

FINAL PUBLIC ORAL EXAMINATION
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

CANDIDATE Clark D. Thompson

TITLE OF DISSERTATION A Complexity Theory for VLSI

DEPARTMENT Computer Science

TIME 9:00 AM, Monday 17 September

PLACE ScH 5409

EXAMINERS H. T. Kung

Jon Bentley

Carver Mead

Michael Shamos

Robert Sproull

A Complexity Theory for VLSI

(Thesis defense)

Clark D. Thompson

ScH 5409

9:00 AM

Monday, Sept. 17

Abstract

The established techniques of computational complexity can be applied to the new problems posed by very large-scale integrated (VLSI) circuits. This thesis develops a "VLSI model of computation" and derives upper and lower bounds on the silicon area and time required to solve the problems of sorting and discrete Fourier transformation. In particular, the area (A) and time (T) taken by any VLSI chip using any algorithm to perform an N point Fourier transform must satisfy $AT^2 > \lfloor N/8 \rfloor^2 \log^2 N$. A more general result for both sorting and Fourier transformation is that $AT^{2x} = \Omega(N^{1+x} \log^{2x} N)$, for all x in the range $0 \leq x \leq 1$. Also, the energy dissipated by a VLSI chip during the solution of either of these problems is at least $\Omega(N^{3/2} \log N)$. The tightness of these bounds is demonstrated by the existence of nearly optimal circuits. This thesis describes both a fast chip ($T = O(\log^3 N)$, $A = O(N^2 / \log^{1/2} N)$) based on the shuffle-exchange interconnection pattern, and a slow chip ($T = O(N^{1/2})$, $A = O(N \log^2 N)$) based on the mesh pattern.

A Complexity Theory for VLSI

(Thesis summary)

C. D. Thompson

14 September 1979

**Carnegie-Mellon University
Computer Science Department**

This research is supported in part by the National Science Foundation under Grant MCS 78-236-76 and a graduate fellowship, and in part by the Office of Naval Research under Contract N00014-76-C-0370.

Table of Contents

1. Introduction	1
2. The VLSI Model of Computation	5
3. Lower Bounds	9
4. Upper bounds	10
5. Contributions of the Thesis	11

List of Figures

Figure 2-1: Tiles.	6
Figure 2-2: Domains of the lower and upper bound models.	7

Abstract

The established techniques of computational complexity can be applied to the new problems posed by very large-scale integrated (VLSI) circuits. This thesis develops a "VLSI model of computation" and derives upper and lower bounds on the silicon area and time required to solve the problems of sorting and discrete Fourier transformation. In particular, the area (A) and time (T) taken by any VLSI chip using any algorithm to perform an N -point Fourier transform must satisfy $AT^2 > [N/8]^2 \log^2 N$. A more general result for both sorting and Fourier transformation is that $AT^{2x} = \Omega(N^{1+x} \log^{2x} N)$, for any x in the range $0 \leq x \leq 1$. Also, the energy dissipated by a VLSI chip during the solution of either of these problems is at least $\Omega(N^{3/2} \log N)$. The tightness of these bounds is demonstrated by the existence of nearly optimal circuits for both sorting and Fourier transformation. This thesis describes both a fast chip ($T = O(\log^3 N)$, $A = O(N^2 / \log^{1/2} N)$) based on the shuffle-exchange interconnection pattern, and a slow chip ($T = O(N^{1/2})$, $A = O(N \log^2 N)$) based on the mesh pattern.

1. Introduction

Very large-scale integrated (VLSI) circuit technology has profoundly changed the size and speed of computing structures. A VLSI microcomputer occupies less than a square centimeter of silicon, yet it outperforms several cubic feet of twenty-year-old computer components. The circuit densities attainable with VLSI are already staggering, and further improvements lie on the horizon. Chips with one hundred thousand transistors are feasible today, and this figure may well increase to ten or twenty million in the next decade [Mead 79].

