Seminar Material For Graduate Students *2001/11/30*

Recent Topics on Programmable Logic Array

Department of Electronics Engineering, Asada Lab. *M1, 16762, Ulkuhan Ekinciel*

Abstract: *The programmable logic array (PLA) is a basic and important building circuit for VLSI chips. After recent improvement techniques on PLA, PLA becomes more attractive for designers especially in GHz technology with deep sub-micron sizing. In this paper we will try to introduce the cross-talk problem solution in DSM, in PLA Network and also try to give the usage of PLA in GHz technology and the advantage of PLA as a solution of Timing Closure.*

1-Introduction

Since their introduction in 1970, programmable logic arrays have been used in large-scale integration and very large-scale integration (LSI/VLSI) chips. Because its regularity and programmability. The programmable logic array is an old [1] but important building block in digital VLSIs.

Continued advances in VLSI technology, along with the development of more sophisticated CAD tools, enable an increase in the level of integration of silicon chips. By integrating more and more circuits on a single chip, system performance improvements can be achieved. However the number of the elements in a chip increases significantly. Although the recent development of CAD tools offers more complicated VLSI designs, it doesn't keep up with the rapid increase integration [2]. Therefore, chips will be limited to a certain size in the near future, even if more chip area is available. In order to enhance the design productivity without degradation of system performance, design styles having regular structure such as programmable logic arrays (PLAs) and circuit techniques for higher performance, even though with the drawback of the large area overhead, necessary [10].

The problem of designing VLSI systems is extremely complex. This complexity can be partially simplified by using PLAs. PLAs provide a regular structure and are attractive to VLSI designers because they require a small number of separate cell designs. They allow for ease of testing while offering simple, rapid expandability [3]. The regular array structure also makes the PLA's timing predictable and controllable, which is quite an attractive feature for the modern GHz circuit design [4].

The other advantage of the PLA is that: In certain logic functions such as the control units of VLSI processors are difficult to implement by random logic. Since the programmable logic arrays (PLAs) can implement almost any Boolean function, they have become popular devices in the realization of both combinational and sequential circuits [5].

However with the decreasing minimum feature size of VLSI fabrication processes, some problems are becoming increasingly common. These problems are like that; cross-talk, electromigration, self-heat and statistical process variations.

The increasing importance of the above electrical effects requires that designers consider the interaction between logical and physical design at the same time. This makes the design task more complex and time consuming. The cross-talk problem is perhaps the most important effect, which places in danger the ability of designers to abstract the logical and physical aspects of design.

Many works have been devoted to the circuit design of the PLA [5]-[9] including recent ones [10]-[16].

This paper is organized as follows. The general concepts of PLA will be reviewed in section II. In section III and IV recent investigated papers on PLA will be introduced as the order of, the usage of PLA in control logic implementation and in timing closure methods, and crosstalk-immune design with PLA network. In section V the content of this paper will be summarized.

2-Brief review of PLA concepts

A programmable logic array (PLA) maps a set of Boolean functions in canonical, two-level sum-of-product form into a geometrical structure.

A typical PLA consists of two major sections or planes. One is called the "AND" plane and the other is called "OR" plane. The AND plane is responsible for generating the product term while the OR plane sums the required terms.

The inputs to the PLA have to be available in complemented and uncomplemented form. Usually the inputs are also latched. The outputs of the PLA

are also usually latched. A wide variety of design styles are available when implementing a PLA. These include static, pseudo-nmos, clocked, domino, etc. Strategies may also include precharge and differential sensing.

