that has a fixed delay, but variable area." Page 8: "Logic re-mapping and restructuring at various stages during the design." Advanced 'sizing driven' placement tool" 'FixedTiming takes care of automatic buffer insertion and cell resizing." Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell. Page 8: "Logic re-mapping and restructuring at various stages during the design." 'Advanced 'sizing driven' placement tool" 'FixedTiming takes care of automatic buffer insertion and cell 	0		placed and simultaneously sized to meet
8Section 8 ("Library analysis"): "A better method is the continuous buffering assumption, based on the ideas of [3]."Indicer cells that have a fixed delay but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable size. Blast Fusion automatically abstracts SuperCell that has a fixed delay, but variable area." Page 8: "Logic re-mapping and restructing at various stages during the design." "Advanced 'sizing driven' placement tool" "FixedTiming takes care of automatic buffer insertion and cell resizing." Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell." Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell." Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell." Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell." Page 8: "Automatic library analysi	7	DrivingLHS, page 6, right column,	Page 6: "SuperCells are functional place
9(A bottary analysis): (1)9(A bottary analysis): (1)9(A bottary analysis): (1)10(1)11(1)11(1)11(1)12(1)13(1)14(1)15 $\frac{10}{M}$ 16 $\frac{10}{M}$ 17 $\frac{1}{1n(h)} - \frac{1}{n^2(h)} p = 0$ 18 $\frac{10}{M} = h(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} p \right] = 0$ 18"Which simplifies to g ln(h) - g - p/h = 0. For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$, for $a between 0.8$ and e east it delay p of the inverters."20(1)21(1)22(2)23(2)24(2)25(2)26(2)27(2)28(2)29(2)20(2)21(2)22(2)23(2)24(2)25(2)26(2)27(2)28(2)29(2)20(2)21(2)22(2)23(2)24(2)25(2)26(2)27(3)28(4)29(2)20(3)21(4)22(4)23(5)24(4)25(4)26(4)<	8	Section 8	holder cells that have a fixed delay but
buffering assumption, based on the ideas of 10 [3]." 11 "The delay of an amplifier in the 12 technology is chosen as follows. Consider a 13 chain of N amplifiers, and a given gain 14 requirement $H = h^N$ for them. For which 15 stage h is the delay $D = N(p + gh)$ 16 minimal? Since $N = \ln(H)/\ln(h)$ the delay is 17 $\frac{1D}{fh} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} \right] g - \frac{1}{\ln^2(h)} \frac{1}{h} p \right] = 0$ 18 18 19 Which simplifies to $g \ln(h) - g - p/h = 0$. 19 For a CMOS inverter, $p = ag$, which gives 10 us $a = h \ln(h) - h$, for $a between 0.8$ and 20 the gain h is about 3.4 to 4.4. The answer 21 given by [8] is that the stage gain should be 22 e parasitic delay p of the inverters." 23 24 24 25 26 26 27 28 28 29 20 20 20 20 20 21 22 23 24 25 26 26 27 28 29 20 20 20 20 20 20 20 20 20 20	0	"A better method is the continuous	variable size. Blast Fusion automatically
10[13].11"The delay of an amplifier in the technology is chosen as follows. Consider a chain of N amplifiers, and a given gain requirement $H = h^N$ for them. For which stage h is the delay $D = N(P + gh)$ minimal? Since $N = \ln(H)/\ln(h)$ the delay is $D = \ln(H)/\ln(h)$ ($p + gh$). Now lets minimize the delay by choosing h:Page 8: "Logic re-mapping and restructuring at various stages during the design." "Advanced 'sizing driven' placement tool" "FixedTiming takes care of automatic buffer insertion and cell resizing."13 $D = \ln(H)/\ln(h)$ ($p + gh$). Now lets minimize the delay by choosing h: $M_{\rm D} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} \right] g - \frac{1}{\ln^2(h)} \ln^p \right] = 0$ 16 $\frac{m}{Th} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} \ln^p \right] = 0$ Nagra (ASPDAC2000Groeneveld), pages 16, 18, 1917 $\frac{1}{\ln^2(h)} \ln p = 0$ Nagra (ASPDAC2000Groeneveld), pages 16, 18, 1918"Which simplifies to g $\ln(h) - g - p/h = 0$. For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$. for a between 0.8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be e=2.718 this analysis ignored the parasitic delay p of the inverters."22Size_analysis ignored the parasitic delay p of the inverters."23Magra (EDP2000Groeneveld), pages 13, 1424Magra (EDP2000Groeneveld), pages 13, 1425Magra (EDP2000Groeneveld), pages 13, 1426Magra (EDP2000Groeneveld), pages 13, 1427Sutherland (1991) logical effort 14: "Sizing-driven' placement; "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change during placement; <th>)</th> <th>buffering assumption, based on the ideas of</th> <th>abstracts SuperCells out of the target library SuperCell that has a fixed delay</th>)	buffering assumption, based on the ideas of	abstracts SuperCells out of the target library SuperCell that has a fixed delay
11"The delay of an amplifier in the technology is chosen as follows. Consider a chain of N amplifiers, and a given gain requirement $H = h^{N}$ for them. For which stage h is the delay $D = N(p + gh)$ minimal? Since $N = \ln(H)/\ln(h)$ the delay is is $D = \ln(H)/\ln(h)$ ($p + gh$). Now lets minimize the delay by choosing h:Page 8: "Logic re-mapping and restructuring at various stages during the design." "Advanced 'szing driven' placement tool" "FixedTiming takes care of automatic buffer insertion and cell resizing." Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell." Page 14: " Magma's patent pending effort based delay models."16 $\frac{1}{fh} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h) \ln p} \right] = 0$ >18"Which simplifies to $g \ln(h) - g - p/h = 0$. For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$. for a between 0.8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be e=2.718 this analysis ignored the parasitic delay p of the inverters.">Magma (ASPDAC2000Groeneveld), pages 16, 18, 192216*Magma (LDP2000Groeneveld), pages 13, 142324*Magma (EDP2000Groeneveld), pages 13, 142425*Magma (EDP2000Groeneveld), pages 13, 142526*Magma (EDP2000Groeneveld), pages 13, 1426*Magma (EDP2000Groeneveld), pages 13, 1427*Magma (EDP2000Groeneveld), pages 13, 1428*Magma (EDP2000Groeneveld), pages 13, 14 </th <th>10</th> <th>[5].</th> <th>but variable area."</th>	10	[5].	but variable area."
technology is chosen as follows. Consider a requirement $H = h^N$ for them. For which stage h is the delay $D = N(p + gh)$ minimal? Since $N = \ln(H)/\ln(h)$ the delay is $D = \ln(H)/\ln(h)(p + gh)$. Now lets minimize the delay by choosing h: $\frac{fD}{fh} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} \right] g - \frac{1}{\ln^2(h)} h p \right] = 0$ Which simplifies to $g \ln(h) - g - p/h = 0$. For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$. for a between 0.8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be e=2.718 this analysis ignored the parasitic delay p of the inverters." Magma (LDP2000Groeneveld), pages e=2.718 this analysis ignored the parasitic delay p of the inverters." Magma (EDP2000Groeneveld), pages 13, 14 13: Sutherland (1991) logical effort 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change during placement; $As a result, delay is (almost) constant"$	11	"The delay of an amplifier in the	Page 8: "Logic re-mapping and
12requirement $\hat{H} = h^N$ for them. For which stage h is the delay $D = N(p + gh)$ minimal? Since $N = \ln(H)/\ln(h)$ the delay is $D = \ln(H)/\ln(h) (p + gh)$. Now lets minimize the delay by choosing h :design N automatic buffer insertion and cell resizing."14 $D = \ln(H)/\ln(h) (p + gh)$. Now lets minimize the delay by choosing h :Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell."15 $\frac{\P D}{\P h} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h) \ln p} \right] = 0$ Page 14: " Magma's patent pending effort based delay models."16 $\frac{\Psi D}{\P h} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h) \ln p} \right] = 0$ Page 14: " Magma's patent pending effort based delay models."17 $\frac{1}{\ln^2(h) - h} for a between 0.8 and 2the gain h is about 3.4 to 4.4. The answergiven by [8] is that the stage gain should bee=2.718 this analysis ignored theparasitic delay p of the inverters."Magma (ASPDAC2000Groeneveld),pages 16, 18, 1922e.2.718 this analysis ignored theparasitic delay p of the inverters."Sutherland (1991) logical effort19: "Sizing-driven' placement" "The gainratio (=Cout/Cin) is maintained isplacement; Sizes change during placement;As a result, delay is (almost) constant"23Magma (EDP2000Groeneveld), pages13, 1424Magma (EDP2000Groeneveld), pages13, 1425Magma (EDP2000Groeneveld), pages13, 1426Magma (EDP2000Groeneveld), pages13, 1427Magma (EDP2000Groeneveld), pages13, 1428Magma (EDP2000Groeneveld), pages13, 1429Magma (EDP2000Groeneveld), pages13$	12	chain of N amplifiers, and a given gain	restructuring at various stages during the design " "Advanced 'sizing driven'
13 stage <i>h</i> is the delay $D = N(p + gh)$ minimal? Since $N = \ln(H)/\ln(h)$ the delay is $D = \ln(H)/\ln(h) (p + gh)$. Now lets minimize the delay by choosing <i>h</i> : 14 15 16 17 16 17 16 19 19 10 19 10 10 19 10 10 10 10 10 10 10 10 10 10	12	requirement $H = h^N$ for them. For which	placement tool" "FixedTiming takes
14 $D = \ln(H)/\ln(h) (p + gh).$ Now lets minimize the delay by choosing h:resizing."15 $\frac{fD}{fh} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} \right] g - \frac{1}{\ln^2(h)} \frac{1}{h} p \right] = 0$ Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell."16 $\frac{fD}{fh} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} \right] g - \frac{1}{\ln^2(h)} \frac{1}{h} p \right] = 0$ Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell."17 $\frac{1}{\ln^2(h)} \frac{1}{h} p \right] = 0$ $Magma (ASPDAC2000Groeneveld), pages 16, 18, 1918"Which simplifies to g \ln(h) - g - p/h = 0.For a CMOS inverter, p = ag, which givesus a = h \ln(h) - h. for a between 0.8 and 2the gain h is about 3.4 to 4.4. The answergiven by [8] is that the stage gain should bee=2.718 this analysis ignored theparasitic delay p of the inverters."Magma (ASPDAC2000Groeneveld),pages 16, 18, 192216: "We know the optimal size-ratio fordelay without knowing the exactvalues" "We can use this to fix delaybefore the parasitics are known!"18: Sutherland (1991) logical effort19: "Sizing-driven' placement, "The gainratio (=Cout/Cin) is maintained isplacement; Sizes change during placement;As a result, delay is (almost) constant"24Magma (EDP2000Groeneveld), pages13, 1425Magma (EDP2000Groeneveld), pages13, 1426Magma (EDP2000Groeneveld), pages13, 1427Magma (EDP2000Groeneveld), pages13, 1428Magma (EDP2000Groeneveld), pages13, 1429Magma (EDP2000Groeneveld), pages13, 1429Magma (EDP2000Groene$	13	stage h is the delay $D = N(p + gh)$ minimal? Since $N = \ln(H)/\ln(h)$ the delay is	care of automatic buffer insertion and cell
minimize the delay by choosing h: $\frac{fD}{fh} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} \right] g - \frac{1}{\ln^2(h)} \frac{1}{h} p \right] = 0$ $\frac{1}{\ln^2(h)} \frac{1}{h} p = 0$ "Which simplifies to $g \ln(h) - g - p/h = 0$. For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$. for a between 0.8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be e = 2.718 this analysis ignored the parasitic delay p of the inverters." Magma (ASPDAC2000Groeneveld), pages 16, 18, 19 16: "We know the optimal size-ratio for delay without knowing the <i>exact</i> values" "We can use this to fix delay before the parasitics are known!" 18: Sutherland (1991) logical effort 19: "Sizing-driven' placement; "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; As a result, delay is (almost) constant" Magma (EDP2000Groeneveld), pages 13, 14 13: Sutherland (1991) logical effort 14: "Sizing-driven' placement; "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; Magma (EDP2000Groeneveld), pages 13, 14 13: Sutherland (1991) logical effort 14: "Sizing-driven' placement; "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; Magma (EDP2000Groeneveld), pages 13, 14 13: Sutherland (1991) logical effort 14: "Sizing-driven' placement; "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement;	14	$D = \ln(H)/\ln(h) (p + gh)$. Now lets	resizing."
$\frac{\eta D}{fh} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} \right] g - \frac{1}{\ln^2(h)} \frac{1}{h} p \right] = 0$ $\frac{1}{\ln^2(h)} \frac{1}{h} p = 0$ Which simplifies to $g \ln(h) - g - p/h = 0$. For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$. for a between 0.8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be e = 2.718 this analysis ignored the parasitic delay p of the inverters." $\frac{1}{23}$ $\frac{1}{24}$ $\frac{1}{25}$ $\frac{1}{26}$ $\frac{1}{27}$ $\frac{1}{28}$ $\frac{1}{$	15	minimize the delay by choosing h :	for qualifying individual library cells and
16 $[m]$ $[$	15	$\left\ \frac{\eta D}{\P h} = \ln(H) \right\ \left\ \frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right\ g - \frac{1}{\ln^2(h)} \ g\ = \frac{1}{\ln^$	building SuperCell."
17 $\frac{1}{\ln^2(h)} \frac{1}{h} p = 0$ end of the based delay models.18"Which simplifies to $g \ln(h) - g - p/h = 0$. For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$. for a between 0.8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be $e=2.718$ this analysis ignored the parasitic delay p of the inverters."> Magma (ASPDAC2000Groeneveld), pages 16, 18, 192016: "We know the optimal size-ratio for delay without knowing the <i>exact</i> values" "We can use this to fix delay before the parasitics are known!" 18: Sutherland (1991) logical effort 19: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; As a result, delay is (almost) constant"21Magma (EDP2000Groeneveld), pages 13, 1422Magma (EDP2000Groeneveld), pages 13, 1423Magma (EDP2000Groeneveld), pages 13, 1424Sutherland (1991) logical effort 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; Magma (tipped) logical effort 13: Sutherland (1991) logical effort 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement;	16		Page 14: " Magma's patent pending
$ \begin{array}{c c c c c c c c c c c c c c c c c c c $	17	$\frac{1}{1-2}\left \frac{1}{p}\right = 0$	enon based delay models.
"Which simplifies to $g \ln(h) - g - p/h = 0$. For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$. for a between 0.8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be $e=2.718$ this analysis ignored the parasitic delay p of the inverters."Magma (ASPDAC2000Groeneveld), pages 16, 18, 192216: "We know the optimal size-ratio for delay without knowing the <i>exact</i> values" "We can use this to fix delay before the parasitics are known!" 18: Sutherland (1991) logical effort 19: "Sizing-driven' placement; Magma (EDP2000Groeneveld), pages 13, 1423Magma (EDP2000Groeneveld), pages 13, 1424Magma (EDP2000Groeneveld), pages 13, 1425Sutherland (1991) logical effort 14: "Sizing-driven' placement; "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; Magma (EDP2000Groeneveld), pages 13, 1426Magma (EDP2000Groeneveld), pages 13, 1427Magma (EDP2000Groeneveld), pages 13, 1428Magma (EDP2000Groeneveld), pages 13, 1429Magma (EDP2000Groeneveld), pages 13, 1420Magma (EDP2000Groeneveld), pages 13, 1421Magma (EDP2000Groeneveld), pages 13, 1422Magma (EDP2000Groeneveld), pages 13, 1423Magma (EDP2000Groeneveld), pages 13, 1424Magma (EDP2000Groeneveld), pages 13, 1425Magma (EDP2000Groeneveld), pages 13, 1426Magma (EDP2000Groeneveld), pages 13, 1427Magma (EDP2000Groeneveld), pages 13, 1428Magma (EDP2000Groeneveld), pages 14 <th>18</th> <th>$\ln^2(h)$ n \int</th> <th></th>	18	$\ln^2(h)$ n \int	
For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$. for a between 0.8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be e=2.718 this analysis ignored the parasitic delay p of the inverters."	10	"Which simplifies to $g \ln(h) - g - p/h = 0$.	 Magma (ASPDAC2000Groeneveld), magazi 16, 18, 10
20 Is u = n m(n) = n. for u between 0.6 and 2 21 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be e=2.718 this analysis ignored the parasitic delay p of the inverters." delay without knowing the exact values" "We can use this to fix delay before the parasitics are known!" 22 23 before the parasitics are known!" 23 23 18: Sutherland (1991) logical effort 19: "Sizing-driven' placement? "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change during placement; As a result, delay is (almost) constant" 24 > Magma (EDP2000Groeneveld), pages 13, 14 25 > Magma (EDP2000Groeneveld), pages 13, 14 26 Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change during placement; 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change during placement; 14: "Sizes change during placement; 14: 14: 14: 14: 14: 14: 14: 14: 14: 14:	19	For a CMOS inverter, $p = ag$, which gives us $a = h \ln(h) - h$ for a between 0.8 and 2	16: "We know the optimal size-ratio for
 given by [8] is that the stage gain should be e=2.718 this analysis ignored the parasitic delay <i>p</i> of the inverters." values" "We can use this to fix delay before the parasitics are known!" 18: Sutherland (1991) logical effort 19: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; As a result, delay is (almost) constant" Magma (EDP2000Groeneveld), pages 13, 14 Sutherland (1991) logical effort 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placemen	20	the gain h is about 3.4 to 4.4. The answer	delay without knowing the <i>exact</i>
 22 parasitic delay p of the inverters." 23 24 25 26 27 28 	21	given by [8] is that the stage gain should be $e=2.718$ this analysis ignored the	values" "We can use this to fix delay
 19: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; As a result, delay is (almost) constant" Magma (EDP2000Groeneveld), pages 13, 14 Sutherland (1991) logical effort 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; 	22	parasitic delay p of the inverters."	18: Sutherland (1991) logical effort
 ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; As a result, delay is (almost) constant" Magma (EDP2000Groeneveld), pages 13, 14 Sutherland (1991) logical effort 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; 	22		19: "Sizing-driven' placement" "The gain
As a result, delay is (almost) constant" As a result, delay is (almost) constant" Magma (EDP2000Groeneveld), pages 13, 14 13: Sutherland (1991) logical effort 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement;	23		placement; Sizes change <i>during</i> placement:
 25 26 26 27 28 28 25 25 26 27 28 28 28 29 29 20 20 20 21 22 23 24 25 25 26 27 28 28 28 28 28 29 29 20 20 20 21 22 23 24 25 25 26 27 28 28 28 29 29 20 20 20 21 22 23 24 25 25 26 27 28 28 29 20 20 20 20 20 20 20 21 22 23 24 25 26 27 27 28 28 28 29 20 21 22 23 24 25 26 27 27 28 28 29 20 <	24		As a result, delay is (almost) constant"
 Magma (EDP2000Groeneveld), pages 13, 14 Sutherland (1991) logical effort 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; 	25		
 26 27 28 28 13, 14 13: Sutherland (1991) logical effort 14: "Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement; 	23		Magma (EDP2000Groeneveld), pages
 27 28 28 28 28 28 27 28 28 28 28 28 28 28 28 29 29 20 20 21 21 22 23 24 25 26 27 27 28 29 29 20 20 21 21 22 23 24 25 26 27 27 28 29 29 20 20 21 21 22 23 24 25 26 27 27 27 28 29 20 21 21 22 23 24 25 26 27 27 27 28 29 20 21 21 22 23 24 25 25 26 27 27 27 28 29 20 21 21 22 23 24 25 26 27 27 27 27 27 27 27 28 29 20 21 21 21 22 23 24 24 25 26 27 27 26 27 <	26		13, 14 13: Sutherland (1991) logical effort
28 ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement;	27		14: "'Sizing-driven' placement" "The gain
	28		ratio (=Cout/Cin) is maintained is placement: Sizes change <i>during</i> placement:
	-0		