The number of transistors on a chip is only one measure of its computational power. In fact, it is a rather misleading one, for the organization of the chip's

circuitry will have a strong effect on its size and speed. This thesis explores the relation between the speed and size of VLSI circuits, using the methodology of complexity theory. The first step in this methodology is to devise an accurate and precisely-defined model of a VLSI chip. Theorems can then be proved, on the basis of the model, to show limits on what a chip can and cannot do. A sample of the latter type of result obtained in this thesis is that any chip that performs an N -point Fourier transform in time (T) must have an area (A) large enough to satisfy $AT^2 > \lfloor N/8 \rfloor^2 \log^2 N$. The corresponding positive result shows that this lower bound on the AT^2 product is tight: chips can be built that achieve this limiting performance to within a constant factor.

The use of a new model of computation in this thesis is justified by the novel aspects of VLSI design. A VLSI chip containing millions of transistors will be organized quite differently from a conventional design of the same complexity, for the constraints in the two domains are so dissimilar. First of all, VLSI is a two-dimensional technology. Its transistors are laid out on the surface of a piece of silicon, and there are only a few layers of metal available for interconnections. Conventional designs, on the other hand, make use of the "two-and-a-half dimensions" offered by printed circuit boards connected with backplane wiring.

A second novel VLSI design constraint is the importance of communication costs relative to the costs of logical operations. The available bandwidth across the midline of a VLSI design is quite small in comparison with the amount of processing power available on either side. This fact is a consequence of the planarity of VLSI and the size of transistors and wires. A small transistor can be formed in the area of one square wire-width. A chip that is ω wire-widths on a side has room for ω^2 transistors, but only ω different signals can cross its midline at a time. Such "cross-chip" communications carry a double penalty in VLSI. A long wire uses up valuable chip area that could be used for other wires, and its driver transistor

occupies even more area. These conceptual arguments for the primacy of communication costs are supported by practical experience: more surface area is devoted to wire than to transistors on today's VLSI chips. Scant attention has been paid to this type of communication cost in conventional computers, and rightfully so. The "two-and-a-half dimensions" used in their construction diminishes the cost of cross-system communication. Once a signal is off a circuit board and into the backplane, it doesn't matter whether it goes to a nearby board or to the farthest one.

A third novelty in VLSI design lies in the economics of production. The best conventional designs are the ones with the fewest chips or "active components," for these tend to be the cheapest to build. The cost of interconnecting the components on a circuit board is normally ignored, not because it is inexpensive, but because it increases monotonically with the component count. Thus component count emerges as the single most important metric of cost of a conventional circuit. This is not the case for VLSI designs. The production cost of a VLSI chip grows with the total area of its layout, and a design with smaller area and more transistors is generally preferable to one with a larger area but fewer transistors. That such design choices are available in VLSI chips is one of the messages of this thesis. In any event, it is the area of a VLSI circuit that should be optimized, rather than the number of transistors in it.

The VLSI model of computation developed in this thesis attempts to account for the true costs of computation in this technology. As indicated above, total circuit area is the most accurate metric of circuit size or production cost. The speed at which a circuit operates is also quite important, both from the standpoint of real-time constraints and as a measure of the incremental cost of a computation (if there is a large enough supply of problems, a faster circuit will eventually be cheaper than a slower one, regardless of initial costs). This thesis demonstrates

that production cost can be traded for incremental cost, i. e., that chip area can be decreased at the expense of chip speed, and vice versa.

One of the main results of this thesis is the following area-time tradeoff. The product of chip area (A) with the square of the time (T) it takes to perform an N -point Fourier transform must satisfy $AT^2 > \lfloor N/8 \rfloor^2 \log^2 N$. This lower bound is nearly the best possible, in the sense that there exist both a fast, large chip and a slow, small chip that nearly achieve these bounds. The slow chip performs a Fourier transform in time proportional to $N^{1/2}$, using area proportional to $N \log^2 N$. The optimality of this chip is immediate, since it satisfies the lower bound quoted above to within a constant factor. A fast chip is also possible; it operates in $\log^3 N$ time, but occupies $N^2/\log^{1/2} N$ area.¹

A more general result bounds the performance of any chip with area A that takes time T to solve an N -element sorting or Fourier transformation problem: $AT^{2x} = \Omega(N^{1+x} \log^{2x} N)$, for all x such that $0 \leq x \leq 1$ (that is, $AT^{2x} \geq cN^{1+x} \log^{2x} N$ for some fixed $c > 0$). The lower bound for Fourier transformation cited in the previous paragraph is a special case ($x=1$) of this result.