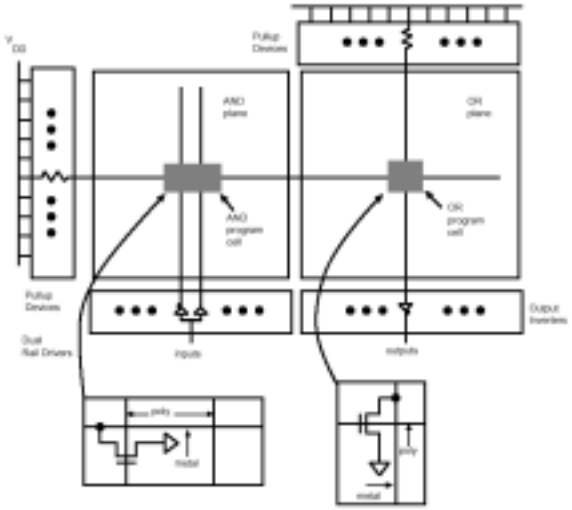


Figure 1: Basic PLA Organization

2.1- ***Design Styles***

2.1.1- Static NMOS and Pseudo-NMOS PLA

NMOS PLA: Pull-up network realized by single NMOS depletion transistor

Pseudo NMOS PLA: Pull-up by high resistance pMOS transistor with permanently grounded gate input.

Since the AND-OR structure is not suited to MOS circuit technology both AND and OR planes are implemented using distributed NOR and NAND gate structures based on deMorgans law:

a) INV-NOR-NOR-INV structure:

$$ab+cd = \overline{(\overline{a}+\overline{b})} + \overline{(\overline{c}+\overline{d})}$$

$$\underbrace{\overline{\underbrace{(\overline{(\overline{a}+\overline{b})} + \overline{(\overline{c}+\overline{d})})}_{\text{NOR}}}}_{\text{INV}}$$

Equation: 1

Properties:

- High static power dissipation
- Small area
- Useful if high speed is not required

b) NAND-NAND structure:

$$ab+cd = \overline{\overline{ab}+\overline{cd}} = \overline{(ab)\,(cd)}$$

Equation: 2

Properties:

- NAND-NAND approach not recommended
- Decreasing performance at increasing number of the inputs (because of series connection of nMOS transistors)
- High static power dissipation

2.1.2- Static CMOS PLA

NOR gates with a large number of inputs should be avoided in CMOS, because devices are in series. Static CMOS PLA are usually realized with NAND-INV-INV-NAND structure in order to avoid long chains of pMOS transistors.

Properties:

- No static power dissipation
- Area increase becomes unacceptable for large PLA's
- Working fast

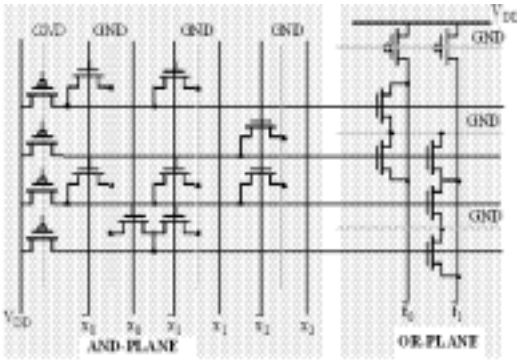


Figure 2: Pseudo-Static PLA

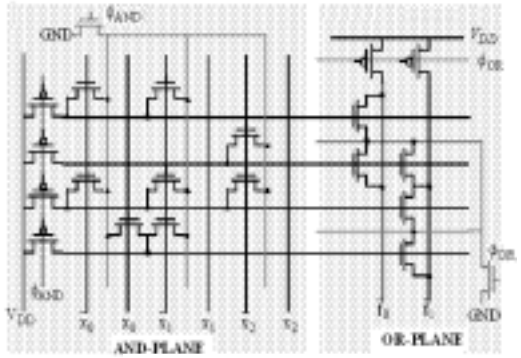


Figure 3: Dynamic PLA

2.1.3- Dynamic CMOS PLA

Properties:

- Less size than static CMOS
- Fast
- 2-phase clocking
- States of $Ø_{AND}$:

$Ø_{AND}$=1:

- No path to ground
- Inputs change
- Both NOR planes are precharged

$Ø_{AND}$=0:

- First NOR plane discharges
- After first NOR plane, the second NOR plane evaluates

- $Ø_{OR}$ is used to latch the second stage and intermediate clock is required to precharge OR plane.