	As a result, delay is (almost) constant"
	 Magma98 (Lukas 'Gain based synthesis' slides), Pages 6, 12 6: "Constant Delay: Pick delays upfront; Use a gain dependent delay model; Compare all gates to the inverter; The initial delay gives the best gain without inserting buffers" 12: "Methodology: Pick delays to meet delay constraints; Optimize gain; Check that system gain is OK; Meet the delays by significant."
	sizing"
29. Buffer Insertion For Area M	Minimization. The following chart lists the
isclosures in the Synopsys Materials (left c	olumn) and in the Magma Materials (right col
a concept of inserting a huffer (a gate that	performs no logical function but boosts signal
le concept of inserting a burier (a gate that	performs no logical function out boosts signal
rength) for area minimization in the consta	nt delay paradigm (to achieve the optimal stag
nd therefore the optimal gain for a stage).	
 Synopsys Patent Application #1, Claim 6, Page 9 the buffers being inserted on non- ritical paths, as determined by subtracting ne delay of the buffer from the slack of the 	 Lukas Slides of 4/22/99, slide 6 6: "Layout Driven Synthesis" "Technology dependent synthesis: sizing, cloning, buffering, resynthesis, remapping"
ath, while estimating the area reduction	
ath, while estimating the area reduction ."	Groeneveld DACtutorial2000, slide 60 "Buffering, cloning and restructuring are used to maintain delay during placement"
 ath, while estimating the area reduction ." Synopsys Patent Application #2, Claims 6-7, Page 21 	Groeneveld DACtutorial2000, slide 60 "Buffering, cloning and restructuring are used to maintain delay during placement"
 ath, while estimating the area reduction Synopsys Patent Application #2, Claims 6-7, Page 21 buffer insertion, the buffers being userted on paths with positive slack. as 	 Groeneveld DACtutorial2000, slide 60 "Buffering, cloning and restructuring are used to maintain delay during placement" Magma (press release, April 28, 1999), page 9
 ath, while estimating the area reduction Synopsys Patent Application #2, Claims 6-7, Page 21 buffer insertion, the buffers being iserted on paths with positive slack, as etermined by subtracting the delay of the uffer from the slack of the path " 	 Groeneveld DACtutorial2000, slide 60 "Buffering, cloning and restructuring are used to maintain delay during placement" Magma (press release, April 28, 1999), page 9 "FixedTiming methodology solves the problem because it delays fundamental
 ath, while estimating the area reduction Synopsys Patent Application #2, Claims 6-7, Page 21 buffer insertion, the buffers being iserted on paths with positive slack, as etermined by subtracting the delay of the uffer from the slack of the path." the buffer is inserted if area is saved." 	 Groeneveld DACtutorial2000, slide 60 "Buffering, cloning and restructuring are used to maintain delay during placement" Magma (press release, April 28, 1999), page 9 "FixedTiming methodology solves the problem because it delays fundamental decisions such as cell sizing (for all cells in the design) and wire loading (width.
 ath, while estimating the area reduction Synopsys Patent Application #2, Claims 6-7, Page 21 buffer insertion, the buffers being iserted on paths with positive slack, as etermined by subtracting the delay of the uffer from the slack of the path." the buffer is inserted if area is saved." 	 Groeneveld DACtutorial2000, slide 60 "Buffering, cloning and restructuring are used to maintain delay during placement" Magma (press release, April 28, 1999), page 9 "FixedTiming methodology solves the problem because it delays fundamental decisions such as cell sizing (for all cells in the design) and wire loading (width, spacing, and buffering) until very late in the flow."
 ath, while estimating the area reduction Synopsys Patent Application #2, Claims 6-7, Page 21 buffer insertion, the buffers being iserted on paths with positive slack, as etermined by subtracting the delay of the uffer from the slack of the path." the buffer is inserted if area is saved." DrivingLHS, page 5, left column, section 6 	 Groeneveld DACtutorial2000, slide 60 "Buffering, cloning and restructuring are used to maintain delay during placement" Magma (press release, April 28, 1999), page 9 "FixedTiming methodology solves the problem because it delays fundamental decisions such as cell sizing (for all cells in the design) and wire loading (width, spacing, and buffering) until very late in the flow."
 ath, while estimating the area reduction Synopsys Patent Application #2, Claims 6-7, Page 21 buffer insertion, the buffers being aserted on paths with positive slack, as etermined by subtracting the delay of the uffer from the slack of the path DrivingLHS, page 5, left column, section 6 betermining whether or not a buffer should e added: 	 Groeneveld DACtutorial2000, slide 60 "Buffering, cloning and restructuring are used to maintain delay during placement" Magma (press release, April 28, 1999), page 9 "FixedTiming methodology solves the problem because it delays fundamental decisions such as cell sizing (for all cells in the design) and wire loading (width, spacing, and buffering) until very late in the flow." Magma (Nov. 99 white paper), pages 8, 12