The general lower bound implies that a chip with performance $A = O(N)$ and $T = O(N^{1/2} \log N)$ would be optimal under any AT^{2x} metric, $0 \leq x \leq 1$. A "slow" chip, based on the mesh interconnection pattern, comes quite close to this performance, solving an N -point sorting problem or an N -point Fourier transformation in $T = O(N^{1/2})$ and $A = O(N \log^2 N)$. A "fast" design also exists for both problems, this one based on the shuffle-exchange interconnection pattern. It can solve an N -point problem in $T = O(\log^3 N)$ and $A = O(N^2/\log^{1/2} N)$, a nearly optimal performance under the AT^2

¹Preparata and Vuillemin [Preparata 79] have recently devised a layout that outperforms the "fast" chip described above. Their layout has an area of $N^2/\log^2 N$ and would take $\log^3 N$ time to compute a Fourier transform, if it were implemented with the multiply-add cells used in the upper bounds of this thesis. Their construction is thus within $\log^2 N$ of the lower bound on AT^2 cost.

metric.

2. The VLSI Model of Computation

A VLSI chip composed of transistors and interconnections is modelled by nodes and wires. A node represents a transistor or a small cluster of transistors; as such, it receives and transmits signals over its connecting wires. A node may also represent a wire junction, in which case it merely copies the signals it receives on any one of its wires onto its other wires. Nodes and wires thus simulate the actions of transistors and wires on a VLSI chip at a very high level of detail.

Nodes are capable of storing a limited amount of information. This enables them to model data storage elements on a VLSI chip. It also allows a collection of nodes and wires to be a complete, self-contained computing structure. The inputs to a computation are stored in a distinguished set of nodes called source nodes. (These correspond to the "input registers" on a VLSI chip.) The output values of a computation are collected in another set of nodes called sink nodes (the "output registers" on a chip). A collection of nodes and wires capable of solving a problem is called a communication graph. A communication graph is thus the formal analog of a VLSI chip.

In order to prove meaningful lower bounds, the VLSI model of computation must be complete, in the sense that a communication graph must exist for every VLSI chip. Otherwise, a negative result for communication graphs would not imply a corresponding negative result (non-existence proof) for VLSI chips: the VLSI chip whose existence disproves the result might not correspond to any communication graph. Chapter 2.2 of the thesis gives a correspondence scheme by which a communication graph can be drawn for every VLSI chip. In brief, a grid of "unit squares" is drawn on the surface of the chip. Each square is small enough that it holds at most one transistor, and at most one wire crosses each unit-length edge.

Nodes are drawn as points in the center of each transistor, and a wire is drawn in the communication graph for each wire on the chip.

Another requirement of the VLSI model of computation is that it be accurate: that the area and time performance of a VLSI chip be predictable from the area and time performance of its communication graph. To this end, the following scheme of area and time costs is proposed for communication graphs.

Both nodes and wires have unit width. A node occupies a unit square, and only one wire can cross each edge of a unit square. Since interconnections can cross over each other to a limited extent on a VLSI chip, two wires of a communication graph are allowed to cross over each other in any unit square. A communication graph can thus be visualized as a collection of the "tiles" of Figure 2-1. This figure shows all possible wire tiles, and one of the many different node tiles. (The node in the last tile has two incoming and one outgoing wires; it could compute the "and" function of the signals it receives on the incoming wires, sending the result out on the outgoing wire.) Each node can have up to four wires. This limitation on node degree is not unreasonable, for the level of detail in the communication graph/VLSI chip correspondence is such that a node is drawn for each transistor or wire junction. Since a transistor has only three connections, four wires per node are certainly sufficient to model any VLSI circuit.