2.2- Noise in PLA

- In dynamic PLAs noise problems on switched supply lines
- Discharging current is generating in the power supply bus
- To reduce noise: Locally grounding the PLA; use of the metal lines for power supply whenever possible (reduced impedance)

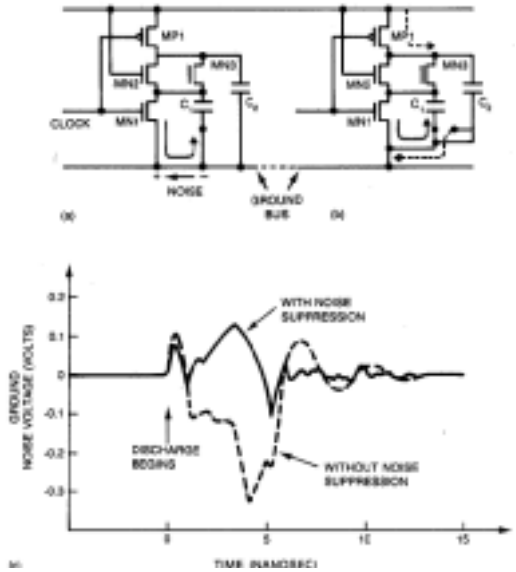


Figure 4: Noise Problem in Dynamic PLA's

2.3- Optimization of PLA

Here are a few basic optimization techniques. In Section 3 and 4 I will introduce some more sophisticated optimization techniques.

2.3.1-Logic Minimization

- Optimizations (minimization) of Boolean equations in order to reduce the number of minterms or literals
- If a term is needed both positive and negative sometimes a reduction can be achieved by using negative logic
- Decoder in front of the AND plane to generate combined input variables

2.3.2-Multiple Side Access & Folding

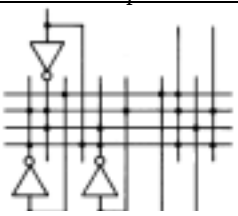


Figure 5: Multiple sided input/output access

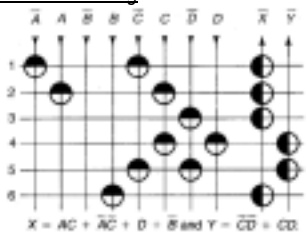
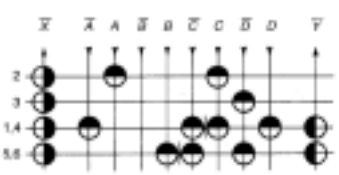


Figure 6: PLA before



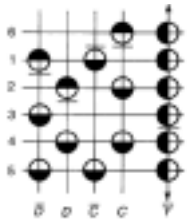


Figure 7: Row-folded PLA Figure 8: Column-folded PLA

An advantage of multiple-sided access and folding is the decreased layout area, but the layout structure has changed and the wiring becomes more difficult.

2.4- PLA Generation

PLA generation is a process by which a set of input signals combines, through a logical sum of products, to form a set of output signals.

Due to the regular nature of the AND and OR cells which are used to implement PLAs, the entire generation process may be automated with little to no loss in efficient space usage. A PLA generator uses as input files which describe the desired functionality. The generator then uses these input files to draw the appropriate schematic as well as layout the AND and OR cells in the necessary order to achieve the necessary functions.

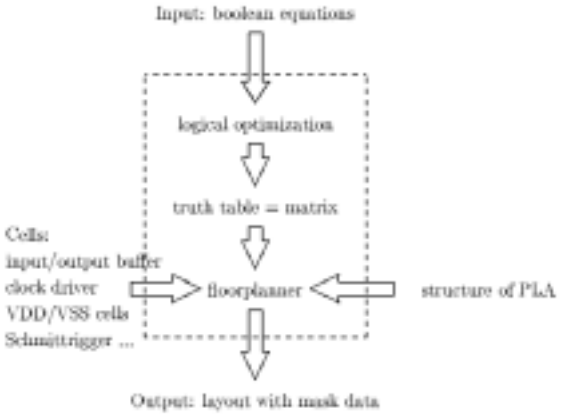


Figure 9: Automatic PLA layout generation

3-*Control Logic Implementation and Timing Closure with PLA [13], [4]*

The growing performance requirements make the use of dynamic circuits in selected places desirable.

Potential timing closure problems due to random control logic can be avoided through the use of structured arrays with predictable electrical characteristics. Array structures are an ideal place to use dynamic circuits. Timing can be very structured, path length is very well known, and physical dimensions are fixed [19].