		-1 Filed 00/13/2003 Fage 15 01 45
	gate g. Let c' be the updated net loads, then	design." "Advanced 'sizing driven'
	$\Delta A = a^{1} (c - c^{2})$. The area of the buffer is $A_{b} - a_{b}c_{b}$, which is added to the circuit. The	placement tool" "FixedTiming takes
	buffer saves area if $a^1 (c - c') > a_b c_b$."	resizing."
		Page 8: "Automatic library analysis tools for qualifying individual library cells and
		building SuperCell." Page 12: " timing is constant"
		"Sizing Driven Placement' technology is the first to address this entirely new
		paradigm. It combines placement, gain driven sizing, re-optimization, buffer
		insertion and congestion avoidance routing
		all in one step.
		➢ Magma (DAC2000PanelGroeneveld),
		page 5 "Magma Fixedtiming" "Actively
		managing wire delay: Through automatic
		buffer insertion"
		Magma (ASPDAC2000Groeneveld), page 22
		22: handling discrete libraries, cloning,
		bulleting
		➢ Magma98 (Lukas 'Gain based
		synthesis' slides), Page 13 "Inserting buffers always costs delay:
		Inserting buffers saves area"
	30. Sizing Driven Placement. The	he following chart lists the disclosures in the
	Synopsys Materials (left column) and in the l	Magma Materials (right column) of the concept of
	changing cell sizes during iterative placemen	t with the objective of holding the delays of each
	changing cen sizes during herative placemen	t with the objective of holding the delays of each
(constant.	
	Synoneye Patent Application #1	Crospaveld DA Ctutorial 2000 slides
	Claims 7, 8; Pages 9, 10	43, 49, 60
	Claim 7: " where step b) [placement] is performed by repeated partitioning steps,	43: "The gain ratio is maintained during placement"
	partitioning the gates in the network into two or more groups, each group being	49: "But large cells and small cells cancel out: some get bigger, others smaller"
	assigned to an subdivision of the plane, alternating the partitioning steps with	60: "Buffering, cloning and restructuring are used to maintain delay during
		- 15 - DECLARATION OF CARL SECH

1	sizing steps"	placement"
2 3	Claim 8: iterative placement improvement: " repeatedly changing the location of one or two gates at a time, while performing a	 Magma (press release, April 28, 1999), page 5
4	sizing step c) after each location change."	Page 5: "The physical engines included in Blast Fusion include a force-directed
5	 Synopsys Patent Application #2, Claims 20-23, Page 23 	placer, detailed placer, global router, track router, and patent-pending "virtual gridless" detailed router "
6	Claim 20: "A method for the placement	gridless detailed fouter.
7	network d) Choosing a target delay for	Magma (Nov. 99 white paper), pages 3,
8	using the target delays; f) Placement of the	6, 8, 12, 15 Page 3: "FixedTiming methodology freezes
9	of the network such that the network meets the network slack as computed by step b) "	the delays before physical design final cell sizes will be determined during a
10	Claim 21: Executing step f) in Claim 20	sizing driven placement fixed delays can
11	steps, each step being followed by a sizing	Page 3: "FixedTiming employs the
12	Claim 22: Placement by partitioning, a	Magma has applied for the first time ever
13	iteration.	to a comprehensive EDA tool." Page 6: "In a nutshell, the FixedTiming
14	Claim 23: Placement by iterative improvement (moving one or two cells at a	methodology consists of the following major steps 1 Target libraries are
15	time), a sizing step being executed after each iteration.	read, as well as timing constraints set 2 The design is mapped onto
16		determined and frozen. 3 The cells are
17		the timing based on the actual loads."
18		holder cells that have a fixed delay but
19		variable size. Blast Fusion automatically abstracts SuperCells out of the target
20		library SuperCell that has a fixed delay,
21		but variable area."
22		restructuring at various stages during the
		design." "Advanced 'sizing driven'
23		placement tool" "Fixed liming takes care of automatic buffer insertion and cell
24		resizing."
25		Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell."
26		Page 12: " timing is constant"
27		"Sizing Driven Placement' technology is the first to address this entirely new
28		paradigm. It combines placement, gain

1		driven sizing, re-optimization, buffer
2		all in one step."
3		Page 15: "Magma's FixedTiming
4		problem by establishing timing as an
-		absolute constraint." "Blast Fusion assigns cell drive strengths to every cell in the
3		design (not just on selected critical paths)
6		from the layout."
7		
8		Magma (DAC2000PanelGroeneveld), pages 4, 5
9		4: "Actively managing wire delay: Through
10		automatic sizing (sizing-driven placement)"
11		5: "Magma Fixedtiming" "Actively
10		sizing (sizing-driven placement); Through
12		buffer insertion"
13		Magma (ASPDAC2000Groeneveld).
14		page 19
15		ratio (=Cout/Cin) is maintained is
16		placement; Sizes change <i>during</i> placement; As a result, delay is (almost)
17		constant"
18		
19		Magma (EDP2000Groeneveld), page
20		"Sizing-driven' placement" "The gain ratio (=Cout/Cin) is maintained is
20		placement; Sizes change <i>during</i> placement;
21		As a result, delay is (annost) constant
22		Magma98 (Lukas 'Gain based
23		synthesis' slides), Page 11 "Use realistic timing constraints: They
24		must be met before placement; Need sizing
25		
26	31. Net Weight Placement. The	following chart lists the disclosures in the Synopsys
27	Materials (left column) and in the Magma Ma	aterials (right column) of the concept of computing a
28	weight for each net that reflects the degree to	which additional load impacts overall circuit area,
		- 17 - DECLARATION OF CARL SECHEN C04-3923 MMC

Cá	ase 3:04-cv-03923-MMC Document 177	-1 Filed 06/13/2005 Page 18 of 45
1	and applying those net weights during placer	nent.
2		
3	 Synopsys Patent Application #1, Claims 9,10, Page 10 	 Magma98 (Lukas 'Gain based synthesis' slides), Pages 9, 10
4	the weight being proportional to the area divided by the typical load "	9: Circuit area $A = a^{2}c$ is given as the usual equation for total area (<i>a</i> is the area sensitivity of a gate)
5	divided by the typical load.	9: Gives formula for calculating net
6	Synopsys Patent Application #2, Claim	inputs to outputs"
7	30, Page 24 ("Weighted Placement") f1) The	$A = a^{T} c = \sum a_{i} c_{i}$
8	calculation of net weights that reflect the	$w_i = \sum_{i=1}^{\infty} \frac{w_j}{i} + a_i$
9	change of network area due to sizing as a function of net length; f2) Placement of the cells of the network where the weighted	j∈ fanin ^h ij
10	network net length is used as a placement	$\mathbf{w} = \mathbf{M}\mathbf{w} + \mathbf{a}$ 9: $\mathbf{A} = \mathbf{a}^{\mathrm{T}}\mathbf{c} = \mathbf{w}^{\mathrm{T}}\mathbf{L}$, where L is the imposed
11	being the sum of the weighted net lengths	capacitance, w is the weight vector 10° "Placement formulation: $A - wI$ · The
12	of all nets, each net length being multiplied by a weight."	active area of the circuit can be expressed
13		10: "Note: Placement programs naturally
14		like to optimize a weighted sum of wire lengths"
15		
16	32. Continuous Gate Sizing. Th	e following chart lists the disclosures in the
17	Synopsys Materials (left column) and in the	Magma Materials (right column) of the concept or
18	employing continuous sizing of a gate to main	intain a constant delay for that gate during logic
19	synthesis and physical design.	
20		
21	Synopsys Patent Application #1, Claim	Lukas Slides of 4/22/99, slides 11
22	3, Page 8 "The method of claim 2, where step c) is	sizing must be unrestricted
23	comprised of steps: d) Determining the parameter C/S for each gate according to delay and gain constraints on the network	Magma (press release, April 28, 1999),
24	e) Determining the size S for each gate by	Page 7: "Timing sign-off is determined at
25	traversing the network, in the direction opposite to the direction of data flow, while	the beginning of the physical design flow using a combination of optimization
26	calculating the load of each gate, the size S of said gate being determined by the	techniques and logical effort abstractions."
27	parameter C/S and the load C of said gate, the load on the preceding gate being	strengths to every cell in the design (not
28	determined by the size of the fanout gates, iterating step e) until convergence."	just cells on selected critical paths) after
		- 18 - DECLARATION OF CARL SECH C04-3923 M
ļ		

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28	 Synopsys Patent Application #2, Claim 15, Page 22 "(sizing algorithm) starting at the primary outputs and traversing the network in the direction opposite to the data flow, while calculating the load of a cell, by summing over the fanout of the cell, the product of the load of the fanout cell divided by the gain plus the net load of the cell." Synopsys Patent Application #2, Claim 20, Page 23 "A method for the placement and sizing of cells of a mapped digital network d) Choosing a target delay for each cell; e) Computing the network slack using the target delays; f) Placement of the cells of the network; g) Sizing of the cells of the network such that the network meets the network slack as computed by step b)." Synopsys Patent Application #2, Claim 24: "(sizing) g1) calculation of the net length based on the available placement information; g2) calculation of the capacitive load." Claim 25: "(sizing algorithm) starting at the primary outputs and traversing the network in the direction opposite to the data flow, while calculating the load a cell, by summing over the fanout cell divided by the gain plus the net load of the cell." 	 layout." Page 9: "FixedTiming methodology solves the problem because it delays fundamental decisions such as cell sizing (for all cells in the design) and wire loading (width, spacing, and buffering) until very late in the flow." Magma (Nov. 99 white paper), pages 3, 4, 6, 8 Page 3: "FixedTiming methodology freezes the delays before physical design final cell sizes will be determined during a sizing driven placement fixed delays can be met based on the real wire load." Page 3: "FixedTiming employs the concepts of logical effort and gain that Magma has applied for the first time ever to a comprehensive EDA tool." Page 4: "Instead of a fluctuating delay, Magma's FixedTiming methodology changing cell sizes." Page 6: "In a nutshell, the FixedTiming methodology consists of the following major steps 1 Target libraries are read, as well as timing constraints set 2 The design is mapped onto SuperCells. All delays in the circuit are determined and frozen. 3 The cells are placed and simultaneously sized to meet the timing based on the actual loads." Page 6: "SuperCells are functional place holder cells that have a fixed delay but variable size. Blast Fusion automatically abstracts SuperCells out of the target library SuperCell that has a fixed delay, but variable area." Page 8: "Logic re-mapping and restructuring at various stages during the design." "Advanced 'sizing driven' placement tool" "FixedTiming takes care of automatic buffer insertion and cell resizing."
1 2	 Synopsys Patent Application #2, Claim 15 Page 22 	determining the actual wire loads from the layout." Page 9: "FixedTiming methodology solves
3 4	"(sizing algorithm) starting at the primary outputs and traversing the network in the direction opposite to the data flow, while calculating the load of a cell, by	the problem because it delays fundamental decisions such as cell sizing (for all cells in the design) and wire loading (width, spacing, and buffering) until very late in
5 6 7	summing over the fanout of the cell, the product of the load of the fanout cell divided by the gain plus the net load of the cell."	 the flow." Magma (Nov. 99 white paper), pages 3, 4, 6, 8
8 9 10	 Synopsys Patent Application #2, Claim 20, Page 23 "A method for the placement and sizing of cells of a mapped digital network d) 	Page 3: "FixedTiming methodology freeze the delays before physical design final cell sizes will be determined during a sizing driven placement fixed delays ca
11 12	Choosing a target delay for each cell; e) Computing the network slack using the target delays; f) Placement of the cells of the network; g) Sizing of the cells of the network such that the network meets the	Page 3: "FixedTiming employs the concepts of logical effort and gain that Magma has applied for the first time ever
13 14	 Nork such that the network meets the network slack as computed by step b)." Synopsys Patent Application #2, 	to a comprehensive EDA tool." Page 4: "Instead of a fluctuating delay, Magma's FixedTiming methodology changing cell sizes."
15 16 17	Claims 24-25, Page 23-24 Claim 24: "(sizing) g1) calculation of the net length based on the available placement information; g2) calculation of the capacitive load of the cells using the net	Page 6: "In a nutshell, the Fixed liming methodology consists of the following major steps 1 Target libraries are read, as well as timing constraints set 2 The design is mapped onto
18 19	length; g3) calculation of the sizes from the capacitive load." Claim 25: "(sizing algorithm) starting at the primary outputs and traversing the	SuperCells. All delays in the circuit are determined and frozen. 3 The cells are placed and simultaneously sized to meet the timing based on the actual loads."
20 21	network in the direction opposite to the data flow, while calculating the load a cell, by summing over the fanout of the cell, the product of the load of the fanout cell	holder cells that have a fixed delay but variable size. Blast Fusion automatically abstracts SuperCells out of the target
22 23	divided by the gain plus the net load of the cell."	library SuperCell that has a fixed delay, but variable area." Page 8: "Logic re-mapping and restructuring at various stages during the
24 25		design." "Advanced 'sizing driven' placement tool" "FixedTiming takes care of automatic buffer insertion and cell
26 27		resizing." Page 8: "Automatic library analysis tools for qualifying individual library cells and building SuperCell."
28		