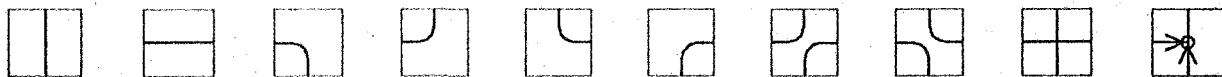


Figure 2-1: Tiles.

The unit of time for a communication graph is obtained from the maximum

bandwidth attained by any wire on the VLSI chip being modelled. One unit of time is thus just sufficient to send one bit of information down a wire.

The total area of a communication graph is the number of squares occupied by its wires and nodes. The total time taken by a communication graph to solve its problem is the number of time units that elapse between its initial state, in which only the source nodes have any information about the problem inputs, to its final state, in which the sink nodes have collected enough information to determine the problem output values.

The astute reader may have noticed that the VLSI model described above is not suitable for proving upper bounds. There is no assurance that a VLSI chip of comparable area and time performance can be derived from any communication graph (the correspondence needed for lower bounds is the reverse one, from chips to graphs). In fact, no such correspondence exists unless a set of restrictions is placed upon communication graphs. A communication graph obeying these restrictions is called admissible, and corresponds to a feasible MOS chip. Admissible communication graphs are used to prove upper bounds in the VLSI model of computation.

The applicability of the lower and upper bound models of computation is summarized by the Venn diagram of Figure 2-2. The universe being studied is that of "all computational structures" that fit in area A and solve an N -point problem P in time T . A (possibly empty) set of communication graphs achieving this area-time performance may be constructed in accordance with the definitions of the lower bound model of computation (see Section 2.1 of the thesis). This set is denoted as " (A,T,P,N) -communication graphs."

The correspondence scheme described in Section 2.2 of the thesis constructively demonstrates that a communication graph can be obtained from any VLSI chip. Thus