There are some favorite recent designs; using PLA structured circuits for overcoming the problem of timing closure and logic implementation for control circuits, such as IBM's Star PowerPC microprocessors [4], [9], [13]-[16]. I will try to introduce two of them in this section.

3.1 "Timing Closure by Design", A High Frequency Microprocessor Design Methodology

Timing closure for large microprocessor designs is becoming more and more difficult as

- Chip complexity increases
- Cross-chip wire delays become more significant
- Dynamic Circuits become more prevalent
- Cycle times shorten

Just as 'Correct by Construction' techniques reduce introduction of layout errors in chip designs, 'Timing Closure by Design' techniques reduce the introduction of timing problems. These timing problems can not be typically found or fixed until late in the design process making it difficult to meet required frequency targets. The ' Timing Closure by Design' methodology has the goals of

- ✓ Achieving the highest possible processor frequency
- ✓ Reducing the design time to achieve that desired frequency.

The main themes of this methodology are:

- ✓ Early timing planning with an eye towards the physical implementation,
- ✓ Using components and design techniques with predictable timing characteristics.

This methodology was used to design an experimental 19 million transistor PowerPC microprocessor that was designed to operate at 1.0 Gigahertz (1.62V, 85 ° C) [14]. A previous 1.0 Gigahertz integer processor [4], [15], [16] was built by using many of the same concepts.

Characteristics of " Timing Closure by Design" are:

- Logic partitioned on timing boundaries
- Predictable Control Structures (PLAs)
- Static Interfaces for Dynamic Circuits
- Low skew clock distribution
- Deterministic Method of macro placement
- Simplified timing analysis
- Refinement method of chip integration with early timing analysis

In " Timing Closure by Design", one of the key points is that using PLA in predictable implementation of dataflow and control circuits.

Using structured circuit and layout approaches can eliminate some of the timing uncertainty in the macro design process. For example, dynamic PLA and comparator structures were used for all of control logic circuit in [4]. The dynamic PLA provides

i. High frequency operation
ii. Quick logic personalization
iii. Predictable area and delay
iv. Early recognition of excess logic for one cycle.

As compared to a standard cell approach, no heuristic logic synthesis or auto placement is required with PLAs or comparators. Those tools may require many runs and adjustments to input parameters to achieve timing requirements, as well as inject uncertainly late in the design process when last minute logic changes are necessary. The PLAs have exclusive latch drivers (one fanout), which are placed adjacent to the PLAs, minimizing input wire delays and consequently input skews. Due to the high performance of the PLAs (300-470ps delay), a single level static or dynamic gate can be connected to the PLA outputs, increasing functionality without significantly adding delay uncertainty. Figure 10 illustrates a template for interfacing different configurations of control PLAs, compare macros and individual gates.

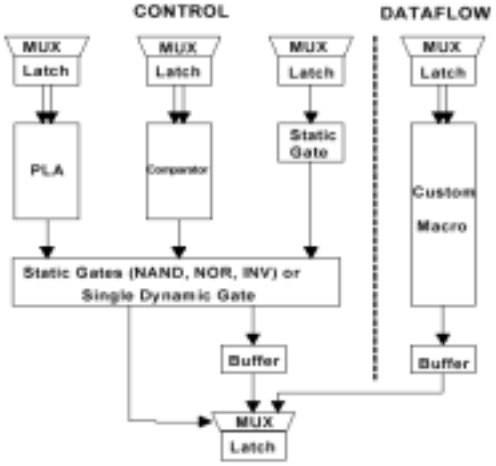


Figure 10:Control Template [13]

3.2 **'Design Methodology for a 1.0 GHz Microprocessor'**

The IBM Gigahertz processor utilized PLAs to implement control logic. The stated reasons for this choice were high speed and the ability to be quickly implemented and this modifies the design.

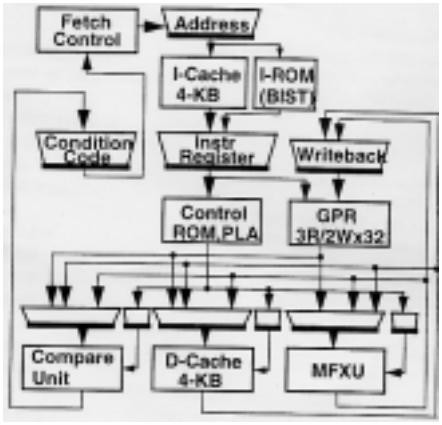


Figure 11: Processor Organization [4]

Figure 11 shows the Processor Organization. As much control logic as possible is combined with in the datapath macros. The rest of the control is partitioned so that it does not need to merge with the datapath other than at the end of the cycle. This in turn allows control logic to use a complete cycle to execute, permitting the use of ROMs and PLAs for control implementation.