1	Magma (DAC2000PanelGroeneveld),
2	pages 4, 5 4: "Actively managing wire delay: Through
3	automatic sizing (sizing-driven
4	placement)" 5: "Magma Fixedtiming" "Actively
5	managing wire delay: Through automatic
6	sizing (sizing-driven placement); Through buffer insertion"
7	
8	Magma (ASPDAC2000Groeneveld), pages 11, 12, 19, 20, 28
9	11: timing + parasitics = size
10	change. But large cells and small cells
11	cancel out: some get bigger, others smaller"
12	19: "'Sizing-driven' placement'" "The gain
13	ratio (=Cout/Cin) is maintained is placement; Sizes change <i>during</i> placement;
14	As a result, delay is (almost) constant"
15	Cell Area unknown; Sum of areas
16	determines chip size. (Additive)" 28: "Magma FixedTiming sizes right" "All
17	cells have exactly the right strength for the
18	capacitance mey unve
10	Magma (EDP2000Groeneveld), pages
20	8, 10, 14, 15 8: "Magma FixedTiming" timing
20	parasitics = size
21	10: "Timing is fixed, As a result, cell sizes change. But large cells and small cells
22	cancel out: some get bigger, others
23	14: "Sizing-driven' placement" "The gain
24	ratio (=Cout/Cin) is maintained is
25	As a result, delay is (almost) constant"
26	15: "Summary FixedTiming" "Delay fixed; Cell Area unknown; Sum of areas
27	determines chip size. (Additive)"
28	

	 Magma98 (Lukas 'Gain based synthesis' slides), Pages 12, 15, 16 12: "Methodology: Pick delays to meet delay constraints; Optimize gain; Check that system gain is OK; Meet the delays by sizing" 15: "Post placement: Pick exact discrete sizes" 16: "Fast mapping, sizing, placement algorithms"
33. Discrete Gate Sizing. The f	collowing chart lists the disclosures in the Syr
Materials (left column) and in the Magma M	Materials (right column) of the concept of emp
liscrete gate sizes with the objective of mai	ntaining a constant delay for that gate during
ynthesis and physical design.	
DrivingLHS, page 5, right column, Section 7	Lukas Slides of 4/22/99, slide 10 parallel sizing (for more drive strength)
"Discrete sizing algorithms"): By using discrete gates in parallel, integer	
nultiples of the gates can be created a luplicated gate can drive exactly twice the	Groeneveld DACtutorial2000, slides 57-59, 64, 67
bad at exactly the same delay. Different izes can be mixed to create non-integer	57-59: library discretization issues
nultiples."	are used to maintain delay during
	61: optimum wire buffering
	67: What makes a good DSM library (few
	inputs, lots of drives, no multi-stage gates)
	 Magma (ASPDAC2000Groeneveld),
	page 22 22: handling discrete libraries, cloning,
	buffering
	Magma08 (Lykes (Coin based
	synthesis' slides), Page 15
	15: "Post placement: Pick exact discrete sizes"

1	34. Area Minimization. The foll	lowing chart lists the disclosures in the Synopsys
2	Materials (left column) and in the Magma M	aterials (right column) of the concept of formulating
3	an equation that calculates the area of a circu	it and using that equation to minimize the area while
4 5	maintaining constant delay.	
6	Synopsys Patent Application #1, Claim	➢ Magma98 (Lukas 'Gain based
7	5, Pages 8-9 "(Area estimation)"	synthesis' slides), Pages 8, 9 8: Contains the usual equation for the
8	Claim 5, pages 8-9: "The method of claim 2, where the logic synthesis of step a) uses	capacitance at the output of gate i, where L is the imposed capacitance, h_{j} 's are the
9	addition to network delay, the network area	gains: \mathbf{r}_{i}
10	being estimated by setting the parameter C/S for each gate according to the method	(1) $c_i = \sum_{j \in fanout} \frac{1}{h_{ij}} + L_i$
11	of claim 2, followed by the method of step e) being used to estimate the sizes of the	(2) $c = Mc + L$ (3) $c = (I-M)^{-1}L$
12	gates, and hence the area of the network."	9: Circuit area $A = a^{T}c$ is given as the usual equation for total area (<i>a</i> is the area
13	Synopsys Patent Application #2, Claim	9: Gives formula for calculating net
14	1, Page 20 Constant delay synthesis consisting of "a	weights: "Weights are calculated from inputs to outputs"
15	method for the structuring and mapping of an unmapped digital network." " using	9: $A = a^{T}c = w^{T}L$, where L is the imposed capacitance, w is the weight vector
16	network slack as an optimization goal, where network slack is calculated assuming	
17	that the delay of the cells of the network is constant with respect to load." "c)	
18	Estimation of the area of the network based on net load."	
19		
20	Synopsys Patent Application #2, Claim 14, Page 22	
21	"(area estimation) c3) calculation of the sizes from the capacitive load, c4)	
22	calculation of the network area by summation of the product of the book area	
23	times the cell size."	
24	DrivingLHS, page 4, right column.	
25	section 5	
26	of the wire load and any primary output load. The capacitance c_i at the output of a	
27	gate <i>i</i> is calculated as the unscaled load, plus the sum of the capacitance on the	
28		
		– 22 – DECLARATION OF CARL SECHEN

1	fanout gates, divided by the gain h_{ij} of input <i>i</i> of fanout gate <i>i</i> .	
2	$a = a + \sum_{j=1}^{n} c_{j}$	
3	$C_i - Q_i + \sum_{j \in fanout(i)} \overline{h_{ij}}$	
4	Using matrix notation, c is the vector of gate capacitances, q is the vector of	
5	unscaled gate loads, and \mathbf{R} is the matrix of the reciprocal gain. The diagonal of \mathbf{R} is 0	
6	if we assume that no cell has an output	
7		
/ 0	$0 \frac{1}{h_{12}} \frac{1}{h_{12}} \dots$	
8	$\begin{bmatrix} 1 & 12 & 13 \\ 1 & 0 & 1 \end{bmatrix}$	
9	$R = \begin{vmatrix} h_{21} & h_{23} \end{vmatrix}$	
10	$\left \frac{1}{h_{0}} \frac{1}{h_{2}} 0 \dots \right $	
11	$\begin{bmatrix} n_{31} & n_{32} \\ \dots & \dots & 0 \end{bmatrix}$	
12	The load equation can now be written as \mathbf{c} = $\mathbf{a} + \mathbf{R}\mathbf{c}$ The capacitance can be solved as	
13	the fixed point of this equation: $\mathbf{a} = (\mathbf{I} \mathbf{P})^{-1} \mathbf{a}$ A positive real solution exists	
14	$\mathbf{C} = (\mathbf{I} - \mathbf{K})^{T} \mathbf{q}$. A positive real solution exists if the largest eigenvalue of \mathbf{R} is less than 1:	
15	$\lambda_1 < 1$. To calculate the area of the network, it is	
16	assumed that the area of a gate depends linearly on the load of that gate. The <i>area</i>	
10	sensitivity of a gate with respect to its output load a (vector a) is the marginal	
17	increase in area as a result of an increase in $1 \le 1 $	
18	linearly on the load, this is the area of gate <i>i</i>	
19	driving 1 unit of load. The area of the network is now $A = \mathbf{a}^{T} \mathbf{c}$."	
20	35. Area Estimation. The followin	ng chart lists the disclosures in the Synonsys
21		
22	Materials (left column) and in the Magma Mate	erials (right column) of the concept of computing a
23	weight for each net and using those net weights	s to estimate circuit area in the constant delay
24	paradigm.	
27	Symponeus Detent Application #2 Claime	Magmall (Lukas Coin based
23	8, Page 21	synthesis' slides), Pages 9, 10
26	" area savings are estimated using net yeights which reflect the change of yeights which reflect t	9: Gives formulas for calculating net weights:
27	network area due to sizing as a function of	$A = a^T c = \sum a_i c_i$
28		

1	Service Present Annulisation #2 Claim	$w = \sum_{j=1}^{w_j} w_j$
2	9, Page 21	$w_i - \sum_{j \in fanin} \frac{1}{h_{ij}} + a_i$
3	by starting at the primary inputs and	$\mathbf{w} = \mathbf{M}_{\mathbf{x}}\mathbf{w} + \mathbf{a}$
4	traversing the network in the direction of the data flow, while calculating the net	$\mathbf{A} = \mathbf{a}^{\mathrm{T}}\mathbf{c} = \mathbf{w}\mathbf{L}$
5	weight of a cell by summing over the fanin of the cell the product of the net weight of	"Weights are calculated from inputs to outputs"
6	the famin cell divided by the gain plus the cells area/load sensitivity, as W $i = sum j$	9: $A = a^{*}c = w^{*}L$, where L is the imposed capacitance, w is the weight vector
7	W_j / g_ij"	10: "Placement formulation: $A = wL$; The active area of the circuit can be expressed
8	Synopsys Patent Application #2, Claim 31 Pages 25-26	10: "Note: Placement programs naturally like to optimize a weighted sum of wire
9 10	("Calculation of net weights") step f1) is performed by starting at the primary inputs	lengths"
11	and traversing the network in the direction of the data flow, while calculating the net weight of a cell by summing over the famin	
12	of the cell the product of the net weight of the fanin cell divided by the gain plus the	
13	cells area/load sensitivity, as W_i = sum_j W_j / g_ij"	
14		
15	36. Stretching Constant Delays.	The following chart lists the disclosures in the
16 17	Synopsys Materials (left column) and in the I	Magma Materials (right column) of the concept of
18	adjusting (or stretching, compressing, trimmi	ng, etc.) the constant delay for gates during the logic
19	synthesis and physical design.	
20		
21	Synopsys Patent Application #2, Claims 10-12, Page 21	Magma98 (Lukas 'Gain based synthesis' slides). Page 14
22	Claim 10: "(stretching) a1) choosing a	14: "Delay trimming: Zero slack
22	delay of each cell based on the slack." Claim 11: " the delay of each cell is	argontinin
24	adjusted equally among the stages which have the same slack."	
25	Claim 12: " the delay is adjusted on each path, such that the slack of each path	
26	becomes 0."	
27	DrivingLHS, page 5, left and right column	
28		

