Parameterized Buffer Cells Integrated Into
An Automated Layout System

Eby Friedman, Wayne Marking, Elaine Iodice,
and Scott Powell
Hughes Aircraft Company
6155 El Camino Real
Carlsbad, California 92008

I) INTRODUCTION

As VLSI design systems become computationally
more efficient and as their ability to generate
chips which provide system level behavior in a
dense and accurate manner improve, it will be
imperative for VLSI chip designs to be generated
quickly with less designer interactivity. The
trend toward true silicon compilation, the
behavioral definition directly implementing
complete and accurate final mask geometries, has
been accelerating dramatically. Major portions
of currently designed VLSI circuits are
automatically routed using powerful,
hierarchical, automated layout systems. The
ability to generate final geometric structures
based on a precise high level description is
currently available for specific problems [1].
Cell compiler systems for generating
Programmable Logical Arrays (PLAs), Read Only
Memories (ROMs), and Finite State Machines
(FSMs) are becoming increasingly commonplace.
In order to generalize the ability to implement
final geometries from behavioral descriptions in
an optimal way, each functional cell must be
automatically designed using well defined,
constraining, circuit parameters. In addition,
to ensure an optimal high performance silicon
compilation capability, mechanisms must be
provided which can automatically parameterize
the functional cells based on application
specific feedback [2].

This paper describes in detail the feedback
mechanism for automatically creating a
parameterized buffer cell from layout extracted
parasitics. In particular, the utility of feedback
mechanisms to silicon compilers will be
emphasized. Section II of this paper describes
the general requirements for providing feedback
mechanisms in silicon compilers between
preliminary extracted information and final cell
geometries. Section III describes the Parasitic
Extractor program, used in generating accurate,
non-statistical, capacitive and resistive
impedances for each automatically generated net.
In section IV, the parameterized buffer subcell
and the cell compiler system are discussed. In
section V, the feedback mechanism between the
extracted parasitic loads and the parameterized
buffer cell system is described. Finally,
section VI contains some concluding remarks
regarding the results described within this paper.

II) THE NEED FOR FEEDBACK MECHANISMS IN
SILICON COMPILERS

The use of sophisticated, hierarchically
controllable, automated layout systems is
becoming increasingly common within the
industry. Their utility has been most
pronounced in the similar requirements of
automated layout of gate arrays and standard
cells. In addition to these design approaches,
the ability to optimally place and route
parameterized cells in an architecturally
unconstrained manner has significant
applicability to the generalized requirements of
a silicon compiler. As large functional blocks
are automatically laid out, significant
quantities of previously undetermined
interconnect are generated. It is well known
and accepted in the design of small geometry
VLSI circuits that a significant portion of the
RC delay of a given signal path is due to this
cell-to-cell interconnect [3]. Therefore, it is
imperative that these capacitive and resistive
loads be calculated accurately and efficiently
and incorporated into the performance analyses
of the VLSI design process. In order for
silicon compilers to achieve the performance
requirements necessary to permit these systems
to become a truly viable VLSI design option,
these parasitic interconnect loads must be
automatically extracted and utilized as feedback
in the design of the various functional cells.

In addition to describing this particular
feedback mechanism necessary in an optimal
silicon compiler, this paper will describe the
general requirements of feedback in silicon
compilation [2]. In summary, three phases must
occur when providing feedback in silicon
compilers. The first is the extraction of
significant data from preliminary constraints
(e.g. layout parasitics, functional
organization, cell placement, signal flow, etc.),
the second is the transformation of that
information into a useful format (e.g. a look-up
table, an expert system, algorithmic
relationships, etc.), and the third is the
insertion of these enhanced parameters into a compatible system so as to constrain the design process in an optimal way (e.g. device geometries, cell aspect ratios, I/O locations, etc.).

III) PARASITIC EXTRACTION OF AUTOMATED LAYOUT

Our present in-house VLSI design system [4] performs physical synthesis in a variety of ways, ranging from the automated layout of standard cells to the automatic generation of programmable logic arrays [5]. In an effort to accurately quantify the parasitic interconnect impedances generated from the automated layout of standard cells, a specific module, the Parasitic Extractor, was written. This program automatically derives, in a precise non-statistical manner, the capacitive and resistive parasitic impedances inherent to automated, multi-layer wiring.

The Parasitic Extractor uses the actual physical layout generated by our in-house automated layout system, HAL (Hughes Automated Layout) [6-7], to automatically calculate the parasitic resistive and capacitive impedances associated with each HAL generated net. As depicted in figure 1, all layout dependent loads such as line-to-ground, line-to-line, and crossover capacitances, feedthrough resistance and capacitance, equivalent pin resistance and capacitance, line resistance, and contact resistance are calculated. In addition, the program accurately derives the effect of a given net's distance to its next nearest neighbor on line-to-line capacitance. At specific intervals along the horizontal segments of the net, the number of empty neighbor tracks to either side of the net being analyzed are counted and a next nearest neighbor factor is used to modify the line-to-line capacitance over this interval.

Once all the resistive and capacitive parasitic components of a given set of nets have been generated, they are tabulated and presented to the design system for use in performance analysis and are used to constrain other portions of the physical synthesis process. Since this program is technology independent, it has general applicability to a varied set of technologies. The program uses a special technology dependent file built to characterize the parasitic interconnect impedances of a given technology.