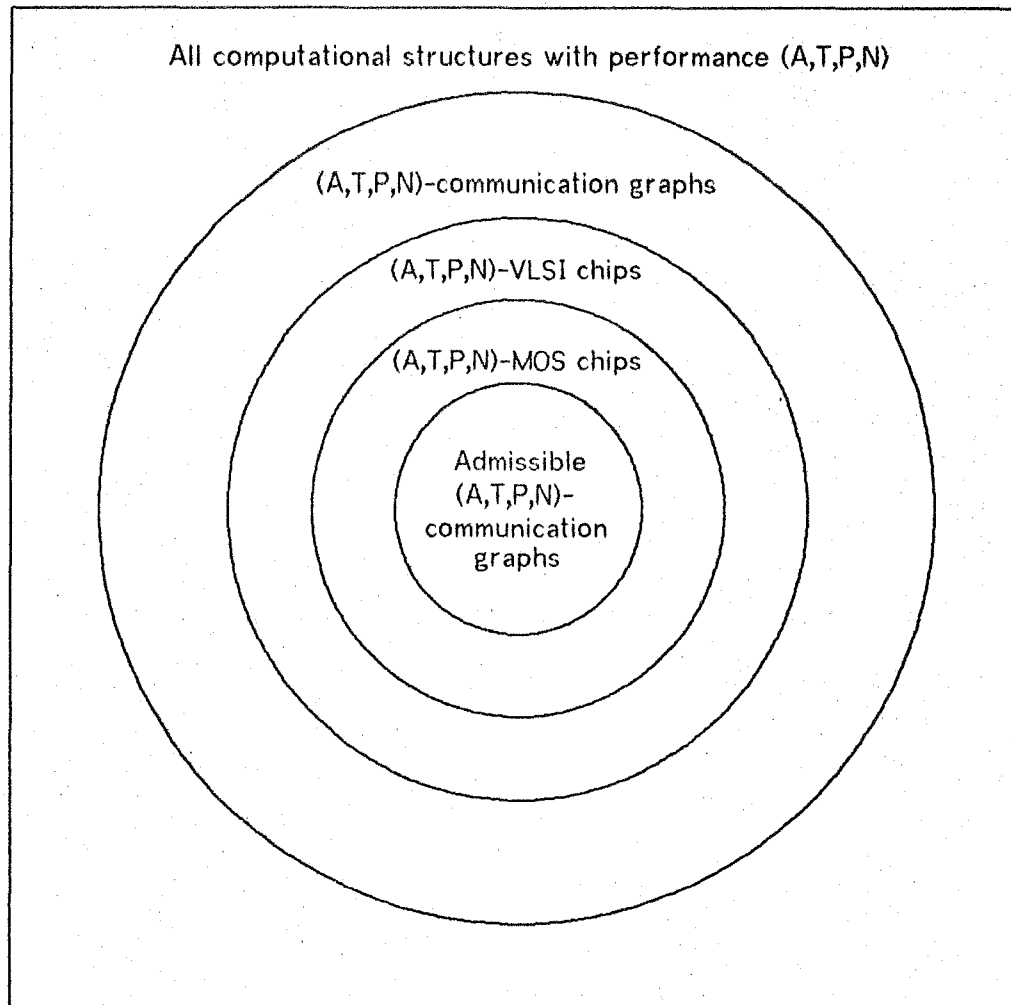


Figure 2-2: Domains of the lower and upper bound models.

the set of " (A,T,P,N) -VLSI chips" (actually, the set of (A,T,P,N) -communication graphs corresponding to VLSI chips) is a subset of all (A,T,P,N) -communication graphs. The generalized MOS process adopted for upper bound proofs is of course only one way of building VLSI chips, so that " (A,T,P,N) -MOS chips" is a proper subset of " (A,T,P,N) -VLSI chips." Finally, the class of "admissible (A,T,P,N) -communication

graphs" defined by the upper bound model of computation (see Section 2.3 of the thesis) form a subset of "(A,T,P,N)-MOS chips."

3. Lower Bounds

The lower bounds of this thesis are all obtained from the analytical technique of "bisecting" a computation. Consider any communication graph (or indeed, any other representation of a VLSI circuit) that is able to evaluate a given function. Now cut enough wires so that it falls apart into two pieces, each containing half of the source nodes (or input registers). The severed wires are said to bisect the graph; each graph has some minimum bisection width which is the smallest number of cuts needed to bisect it. The thesis contains a proof that the area (A) of any communication graph is bounded by $A \geq \omega^2/4$, where ω is its minimum bisection width.

In general, a bisected communication graph will be unable to perform its computation, since each side of the graph has access to only half of the input values. Some information about these input values must cross the bisection; the amount of necessary information flow across any bisection of a function f is called its informational complexity, $H(f)$. The thesis evaluates the informational complexity of the N -point Fourier transformation and sorting problems, proving that these have $H(f) = \Omega(N \log N)$.

A lower bound on the time required to compute a function f on a communication graph may be obtained from $H(f)$ and ω , that is, from the informational complexity of the function and the minimum bisection width of the graph. The available bandwidth across the minimum bisection of a graph is proportional to ω , the number of wires crossing it. At least $H(f)$ bits must cross this bisection, by the definition of informational complexity. Thus the time T to compute f on a graph of width ω is bounded by $T = \Omega(H(f)/\omega)$.

The area result mentioned above, $A \geq \omega^2/4$, and the time result $T = \Omega(H(f)/\omega)$ combine immediately to form $AT^2 = \Omega(H(f)^2)$. For sorting and Fourier transformation, $H(f) = \Omega(N \log N)$, yielding $AT^2 = \Omega(N^2 \log^2 N)$.