1	"Stretching is a method for area optimization by changing the delays of the	
2	gates. Note that here delay is an independent variable and size load and	
3	area depend on the delay."	
4	"Stretching the delay of a gate increases its gain and decreases its size, thus saying	
5	area."	
6	"Section 2 [due to Sutherland, et al.] suggests an interesting and simple	
7	approach to stretching. According to the simplified path sizing problem in that	
8	section, the effort delay of each stage should be equal. Assuming that the effort	
9	delay of each stage already was equal, this means that the delay of each stage needs to	
10	be increased by the same amount, regardless of function load or size."	
11	regulatess of function, four of side.	
12	Comparison Of Concepts To Public	e Disclosures
13	37. I next compared the concepts	that I had identified in the Synopsys Materials with
14	information in Exhibits N, P, Q, R, S, T, and	VV (collectively "the Public Materials").
15	38. Constant Delay. The followi	ng chart lists the disclosures in the Synopsys
16	Materials (left column) and in the Public Mat	terials (right column) of the concept of holding the
17	dalay accordiated with each gate constant duri	ng logic synthesis and physical design
18		ng logic synthesis and physical design.
19	Synopsys Patent Application #1,	OttenSlides ICCAD '96, slides 32, 37,
20	Claim 1, Page 7: " a) logic synthesis of the digital	49, 50, 51: 32: "If $C_{\rm I}/C_{\rm in}$ is kept constant, delay
21	network, using network delay as an optimization goal, where the delay of each	doesn't vary either" (constant delay) 37: "Principle of uniform stage effort"
22	gate is presumed constant. b) Placement of the gates in a two dimensional plane. c)	(gain) (Sutherland/Sproul) 49: formula for area minimization Area =
23	Sizing of the gates in the network such that the network meets the cycle delay as	$\mathbf{a}^{\mathrm{T}}\mathbf{c}$, where $\mathbf{c} = (\mathbf{I} - \mathbf{F})^{-1}\mathbf{q}$ (q is imposed cap.)
24	computed by step a)."	49: "constant delay" 50: "delays are constant"
25	DrivingLHS. page 1. left column and	51: "implicit sizing" "no iterative timing analysis"
26	right column: "In the ICCAD proceedings of 1005	
27	authors from Digital Equipment	 ICCAD'96 brochure, intended tutorial abstract_page 24:
28	Corporation published two papers, in	ausuaci, page 24.
		- 25 - DECLARATION OF CARL SECHEN C04-3923 MMC
1		

 "The essence of all these papers is that they reverse the causality between the delay and gate size. Normally, the <i>delay</i> of a gate is expressed as a function of its capacitive <i>load</i> and the <i>size</i> of the gate is assumed to be a given constant DrivingLHS, page 2, left column: "In the new formulation, the delay becomes a parameter of the design, which is adjusted by an optimization algorithm, and its effect on gate size is observed using load analysis. Since delay is now constant with respect to size and load, this approach will be called the constant delay is a 'conscious' design decision." DrivingLHS, page 3, left column: "It is interesting though that under these assumptions the delays on a path should be the same, apart from the intrinsic delay. DrivingLHS, page 3, left column: "It is interesting though that under these assumptions the delays on a path should be the same, apart from the intrinsic delay. DrivingLHS, page 3, left column: "It is interesting though that under these assumptions the delays on a path should be the same, apart from the intrinsic delay. DrivingLHS, page 3, left column: "It is interesting though that under these assumptions the delays on a path should be the same, apart from the intrinsic delay. DrivingLHS, page 3, left column: "It is interesting though that under these assumptions the delays on a path should be the same, apart from the intrinsic delay. DrivingLHS, page 3, left column: "It is interesting though that under these assumptions the delays on a path should be the same, apart from the intrinsic delay. DrivingLHS, page 3, left column: "It is interesting though that under these assumptions the delays on a path should be the same, apart from the intrinsic delay. DrivingLHS, page 3, left column: "It is interesting though that under these assumptions the delays on a path should be the same, apart from the intrinsic delay. DrivingLHS, page 3, left column: "It is interesting th	 keep the delay constant, while expressing the size as a function of the load." "In a companion paper [2] Eric Lehman et al. use the constant delay model in a mapping algorithm." "In an earlier paper [5] the constant delay model is applied to fanout optimization." "The constant delay model is not really different than conventional models. The difference merely is which variable is held constant, when the load on a gate changes." "The essence of all these papers is that they reverse the causality between the delay and gate size. Normally, the <i>delay</i> of a gate is expressed as a function of its capacitive <i>load</i> and the <i>size</i> of the gate is assumed to be a given constant Divise LUE = 2.1.6 a.d.
--	---

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18	It also should be noted that in the constant delay model, the <i>delay is constant by definition.</i> "	timing budgets, or rather keep them right on target Layout synthesis should produce a network in which each gate causes exactly that delay. This is called <i>constant delay synthesis</i> [6 – Lehman, et al.]. Given a fixed delay for a gate, its size becomes a function of the output capacitance." " buffer insertion can only serve as an area reduction trick." Page 5, Section 4.1 ("Fixed delay"): describes Sutherland's results and introduces the notion of fixed delay synthesis. "The important observation is that τ can be kept constant by fixing $f = C_{in}/C_L$. This leads to a new paradigm in synthesis [6,9]: any delay imposed by synthesis can be realized, provided that the sizes of the gates can be continuously adjusted. In [6] the authors show that the size of a gate varies linearly with the load under constant delay. This enables size assignment after logic synthesis has fixed the <i>scaling factor</i> <i>f</i> for all gates"
18 19		the cell's delay a linear function of load. Our model is based on a different, but equally fundamental linearity in the equation relating area. delay, and load:
20		namely, we may keep a cell's delay
21		constant by making its area a linear function of load."
22		Page 1, left column: "We propose a new model for continuously-sized CMOS gates
23		In this model, a cell's delay will be held
24		constant. As the cell's load changes, the cell's size automatically grows exactly
25		enough to hold delay constant; making its area a function in fact a linear function
26		of load."
27		Page 1, right column: "Our own application is for continuously-sized, full
28		custom designs. However, the delay model
20		- 27 - DECLARATION OF CARL SI C04-392

1	such as high-end standard cell, where there
2	are many sizes of each cell. Essentially, it applies to any technology where cell sizing
3	to obtain a desired delay is viable."
4	modeling has been used frequently in
5	technology-independent algorithms."
5	different. We use the constant-delay model
6	in a technology-mapped realm as the
7	representation."
8	Page 1, right column: "A companion paper
9	[14] takes a different approach. Just as previous works have used a simple delay
10	model to explore many different
11	structurings of a technology independent network. [14] uses our delay model's
11	computational simplicity to explore a wide
12	space of network restructurings in the mapped realm."
13	
14	Sutherland et al.
15	Lukas, Kudva, Shenoy '98 (Size Indep.
16	Synthesis), pages 1, 2, 3:
17	applies to the concept of 'logical effort' to
18	logic synthesis. The concept of logical
10	independent delay model for logic
19	synthesis. Methods to determine initial
20	analysis and delay optimization methods
21	based on 'electrical effort' or gain are
22	the best way to use the methodology, it
23	shows that discrete libraries can also be
24	an independent variable as opposed to gate
25	size which is traditionally used."
20 26	Page 1, left column: "In the ICCAD
20	proceedings of 1995, two papers were published by authors from Digital
27	Equipment Corporation, which proposed a
28	synthesis method based on a delay model

1	for continuously sizable libraries. In the
2	first paper [1] Joel Grodstein et al. observe that if continuous sizing of gates is
3	held constant. A gate is sized by multiplying all transistor widths by the
4	same <i>gate size</i> The delay of a gate can
5	be held constant by increasing the width of the transistors proportionally to the
6	capacitive load that needs to be driven.
7	the delay constant, while expressing the size as a function of the load."
8	Daga 1 laft column:
9	"In a companion paper [2] Eric Lehman et
10	al. use the constant delay model in a mapping algorithm."
11	Page 1, left column:
12	"In an earlier paper [5] the constant delay model is applied to fanout optimization."
13	Page 1, right column: What is alleged to be
14	new over Sutherland and Grodstein is 'stretching' and 'compressing': "There is a
15	significant difference between the constant
16	delay model as proposed in [1, Grodstein] and the size independent model which is
17	derived from [3, Sutherland, et al.] and
18	constant delay model, the delay is
19	determined during initial library analysis and kept constant throughout synthesis. In
20	the size independent delay model on the other hand, the library is analyzed to
21	determine the size independent delay
22	model for the library as well as for determining initial conditions, but the
23	delay of a gate may be changed during the
24	appropriate delay optimization techniques.
25	Page 1, right column:
26	"The size independent model is no different than conventional models. The
27	difference merely is which variable is held constant, when the load on a gate changes."
28	Page 1, right column:
	- 29 - DECLARATION OF CARLS

1		"The essence of all the papers discussed is that they reverse the relationship between
2		the delay and gate size. Normally, the <i>delay</i> of a gate is expressed as a function of its capacitive <i>load</i> and the size of the gate
5 4		is assumed to be a given constant."
5		Page 2, left column:
6		becomes a parameter of the design, which is adjusted by an optimization algorithm.
7		and its effect on gate size is observed using area analysis. Since delay is determined
8		independent of size, this approach will be called the <i>size independent delay model</i> .
9		Note that the delay of a gate does not need to remain constant throughout synthesis,
10		design decision."
11		Page 3, left column: "It is interesting though that under these
12		assumptions the delays on a path should be the same, apart from the intrinsic delay."
13		Page 3, right column (Section 4, "Size
14 15		independent synthesis"): "Evaluating a synthesis change in the network becomes
15 16		easier: since the delays are size
10		of time to recalculate slacks can be
18		reduced.
19	39. Constant Delay Synthesis.	The following chart lists the disclosures in the
20	Synopsys Materials (left column) and in the	Public Materials (right column) of the concept of
21	applying constant delay to the synthesis of di	gital circuits (including structuring and mapping).
22		
23	 Synopsys Patent Application #2, Claim 1, Page 20 	 OttenSlides ICCAD '96, slide 51 51: "implicit sizing" "no iterative timing
24	Constant delay synthesis consisting of "a method for the structuring and mapping of	analysis"
25	an unmapped digital network." " using network slack as an optimization goal,	 ICCAD'96 brochure, intended tutorial
26	where network slack is calculated assuming that the delay of the cells of the network is	abstract, page 24"We will assume that, using custom
27 28	constant with respect to load." "c) Estimation of the area of the network based on net load."	layout cell generators, continuous gate sizing is possible."
		- 30 - DECLARATION OF CARL SECHE

1		
1 2 3 4 5	 Synopsys Patent Application #2, Claim 13, Page 22 "(globally optimal mapping)" b1) a traversal of the network from primary inputs and registers from left to right, while choosing at each cell the fastest matching books from all available matching books, using the constant delays of the books and 	 "We propose to make delay an independent variable, and remove both load and gate size from the delay equation" "This enables new formulations of many existing optimization algorithms and potentially affects the entire path from high level synthesis down to cell layout."
6 7	the fastest arrival times of the fanins of the matching book. b2) a traversal of the	 "With constant delay, the size of a gate varies as a function of its load."
8	at each cell the fastest matching book from the candidates selected during the previous	• "We will extend Sutherland & Sproul's theory of "Logical Effort" and apply it
9	traversal."	to logic synthesis.""In this context, we review the mapping
10	 Synopsys Patent Application #2, Claim 20 Page 23 	algorithm of Eric Lehman, et al."
11	"A method for the placement and sizing of cells of a mapped digital network d	 OttenISPD'98 (Global Wires
12 13	Choosing a target delay for each cell; e) Computing the network slack using the	Harmful?), page 4, right column "This leads to a new paradigm in synthesis
14	target delays; f) Placement of the cells of the network; g) Sizing of the cells of the network such that the network meets the	Lukas]: any delay imposed by synthesis can be realized, provided that the sizes of
15	network slack as computed by step b)."	the gates can be continuously adjusted." "Under the constant delay paradigm"
16		> OttenISPD'08 (Global Wires
17		Harmful?), page 5, left column
18 19		under the paradigm." " layout synthesis is capable of realizing
20		gates with a priori imposed delays."
21		 OttenDAC'98 (Planning for
22		 Performance), page 5 (section 4) Page 5, Section 4 ("Layout Synthesis"):
23		" layout synthesis should realize the functional blocks in such a way that the delaye in the blocks do not
24		timing budgets, or rather keep them right on target I ayout synthesis should
25		produce a network in which each gate causes exactly that delay. This is called
26		<i>constant delay synthesis</i> [6 – Lehman, et al.]. Given a fixed delay for a gate, its size
∠/ 28		becomes a function of the output capacitance."