IV) PARAMETERIZED BUFFER CELLS

One of the most common cell types found in integrated circuits are simple inverters. In addition to providing signal inversion, they are most often used as the simplest and most compact way to provide a particular current drive. The resistive and capacitive parasitic interconnect impedances coupled with the fanout and fanin cell dependent impedances define the required load that must be driven by an optimally sized buffer cell. Therefore, the on-chip performance, specified by system requirements, defines precisely the current necessary to drive the signal interconnect load in the required amount of time. Depending upon the signal loads being driven, a certain sized buffer cell will minimize the overall path delay. The literature [8-9] describes in detail optimal inverter configurations and width ratios which minimize propagation delay through a series of cascaded inverters. However, for a single stage and a

![Figure 1. Extracted Parasitics for Each Net](image-url)
given load, only the application specific requirements determine the precise device width necessary to minimize path delay. Therefore, the ability to automatically define a buffer cell as a function of its current drive has important applications to large VLSI circuits where many buffers exist to provide optimal current drive.

In a simple CMOS inverter, the geometric widths of the P- and N-channel devices can be used to define or parameterize the buffer current drive. Other than the transistor channel lengths which are typically kept at their minimum for maximum cell density, the channel width of a device is the only significant layout dependent parameter capable of accurately controlling current drive. In addition, since the ratio of P-channel on-resistance to N-channel on-resistance is constant for a particular process technology, the P-channel and N-channel device widths are directly related. Therefore, the current drive of a buffer cell can be optimized by using a single parameter, the P-channel geometric width.

Figure 2 depicts the layout of the required CMOS subcells making up the parameterized buffer cell system. For any given P-channel width, the same input and output subcells are used. As shown, each transistor subcell contains two polysilicon stripes to maximize current density per area. The power and ground buses as well as the common P- and N-channel drains are shared by the parallel transistors. Polysilicon has been placed on both the top and bottom of each of the transistor cells for maximum flexibility in automated layout. The initial and final N-channel and P-channel polysilicon gates are connected by both the input subcell and the output subcell to minimize polysilicon resistance.

The cell height as well as the power and ground buses are of standard dimension so as to be compatible with the cell library of the buffer cell. Every parameterized buffer cell is made up of three significantly different subcells. Butted to the left and right portions of the buffer cell are the input and output subcells, respectively. A cascaded series of transistor subcells is placed between these two subcells. The final inverter is a parallel combination of transistor subcells. For a given set of technology dependent design rules and a standard cell height, layout dependent maximum and minimum transistor widths are defined. Within this range of P-channel transistor widths, the additional required transistor subcells have been designed. Depending upon the required P-channel width, the parameterized transistor subcells together and places them between the input and output subcells. A flow diagram of the system for generating parameterized buffer cells from extracted interconnect parasitics is shown in figure 3.

In addition to the creation of the final geometric artwork, as depicted in figure 3, a cell library description of the buffer cell is generated defining the size, pin locations, pin types, and pin equalities for use in HAL. Therefore, for any given P-channel device width, a buffer cell is quickly and optimally designed in mask defined geometries as well as for

Figure 3. Flow Diagram of Parameterized Buffer Cell System
immediate use in automated layout. Also, if the buffer cell transistor width exceeds a certain technology dependent dimension, a second buffer cell, of a pre-defined, ratioed, geometric width is created and butted in front of the original buffer cell. The addition of this cell ensures that the buffer cell does not excessively load down its previous stage and provides a minimum propagation delay for the signal path [8-9].

V) FEEDBACK BETWEEN THE PARASITIC EXTRACTOR AND THE PARAMETERIZED BUFFER CELL SYSTEM

As depicted in figure 3, once the Parasitic Extractor has generated interconnect impedance information, it is necessary to transform the loading information into a format compatible with the parameterized buffer cell system. As described in the previous section, the optimal P-channel width of a CMOS inverter is dependent upon two constraints: the signal loading and the performance requirements of the critical signal path. In order to maximize the generality of this design system, a look-up transformation table characterizing a variety of design tradeoffs is necessary. Depending upon the precise performance requirements of a particular circuit application and its off-chip loading, an optimal P-channel device width can be chosen to parameterize the buffer cells which drive the critical signal nets.

This look-up transformation table is characterized by different power/speed/area tradeoffs as a function of signal net loading and system performance requirements. A look-up transformation table was selected as the most appropriate technique for the parasitic impedance feedback mechanism due to its ease of implementation. It must be emphasized, however, that for different feedback mechanisms, different transformation techniques are preferable.

VI) CONCLUSIONS

As the need for more efficient VLSI-oriented silicon compilers becomes apparent, it will be increasingly important to extract useful information from the design process and transform this information into an appropriate format which will optimize the synthesis process. Depending upon the feedback mechanism, various transformation techniques are preferable when optimizing the physical implementation process. The appropriate transformation technique is dependent upon the particular extracted information and the resulting cell function which utilizes the information.

Finally, the specific feedback mechanism for extracting accurate layout dependent interconnect parasitics and the use of this information to choose an optimal device width for a parameterized CMOS inverter is described. The immediate benefits of utilizing extracted interconnect parasitics to automatically parameterize buffer cells on circuit density, performance, and power dissipation depict only a portion of the value of providing feedback in silicon compilation.

VII) ACKNOWLEDGMENTS

The authors would like to acknowledge G. Kramer for his technical assistance and M. Carchidi for his help in proofreading this paper.

VIII) REFERENCES