Since there are N source nodes (input registers) in a graph computing an N -input function, the area bound $A \geq N$ is immediate. Coupling $A = \Omega(N + \omega^2)$ with $T = \Omega(N \log N)/\omega$, and choosing the ω that minimizes the product AT^{2x} , the general lower bound $AT^{2x} = \Omega(N^{1+x} \log^{2x} N)$ may be obtained for any x in the range $0 \leq x \leq 1$.

4. Upper bounds

The upper bound constructions of the thesis are implementations of the bitonic sort and the fast Fourier transform (FFT) algorithms. Highly parallel versions of these algorithms have been developed elsewhere [Batcher 68, Stevens 71, Thompson 77] for mesh-connected and shuffle-exchange computers. These versions solve an N -point problem on a parallel computer with N processing elements, and can be used as the basis for VLSI implementations.

There are two novel aspects to the upper bound constructions of this thesis. First of all, the shuffle-exchange interconnection pattern must be laid out on the surface of a VLSI chip. This is proved to require at least $\Omega(N^2/\log^2 N)$ area, and an explicit layout is shown that requires only $O(N^2/\log^{1/2} N)$ area. These are the best, and indeed the only reported results for the problem of embedding the shuffle-exchange graph.² The second contribution of the upper bound proofs is a precise characterization of the VLSI area, time, and functionality of the processing elements used in a parallel implementation of an FFT or a bitonic sort.

²Quite recently, Dan Hoey and Charles Leiserson have obtained an $O(N^2/\log N)$ area embedding of the shuffle-exchange graph.

5. Contributions of the Thesis

The main contributions of this thesis fall into four areas.

1. A new model of computation is developed, suited to the study of the area and time performance of VLSI chips.
2. A lower bound is obtained on the area (A) occupied by a graph when embedded in the plane, in terms of its minimum bisection width ω . For a k -level planar embedding, $A > \omega^2/4k^2$.
3. The informational complexity of a function is defined, determining the difficulty of computing that function on a VLSI chip.
4. Nearly tight upper and lower bounds are derived on achievable area-time performance for sorting and Fourier transformation, as summarized in the list and table below.
 - An N -element sorting or Fourier transformation problem can be solved on a chip of area $A = O(N \log^2 N)$ and time $T = O(N^{1/2})$, using a mesh-based interconnection scheme. This performance is optimal under the AT^2 metric, and it is near-optimal for any AT^{2x} metric, $0 \leq x \leq 1$.
 - An N -element sorting or Fourier transformation problem can be solved on a chip of area $A = O(N^2/\log^{1/2} N)$ and time $T = O(\log^3 N)$, using an interconnection scheme based on the shuffle-exchange graph. This performance is nearly optimal under the AT^2 metric.

	<u>AT^2</u>	<u>AT^{2x}</u>
<u>Upper bounds</u>		
Mesh-based designs	$O(N^2 \log^2 N)$	$O(N^{1+x} \log^2 N)$
Shuffle-exchange designs	$O(N^2 \log^{11/2} N)$	$O(N^{1+x} \log^{(6x-1)/2} N)$
<u>Lower bounds</u>		
(For any design)	$\Omega(N^2 \log^2 N)$	$\Omega(N^{1+x} \log^{2x} N)$

References

[Batcher 68]

Batcher, K. E.
Sorting networks and their applications.
In *AFIPS Spring Joint Computer Conference*, pages 307-314. ,
1968.

[Mead 79]

Mead, Carver A. and Conway, Lynn A.
Introduction to VLSI Systems.
in preparation, 1979.

[Preparata 79]

Preparata, Franco P. and Vuillemin, Jean.
*The Cube-Connected Cycles: A Versatile Network for Parallel
Computation*.
Technical Report 356, Institut de Recherche d'Informatique et
d'Automatique, June, 1979.

[Stevens 71]

Stevens, J. E.
A Fast Fourier Transform Subroutine for Illiac IV.
Technical Report, Center for Advanced Computation, Illinois,
1971.

[Thompson 77]

Thompson, C. D. AND KUNG, H.T.
Sorting on a mesh-connected parallel computer.
Communications of the ACM 20(4):263-271, April, 1977.