1	area reduction trick."
2	• Page 5, Section 4.1 (Fixed delay): describes Sutherland's results and
3	introduces the notion of fixed delay
1	synthesis. "The important observation is that τ can be
4	kept constant by fixing $f = C_{in}/C_L$. This
5	any delay imposed by synthesis can be
6	realized, provided that the sizes of the gates can be continuously adjusted. In [6] the
7	authors show that the size of a gate varies
8	This enables size assignment after logic
9	synthesis has fixed the <i>scaling factor f</i> for all gates"
10	
10	➢ Grodstein, et al., page 1
11	• Page 1, abstract, left column: "We
12	logic synthesis. A traditional model
13	treats the area of a library cell as
14	linear function of load. Our model is
15	based on a different, but equally
15	relating area, delay, and load: namely,
16	we may keep a cell's delay constant by
17	making its area a linear function of load."
18	• Page 1, left column: "We propose a
19	new model for continuously-sized CMOS gates. In this model, a cell's
20	delay will be held constant. As the
 21	cell's load changes, the cell's size automatically grows exactly enough to
21	hold delay constant; making its area a
22	tunction in fact a linear function of load."
23	• Page 1, right column: "Our own
24	application is for continuously-sized, full custom designs. However, the
25	delay model is also applicable to other
26	methodologies, such as high-end standard cell, where there are many
27	sizes of each cell. Essentially, it applies
21	to any technology where cell sizing to obtain a desired delay is viable."
28	

ase 3:04-cv-03923-MMC Document 177	7-1 Filed 06/13/2005 Page 33 of 45
	 Page 1, right column: "Constant-delay modeling has been used frequently in technology-independent algorithms." Page 1, right column: "Our approach is different. We use the constant-delay model in a technology-mapped realm as the synthesis tool's most exact library
	 Page 1, right column: "A companion paper [14] takes a different approach. Just as previous works have used a simple delay model to explore many different structurings of a technology independent network, [141 uses our delay model's computational simplicity to explore a wide space of network restructurings in the mapped realm."
	Sutherland et al.
	 Lukas, Kudva, Shenoy '98 (Size Indep. Synthesis), page 3 (Sections 4, 5) "Size independent synthesis"): "Evaluating a synthesis change in the network becomes easier: since the delays are size independent in the new model, the amount of time to recalculate slacks can be reduced"
40. Constant Delay Set Via Op	timal Gain. The following chart lists the disclosu
of selecting the best gain for each gate and u associated with that gate.	using that gain to determine the constant delay
 Synopsys Patent Application #1, Claim 2, Page 7 "(Choice of constant delay) Claim 2, Page 7: "The method of claim 1, where the parameter C/S for each type of gate is chosen before step 1) and is used during step a) to calculate the cycle delay, said parameter being chosen to have the largest possible value such that a long chain of identical gates of this type, each 	 OttenDAC'98 (Planning for Performance), page 5, section 4.1 ("Fixed Delay") Describes Sutherland's results and introduces the notion of fixed delay synthesis. "The important observation is that τ can be kept constant by fixing f = C_{in}/C_L. This leads to a new paradigm in synthesis [6,9]:
	- 33 - DECLARATION OF CARL SECH C04-3923 M

1 2 3 4 5	gate in the chain having identical value of parameter C/S, said chain cannot have simultaneously improved delay and improved gain by adding a buffer at some point to the same chain, even when the parameter C/S is chosen optimally after adding the buffer."	any delay imposed by synthesis can be realized, provided that the sizes of the gates can be continuously adjusted. In [6] the authors show that the size of a gate varies linearly with the load under constant delay. This enables size assignment after logic synthesis has fixed the <i>scaling factor f</i> for all gates"
6 7 8 9 10 11 12 13 14	 Synopsys Patent Application #2, Claim 3, Page 20 "the delay of the book being chosen by choosing a load size ratio C/S for each book, which is independent of the network." Synopsys Patent Application #2, Claim 5, Page 20 " a parameter C/S for each book is chosen to have the largest possible value such that a long chain of identical books each cell in the chain having identical value of parameter C/S, said chain cannot have simultaneously improved delay and improved gain by adding a buffer at some point to the same chain, even when the parameter C/S is chosen optimally after 	 Sutherland et al. Lukas, Kudva, Shenoy '98 (Size Indep. Synthesis), pages 2, 3, Section 10 (pages 5, 6) Page 2, left column: "In the new formulation, the delay becomes a parameter of the design, which is adjusted by an optimization algorithm, and its effect on gate size is observed using area analysis. Since delay is determined independent of size, this approach will be called the <i>size independent delay model</i>. Note that the delay of a gate does not need to remain constant throughout synthesis, but a change in delay is a 'conscious' design decision."
 15 16 17 18 19 20 21 	 adding the buffer." Synopsys Patent Application #2, Claim 20, Page 23 "A method for the placement and sizing of cells of a mapped digital network d) Choosing a target delay for each cell; e) Computing the network slack using the target delays; f) Placement of the cells of the network; g) Sizing of the cells of the network such that the network meets the network slack as computed by step b)." 	 Page 3, left column: "It is interesting though that under these assumptions the delays on a path should be the same, apart from the intrinsic delay." Page 3, right column (Section 4, "Size independent synthesis"): "Evaluating a synthesis change in the network becomes easier: since the delays are size independent in the new model, the amount of time to recalculate slacks can be reduced."
 22 23 24 25 26 27 28 	 DrivingLHS, page 6, right column, Section 8 ("Library analysis"): "A better method is the continuous buffering assumption, based on the ideas of [3]." "The delay of an amplifier in the technology is chosen as follows. Consider a chain of <i>N</i> amplifiers, and a given gain requirement <i>H</i> = h^N for them. For which stage <i>h</i> is the delay <i>D</i> = N(<i>p</i> + gh) 	 Page 5, right column, Section 10 (Library analysis): "A better method is the continuous buffering assumption, based on the ideas of [3]." [Sutherland, et al.] Then, this follows Sutherland, et al., as well: "The delay of an amplifier in the technology is chosen as follows. Consider a chain of <i>N</i> amplifiers, and a given gain requirement <i>H</i> = <i>h^N</i> for them. For which stage <i>h</i> is

minimal? Since $N = \ln(H)/\ln(h)$ the delay is $D = \ln(H)/\ln(h) (p + gh)$. Now lets minimize the delay by choosing h:	the delay $D = N(p + gh)$ minimal? Since $N = \ln(H)/\ln(h)$ the delay is $D = \ln(H)/\ln(h)$ ($p + gh$) Now lets	
$\frac{\P D}{\P h} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \frac{1}{\ln^2(h)} \right] g - \frac{1}{\ln^2(h)} dh$	minimize the delay by choosing h :	
$\begin{bmatrix} n & l(m(n) & m(n)) \\ 1 & 1 \end{bmatrix}$	$\frac{\P D}{\P h} = \ln(H) \left[\left(\frac{1}{\ln(h)} - \frac{1}{\ln^2(h)} \right) g - \right]$	
$\frac{1}{\ln^2(h)} \frac{1}{h} p = 0$	$\frac{1}{1} \frac{1}{n} = 0$	
"Which simplifies to $g \ln(h) - g - p/h = 0$. For a CMOS inverter, $p = ag$, which gives	$\ln^2(h) h^P \int$	
us $a = h \ln(h) - h$. for a between 0.8 and 2 the gain h is about 3.4 to 4.4. The answer given by [8] is that the stage gain should be	"Which simplifies to $g \ln(h) - g - p/h = 0$.	
e=2.718 this analysis ignored the parasitic delay p of the inverters."	ln(h) = h + a. For a self-load between 0.7 and 0.9 for the amplifiers. If this is not	
	ignored, the answer is closer to 3.6."	
41. Buffer Insertion For Area	Minimization The following chart lists the	
41. Durier insertion For Area Minimization. The following chart lists the		
disclosures in the Synopsys Materials (left column) and in the Public Materials (right column) of		
the concept of inserting a buffer (a gate that performs no logical function but boosts signal		
the concept of inserting a burier (a gate that	performs no logical function but boosts signal	
strength) for area minimization in the consta	ant delay paradigm (to achieve the optimal stage ef	
strength) for area minimization in the const and therefore the optimal gain for a stage).	ant delay paradigm (to achieve the optimal stage ef	
 strength) for area minimization in the constant and therefore the optimal gain for a stage). > Synopsys Patent Application #1, Claim 	ant delay paradigm (to achieve the optimal stage ef	
 strength) for area minimization in the constant and therefore the optimal gain for a stage). Synopsys Patent Application #1, Claim 6, Page 9 	 A performs no logical function but boosts signal ant delay paradigm (to achieve the optimal stage ef OttenDAC'98 (Planning for Performance), page 5, section 4 	
 strength) for area minimization in the constant and therefore the optimal gain for a stage). Synopsys Patent Application #1, Claim 6, Page 9 " the buffers being inserted on non-critical paths, as determined by subtracting 	 > OttenDAC'98 (Planning for Performance), page 5, section 4 Page 5, Section 4 ("Layout Synthesis"): 	
 strength) for area minimization in the constant and therefore the optimal gain for a stage). Synopsys Patent Application #1, Claim 6, Page 9 " the buffers being inserted on non-critical paths, as determined by subtracting the delay of the buffer from the slack of the path, while estimating the area reduction 	 > OttenDAC'98 (Planning for Performance), page 5, section 4 Page 5, Section 4 ("Layout Synthesis"): " layout synthesis should realize the functional blocks in such a way that the delays in the blocks do not exceed their 	
 strength) for area minimization in the const and therefore the optimal gain for a stage). Synopsys Patent Application #1, Claim 6, Page 9 " the buffers being inserted on non-critical paths, as determined by subtracting the delay of the buffer from the slack of the path, while estimating the area reduction" 	 Deforms no logical function but boosts signal ant delay paradigm (to achieve the optimal stage ef Performance), page 5, section 4 Page 5, Section 4 ("Layout Synthesis"): layout synthesis should realize the functional blocks in such a way that the delays in the blocks do not exceed their timing budgets, or rather keep them right on target Layout synthesis should 	
 strength) for area minimization in the const and therefore the optimal gain for a stage). Synopsys Patent Application #1, Claim 6, Page 9 the buffers being inserted on noncritical paths, as determined by subtracting the delay of the buffer from the slack of the path, while estimating the area reduction Synopsys Patent Application #2, Claims 6-7 Page 21 	 Defiorms no logical function but boosts signal ant delay paradigm (to achieve the optimal stage ef Performance), page 5, section 4 Page 5, Section 4 ("Layout Synthesis"): layout synthesis should realize the functional blocks in such a way that the delays in the blocks do not exceed their timing budgets, or rather keep them right on target Layout synthesis should produce a network in which each gate causes exactly that delay. This is called <i>constant delay synthesis</i> [6 – Lehman et] 	
 strength) for area minimization in the const and therefore the optimal gain for a stage). Synopsys Patent Application #1, Claim 6, Page 9 " the buffers being inserted on non-critical paths, as determined by subtracting the delay of the buffer from the slack of the path, while estimating the area reduction" Synopsys Patent Application #2, Claims 6-7, Page 21 " buffer insertion, the buffers being inserted on paths with positive slack, as 	 Performs no logical function but boosts signal ant delay paradigm (to achieve the optimal stage ef Performance), page 5, section 4 Page 5, Section 4 ("Layout Synthesis"): layout synthesis should realize the functional blocks in such a way that the delays in the blocks do not exceed their timing budgets, or rather keep them right on target Layout synthesis should produce a network in which each gate causes exactly that delay. This is called <i>constant delay synthesis</i> [6 – Lehman, et al.]. Given a fixed delay for a gate, its size becomes a function of the output 	
 strength) for area minimization in the const and therefore the optimal gain for a stage). Synopsys Patent Application #1, Claim 6, Page 9 " the buffers being inserted on non-critical paths, as determined by subtracting the delay of the buffer from the slack of the path, while estimating the area reduction" Synopsys Patent Application #2, Claims 6-7, Page 21 " buffer insertion, the buffers being inserted on paths with positive slack, as determined by subtracting the delay of the slack of the path." 	 Defioring no logical function but boosts signal ant delay paradigm (to achieve the optimal stage ef Performance), page 5, section 4 Page 5, Section 4 ("Layout Synthesis"): " layout synthesis should realize the functional blocks in such a way that the delays in the blocks do not exceed their timing budgets, or rather keep them right on target Layout synthesis should produce a network in which each gate causes exactly that delay. This is called <i>constant delay synthesis</i> [6 – Lehman, et al.]. Given a fixed delay for a gate, its size becomes a function of the output capacitance." " buffer insertion can only serve as an area reduction trick." 	
 strength) for area minimization in the const and therefore the optimal gain for a stage). Synopsys Patent Application #1, Claim 6, Page 9 " the buffers being inserted on non-critical paths, as determined by subtracting the delay of the buffer from the slack of the path, while estimating the area reduction" Synopsys Patent Application #2, Claims 6-7, Page 21 " buffer insertion, the buffers being inserted on paths with positive slack, as determined by subtracting the delay of the slack of the path." " buffer insertion, the buffers being inserted on paths with positive slack, as determined by subtracting the delay of the buffer from the slack of the path." 	 Performs no logical function but boosts signal ant delay paradigm (to achieve the optimal stage ef Performance), page 5, section 4 Page 5, Section 4 ("Layout Synthesis"): " layout synthesis should realize the functional blocks in such a way that the delays in the blocks do not exceed their timing budgets, or rather keep them right on target Layout synthesis should produce a network in which each gate causes exactly that delay. This is called <i>constant delay synthesis</i> [6 – Lehman, et al.]. Given a fixed delay for a gate, its size becomes a function of the output capacitance." " buffer insertion can only serve as an area reduction trick." 	

1 2 3	Determining whether or not a buffer should be added: "Additionally, a buffer can save area. This can be checked using area analysis. Assume a buffer <i>h</i> is added at the output of	 Lukas, Kudva, Shenoy '98 (Size Indep. Synthesis), page 4 Page 4 (right column), Section 7 ("Area
4	gate g. Let c' be the updated net loads, then $\Delta A = a^{T} (c - c')$. The area of the buffer is	optimization"), contains this on determining whether or not a buffer
5	$A_b - a_b c_b$, which is added to the circuit. The buffer saves area if $a^T (c - c') > a_b c_b$."	"Additionally, a buffer must save area. This
6		Assume a buffer b is added at the output of gate g . Let c' be the updated net loads, then
7 8		$\Delta A = a^{T} (c - c^{2})$. The area of the buffer is $A_{b} - a_{b}c_{b}$, which is added to the circuit. The buffer saves area if $a^{T} (c - c^{2}) > a_{b}c_{b}$
9	42. Sizing Driven Placement. T	he following chart lists the disclosures in the
10	Synopsys Materials (left column) and in the I	Public Materials (right column) of the concept of
11	changing cell sizes during iterative placemen	t with the objective of holding the delays of each cell
12 13	constant.	
 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 	 Synopsys Patent Application #1, Claims 7, 8; Pages 9, 10 Claim 7: " where step b) [placement] is performed by repeated partitioning steps, partitioning the gates in the network into two or more groups, each group being assigned to an subdivision of the plane, alternating the partitioning steps with sizing steps" Claim 8: iterative placement improve ment: " repeatedly changing the location of one or two gates at a time, while performing a sizing step c) after each location change." Synopsys Patent Application #2, Claims 20-23, Page 23 Claim 20: "A method for the placement and sizing of cells of a mapped digital network d) Choosing a target delay for each cell; e) Computing the network slack using the target delays; f) Placement of the cells of the network; g) Sizing of the cells of the network such that the network meets the network slack as computed by step b)." Claim 21: Executing step f) in Claim 20 "where step f) is performed in gradual steps, each step being followed by a sizing 	 Lukas, Kudva, Shenoy '98 (Size Indep. Synthesis), page 5 Section 8 (page 5, left column) entitled "Placement and size independence". "The quality of the placement solution relies on the relative net weighting metric used" "Since placement depends strongly on the areas of the cells and their physical dimensions, any attempt to use the size independent network for placement must ensure that the area estimation is carried out with sufficient accuracy. Placement algorithms can then be modified to including sizing. For example one may consider a bipartitioning placement program that begins sizing the gates after a certain number of cuts based on Steiner estimates for wire length."

ase 3:04-cv-03923-MMC Document 177	-1 Filed 06/13/2005 Page 37 of 45
step g)" Claim 22: Placement by partitioning, a sizing step being executed after each iteration. Claim 23: Placement by iterative improvement (moving one or two cells at a time), a sizing step being executed after each iteration.	
43. Net Weight Placement. The	following chart lists the disclosures in the Syno
Materials (left column) and in the Public Mat	terials (right column) of the concept of computir
weight for each net that reflects the degree to	which additional load impacts overall circuit are
and applying those net weights during placer	nent.
 Synopsys Patent Application #1, Claims 9,10, Page 10 deal with net weights for placement. " the weight being proportional to the area divided by the typical load." 	 Lukas, Kudva, Shenoy '98 (Size Indep. Synthesis), page 5 Section 8 (page 5, left column) entitled "Placement and size independence". "The enablity of the placement of the second se
 Synopsys Patent Application #2, Claim 30, Page 24 ("Weighted Placement") f1) The calculation of net weights that reflect the change of network area due to sizing as a function of net length; f2) Placement of the cells of the network where the weighted network net length is used as a placement objective, the weighted network net length being the sum of the weighted net lengths of all nets, each net length being multiplied by a weight." 	"The quality of the placement solution relies on the relative net weighting metric used" "Size independent synthesis provides a natural form of net weights. If we choose to associate with each net, the area sensitivity of the gate driving it, then the objective function in the placement problem (or a component of it) measures the total active area required to meet the target constraints. Since placement depends strongly on the areas of the cells and their physical dimensions, any attempt to use the size independent network for placement must ensure that the area estimation is carried out with sufficient accuracy.
44. Continuous Gate Sizing. The Synonsys Materials (left column) and in the l	e following chart lists the disclosures in the Public Materials (right column) of the concept of
employing continuous sizing of a gata to mo	intain a constant delay for that gots during locic
employing continuous sizing of a gate to man	mani a constant delay for that gate during logic
synthesis and physical design.	
	- 37 - DECLARATION OF CARL SEC C04-3923 N

1		
\mathbf{r}	Synopsys Patent Application #1, Claim	OttenSlides ICCAD '96, slide 52
2	3, Page 8	"how to generate continuously sizeable
3	"The method of claim 2, where step c) is comprised of steps: d) Determining the	noraries:
1	parameter C/S for each gate according to	
-	delay and gain constraints on the network.	Sutherland et al.
5	traversing the network, in the direction	
6	opposite to the direction of data flow, while	➢ Lukas, Kudva, Shenoy '98 (Size Indep.
0	calculating the load of each gate, the size S	Synthesis), pages 4, 5
7	parameter C/S and the load C of said gate.	Page 4, right column:
8	the load on the preceding gate being	Also: "Stretching is a method for area optimization by changing the delays of the
Ŭ	determined by the size of the fanout gates,	gates. Note that here delay is an
9	norating step c) until convergence.	independent variable and size, load and
10		area depend on the delay. Stretching the
1.1	Synopsys Patent Application #2, Claim 15, Page 22	delay of a gate increases its gain and
11	"(sizing algorithm) starting at the	decreases its size, thus saving area.
12	primary outputs and traversing the network	algorithm, except that the stretching
10	in the direction opposite to the data flow,	algorithm changes delays as design
13	summing over the fanout of the cell, the	decisions and uses area as an objective
14	product of the load of the fanout cell	function. Here the delays of the gates are
15	divided by the gain plus the net load of the cell "	Increased to save area."
15		of the size independent delay model
16		approach is that each gate can be sized
17	Synopsys Patent Application #2, Claim 20, Page 23	linearly to a given load. If the sizes of the
17	"A method for the placement and sizing of	transistors can continuously be adjusted,
18	cells of a mapped digital network d)	this is close to accurate."
19	Choosing a target delay for each cell; e) Computing the network slack using the	
	target delays; f) Placement of the cells of	
20	the network; g) Sizing of the cells of the	
21	network slack as computed by step b)."	
22	1 5	
22	Synonsys Patent Application #2	
23	Claims 24-25. Page 23-24	
24	Claim 24: "(sizing) g1) calculation of	
∠4	the net length based on the available	
25	the capacitive load of the cells using the net	
26	length; g3) calculation of the sizes from the	
	capacitive load." Claim 25: "(sizing algorithm) starting at	
27	the primary outputs and traversing the	
28	network in the direction opposite to the	

Ca	ase 3:04-cv-03923-MMC Document 177	-1 Filed 06/13/2005 Page 39 of 45
1	data flow, while calculating the load a cell,	
2	by summing over the fanout of the cell, the product of the load of the fanout cell	
3	divided by the gain plus the net load of the cell."	
4		
5	45. Discrete Gate Sizing. The fo	ollowing chart lists the disclosures in the Synopsy
6	Materials (left column) and in the Public Ma	terials (right column) of the concept of employing
7	discrete gate sizes with the objective of main	taining a constant dalay for that gate during logic
8	discrete gate sizes with the objective of man	taining a constant delay for that gate during logic
9	synthesis and physical design.	
10	DrivingLHS, page 5, right column.	OttenSlides ICCAD '96. slide 52
11	Section 7 ("Discrete sizing algorithms"):	"how to handle discrete libraries?"
12	"By using discrete gates in parallel, integer multiples of the gates can be created	Lukas Kudya Shenov '98 (Size Inden)
3	duplicated gate can drive exactly twice the	Synthesis), pages 1, 5
14	sizes can be mixed to create non-integer	applies to the concept of 'logical effort' to
15	multiples.	logic synthesis. The concept of logical effort is used to motivate the size
6		independent delay model for logic
17		delays for library cells are discussed. Area
8		analysis and delay optimization methods based on 'electrical effort' or gain are
9		discussed." " continuous gate sizing as
20		shows that discrete libraries can also be
21		supported. The method uses delay/gain as an independent variable as opposed to gate
22		size which is traditionally used."
22		algorithms):
2		An algorithm is needed to round the computed continuous sizes to the discrete
25		a network all of whose gates have only
26		delays assigned to them (say by a delay optimization program), into a network with
-0 07		all gates consisting of sized gates from a given library."
21		
.0		

1	46. Area Minimization. The fol	lowing chart lists the disclosures in the Synopsys	
2	Materials (left column) and in the Public Materials (right column) of the concept of formulating		
3	an equation that calculates the area of a circuit and using that equation to minimize the area while		
4 5	maintaining constant delay.		
6	Synopsys Patent Application #1 Claim	OttenSlides ICCAD '96 slide 49	
7	5, Pages 8-9 "(Area estimation)"	formula for area minimization: 49: c _i : capacitance at output i:	
8	Claim 5, pages 8-9: "The method of claim 2, where the logic synthesis of step a) uses	$c_i = q_i + \sum_{j=1}^{r} \frac{c_j}{r}$	
9	network area as an optimization goal, in addition to network delay, the network area	Area = $\mathbf{a}^{\mathrm{T}}\mathbf{c}$, where $\mathbf{c} = (\mathbf{I} - \mathbf{F})^{-1}\mathbf{q}$ (q is	
10	being estimated by setting the parameter C/S for each gate according to the method	imposed capacitance)	
11	of claim 2, followed by the method of step e) being used to estimate the sizes of the	 OttenISPD'98 (Global Wires Harmful?) page 4 right column 	
12	gates, and hence the area of the network."	Presents the usual formula for area	
13	Synopsys Patent Application #2, Claim	"If we collect the imposed capacitances in a vector \mathbf{q} and the capacitances at the	
14	1, Page 20 Constant delay synthesis consisting of "a	output in a vector \mathbf{c} we obtain the following relation:	
15	method for the structuring and mapping of an unmapped digital network." "	$(I - Nf^{D})c = q$	
16	network slack as an optimization goal, where network slack is calculated assuming	The matrix N has the zero/non-zero pattern of the (directed) gate-to-gate incidence	
17	that the delay of the cells of the network is constant with respect to load." "c)	matrix and can contain the individual relations between the input and output	
18	Estimation of the area of the network based on net load."	capacitance. The area of the network is then equal	
19		$(\mathbf{a} \bullet \mathbf{f})^T \mathbf{c} = (\mathbf{a} \bullet \mathbf{f})^T (\mathbf{I} - \mathbf{N}\mathbf{f}^D)^{-1} \mathbf{q}$	
20	Synopsys Patent Application #2, Claim 14, Page 22	find out whether inserting buffers does save	
21	"(area estimation) c3) calculation of the sizes from the capacitive load, $c^{(4)}$	inserted if they do not cause violations in	
22	calculation of the network area by	introduce additional delay locally which is	
23	times the cell size."	there."	
24	Driving LUC mass 4 right solumn	Lukas, Kudva, Shenoy '98 (Size Indep.	
25	section 5	Synthesis), pages 4, 5 Page 4, (Section 6, "Area Estimation"):	
26	"Let q_i be the unscaled load, which consists of the wire load and any primary output	The key concepts, including their development, was also presented in Otten's	
27	load. The capacitance c_i at the output of a gate <i>i</i> is calculated as the unscaled load,	slides.	
28	plus the sum of the capacitance on the	not end points and can be scaled (which is	
		- 40 - DECLARATION OF CARL SECHEN	

1	fanout gates, divided by the gain h_{ij} of input <i>i</i> of fanout gate <i>i</i> .	typically not the case), one can use the following theoretical formulation."
2	$c_i = q_i + \sum \frac{c_j}{h_i}$	$\begin{bmatrix} 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$
3	$j \in fanout(i)$ h^{uy} Using matrix notation, c is the vector of	$\begin{bmatrix} 0 & \frac{1}{h_{12}} & \frac{1}{h_{13}} & \dots \\ 1 & 1 & 1 \end{bmatrix}$
4 5	gate capacitances, q is the vector of unscaled gate loads, and R is the matrix of the reciprocal gain. The diagonal of R is 0	$\mathbf{R} = \begin{vmatrix} \frac{1}{h_{21}} & 0 & \frac{1}{h_{23}} & \dots \\ 1 & 1 & 1 \end{vmatrix}$
6	if we assume that no cell has an output connected to an input of itself.	$\frac{1}{h_{31}}$ $\frac{1}{h_{32}}$ 0
7	$0 \frac{1}{h_{12}} \frac{1}{h_{13}} \dots$	
8	$\mathbf{R} = \begin{bmatrix} \frac{1}{h} & 0 & \frac{1}{h} & \dots \end{bmatrix}$	"Where q_i be the unscaled load, which consists of the
9	$\frac{1}{\frac{1}{21}} = \frac{1}{23}$	$c_{i} = a_{i} + \sum \frac{c_{j}}{c_{j}}$
10	$\begin{bmatrix} h_{31} & h_{32} \\ \dots & \dots & 0 \end{bmatrix}$	$\begin{array}{ccc} & & & & \\ & & & \\ & & & \\ &$
11	The load equation can now be written as $\mathbf{c} = \mathbf{a} + \mathbf{B}\mathbf{c}$. The capacitance can be solved as	capacitance c_i at the output of a gate <i>i</i> is calculated as the unscaled load, plus the
12	the fixed point of this equation: $\mathbf{c} = (\mathbf{I} \cdot \mathbf{R})^{-1} \mathbf{a}$ A positive real solution exists	sum of the capacitance on the fanout gates, divided by the gain h_{ii} of input <i>i</i> of fanout
13	if the largest eigenvalue of R is less than 1: $\lambda_1 < 1$	gate j. Using matrix notation, c is the vector of gate capacitances g is the vector
14	To calculate the area of the network, it is assumed that the area of a gate depends	of unscaled gate loads, and \mathbf{R} is the matrix of the reciprocal gain. The diagonal of \mathbf{R} is
15	linearly on the load of that gate. The <i>area</i> sensitivity of a gate with respect to its	0 if we assume that no cell has an output connected to an input of itself. The load
10	output load a_i (vector a) is the marginal increase in area as a result of an increase in	equation can now be written as $\mathbf{c} = \mathbf{q} + \mathbf{R}\mathbf{c}$. The capacitance can be solved as the fixed
18	load. Since the area is assumed to depend linearly on the load, this is the area of gate i driving 1 unit of load. The area of the	point of this equation: $\mathbf{c} = (\mathbf{I} - \mathbf{R})^{-1}\mathbf{q}$. A positive real solution exists if the largest eigenvalue of R is less than 1: $\lambda_1 < 1$.
19	network is now $A = \mathbf{a}^{\mathrm{T}} \mathbf{c}$."	The <i>area sensitivity</i> of a gate with respect
20		to its output load a_i (vector a) is the marginal increase in area as a result of an
21		to depend linearly on the load, this is the
22 22		area of the network is now $A = \mathbf{a}^{\mathrm{T}} \mathbf{c}$."
23 24		
2 4 25	47. Area Estimation. The follow	ving chart lists the disclosures in the Synopsys
23 26	Materials (left column) and in the Public Mat	terials (right column) of the concept of computing a
27	weight for each net and using those net weigh	hts to estimate circuit area in the constant delay
28	paradigm.	
		- 41 - DECLARATION OF CARL SECHEN C04-3923 MMC

	Synopsys Patent Application #2, Claim	Lukas, Kudva, Shenoy '98 (Size Indep.	
<u>'</u>	8, Page 21	Synthesis), page 5	
;	" area savings are estimated using net	"The quality of the placement solution	
	network area due to sizing as a function of	relies on the relative net weighting metric	
	net length."	used"	
		natural form of net weights. If we choose to associate with each pet, the area sensitivity	
	Synopsys Patent Application #2, Claim 9, Page 21	of the gate driving it, then the objective function in the placement problem (or a	
	" calculation of net weights is performed by starting at the primary inputs and	component of it) measures the total active area required to meet the target constraints.	
	traversing the network in the direction of the data flow, while calculating the net weight of a cell by summing over the famin	Since placement depends strongly on the areas of the cells and their physical	
	of the cell the product of the net weight of the fanin cell divided by the gain plus the	independent network for placement must ensure that the area estimation is carried	
	cells area/load sensitivity, as W_i = sum_j W_j / g_ij"	out with sufficient accuracy. Placement algorithms can then be modified to	
		including sizing. For example one may consider a bipartitioning placement	
	 Synopsys Patent Application #2, Claim 31, Pages 25-26 	program that begins sizing the gates after a certain number of cuts based on Steiner	
	("Calculation of net weights") step f1) is performed by starting at the primary inputs	estimates for wire length."	
	of the data flow, while calculating the net		
	weight of a cell by summing over the fanin		
	the fanin cell divided by the gain plus the		
	cells area/load sensitivity, as W_i = sum_j		
	W_j / g_ij"		
	48. Stretching Constant Delays.	The following chart lists the disclosures in the	
	Synopsys Materials (left column) and in the Public Materials (right column) of the concept of		
	adjusting (or stretching, compressing, trimmi	ng, etc.) the constant delay for gates during the lo	
ļ	augusting (or succoming, compressing, uninning, etc.) the constant delay for gates during the		
	synthesis and physical design.		
	Synopsys Patent Application #2	▶ Lukas, Kudva, Shenov '98 (Size Indep.	
	Claims 10-12, Page 21	Synthesis), pages 1, 2, 3, 4	
	Claims 10-12, Page 21 Claim 10: "(stretching) a1) choosing a delay for each book a3) adjusting the	Synthesis), pages 1, 2, 3, 4 Page 1, right column: What is alleged to be	
	Claims 10-12, Page 21 Claim 10: "(stretching) a1) choosing a delay for each, book, a3) adjusting the delay of each cell based on the slack." Claim 11: " the delay of each cell is	Synthesis), pages 1, 2, 3, 4 Page 1, right column: What is alleged to be new over Sutherland and Grodstein is 'stretching' and 'compressing': "There is a	
	Claims 10-12, Page 21 Claim 10: "(stretching) a1) choosing a delay for each, book, a3) adjusting the delay of each cell based on the slack." Claim 11: " the delay of each cell is adjusted equally among the stages which	Synthesis), pages 1, 2, 3, 4 Page 1, right column: What is alleged to be new over Sutherland and Grodstein is 'stretching' and 'compressing': "There is a significant difference between the constant	
	Claims 10-12, Page 21 Claim 10: "(stretching) a1) choosing a delay for each, book, a3) adjusting the delay of each cell based on the slack." Claim 11: " the delay of each cell is adjusted equally among the stages which have the same slack."	Synthesis), pages 1, 2, 3, 4 Page 1, right column: What is alleged to be new over Sutherland and Grodstein is 'stretching' and 'compressing': "There is a significant difference between the constant delay model as proposed in [1, Grodstein] and the size independent model which is	

1	path, such that the slack of each path	derived from [3, Sutherland, et al.] and
2	becomes 0."	applied to synthesis in this paper. In the
2		constant delay model, the delay is determined during initial library analysis
3	DrivingLHS, page 5, left and right column	and kept constant throughout synthesis. In
4	"Stretching is a method for area	the size independent delay model on the
5	optimization by changing the delays of the gates. Note that here delay is an	other hand, the library is analyzed to determine the size independent delay
6	area depend on the delay."	determining initial conditions, but the delay
7	"Stretching the delay of a gate increases its	of a gate may be changed during the course of synthesis as determined by appropriate
8	gain and decreases its size, thus saving area."	delay optimization techniques.
9	"Section 2 [due to Sutherland, et al.]	Page 2, left column:
10	suggests an interesting and simple approach to stretching. According to the simplified path sizing problem in that	"In the new formulation, the delay becomes a parameter of the design, which is adjusted by an optimization algorithm, and its effect
11	section, the effort delay of each stage should be equal. Assuming that the effort	on gate size is observed using area analysis. Since delay is determined
12	delay of each stage already was equal, this means that the delay of each stage needs to	independent of size, this approach will be called the <i>size independent delay model</i>
13	be increased by the same amount,	Note that the delay of a gate does not need
14	regardless of function, load or size."	but a change in delay is a 'conscious' design decision."
15		
16 17		Page 3, right column (Section 4, "Size independent synthesis"): "Evaluating a synthesis change in the network becomes
17		easier: since the delays are size
18		independent in the new model, the amount
19		reduced."
20		Page 4, right column:
21		optimization by changing the delays of the
22		independent variable and size, load and
23		delay of a gate increases its gain and
24		decreases its size, thus saving area. Stretching is like a continuous sizing
25		algorithm, except that the stretching algorithm changes delays as design
26		decisions and uses area as an objective function. Here the delays of the gates are
27		increased to save area."
28		

4

5

1

Summary and Conclusions

49. As illustrated in the first set of charts above (paragraphs 26-36), each of the 11 concepts in the Synopsys Materials was disclosed at least once in the Magma Materials, and many concepts were disclosed multiple times in the Magma Materials.

50. As illustrated in the second set of charts above (paragraphs 38-48), each of the 11
concepts in the Synopsys Materials was disclosed at least once in the Public Materials, and many
concepts were disclosed multiple times in the Public Materials.

51. The paper "Size Independent Synthesis" by van Ginneken, Kudva and Shenoy
includes all of the concepts in the paper "Driving on the Left Hand Side of the Performance
Speedway" by van Ginneken. "Size Independent Synthesis" by van Ginneken, Kudva and Shenoy
has only one section that contains any significant new content over the previous van Ginneken
paper, and that is Section 8 (page 5, left column) entitled "Placement and size independence".

52. Constant delay was not a new concept when Grodstein wrote his paper (for
 example, see the comment on page 1, right column). Applying constant delay to the technology
 dependent case was introduced by Grodstein, et al. Sutherland, et al. in combination with
 Grodstein, et al. provides the theory of logical effort plus the notion of constant delay, applied to
 the technology dependent case. This combination teaches at least part of all of the concepts
 disclosed in the Synopsys Materials.

53. The paper "Driving on the Left Hand Side of the Performance Speedway"
acknowledges that: "The constant delay model is not really different than conventional models.
The difference merely is which variable is held constant, when the load on a gate changes."
Compare this to what Otten says in his ICCAD'96 tutorial slides: slide 52: "don't be fooled: the
problem did not change but the viewpoint did!"

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

The paper "Size Independent Synthesis" by van Ginneken, Kudva, and Shenoy 54. 2 concedes that logical effort and the notion of gain was introduced by Sutherland et al. It further 3 concedes that Grodstein et al. introduced constant delay, and the notion of constant delay synthesis. The paper claims to introduce the concept of stretching and compressing the delays. However, Otten's ICCAD tutorial had already revealed the concept of stretching.

55. Magma was very, very open with what they were working on, from presentations at conferences, trade shows, discussions with professors (such as myself). It would be inconceivable to me that Synopsys would not have known they were using logical effort plus constant delay synthesis.

56. Submitting a paper to a conference for publication is universally understood to be contrary to protecting trade secrets. A panel of experts reviews the papers without having to formally sign any type of nondisclosure agreement, nor is there any type of requirement to destroy the papers after reviewing them. There is simply no mechanism in place to protect trade secrets once a paper is submitted for possible presentation at a conference. This is universally known to those who submit papers.

I declare under penalty of perjury under the laws of the United States that the foregoing is true and correct. Executed this 10th day of June, 2005 in San Francisco, CA.

Dalh chen