Read Only Memories (ROMs) and Programmable Logic Arrays (PLAs)' s macros use dynamic circuits to meet their timing requirements. Both the ROM and PLA layouts are automatically personalized from their VHDL behavioral models. For the PLA, the VHDL is compiling, minimized through Espresso [17], and then passes to the layout personalization code. It should be noticed that the VHDL is carefully crafted to fit into the PLA and ROM structures and partitioned bases on the floorplanning of the macros. Using these structured arrays give predictable areas and delay along with fast turn around of logic to layout. See Table 1 for a comparison of static random-logic macros (RLMs) using standard cells and dynamic structured arrays (ROMs and PLAs). The high speed of the dynamic circuits and timing predictability of using arrays are the compelling reason for choosing these types of the macros for the processor control logic.

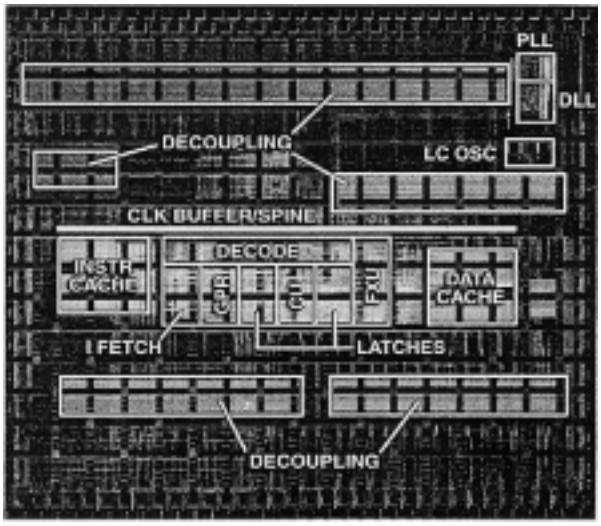


Figure 12: Chip Micrograph [4]

Figure 12 shows a chip micrograph with the major macros. All the chip level macros are hand placed to control and predict wire delays and noise across the chip. These placement-induced are of first-order importance due to the frequency objective of the design. The redrive inverters provide gain; by placing them carefully they also act as filters for dynamic signals thereby improving the chip's noise immunity.

Table 1:Control Logic Implementation Styles [4]

	Static Standard Cell	Dynamic Arrays
Performance	Slow, due to static circuits, variation in path delays	Fast, due to dynamic circuits, all paths have similar delay
Tools	Requires Synthesis, Placement and Routing (heuristic programs) high CPU	Requires Layout generator, not easily migrated to new technologies Low CPU
Area	Depends on logic, performance and tool operation, maybe more or less than Arrays	Predictable, less dependence on actual logic
Capacity	Logic limited by tool capacity	Logic limited by maximum array size (less than standard cell)
Predictability	Unknown delay and area until after routing, may change drastically after small logic change, many iterations of the tools may be required to achieve specifications	ROM has fixed area and delay, PLA area and delay vary slightly with logic changes

4-Crosstalk-Immune Design in PLA Network [8]

In this section, I will introduce a VLSI design methodology to address the cross-talk problem, which is becoming increasingly important in Deep Sub-Micron (DSM) IC design.

4.1 Occurrence and Effects

Cross-talk typically occurs between adjacent wires on the same metal layer, when the cross-coupling capacitance between these wires is large enough for them to affect each other's electrical characteristics. As the minimum feature size of VLSI fabrication process reaches the 0.1um range, process engineers are forced to increase the height of the wires in relationship to their width, in order to keep their sheet resistivity from increasing quadratically. This is in turn increases the cross-coupling capacitance between a wire and its neighbors as a fabrication of its total capacitance, resulting in cross-talk problems. In particular, cross-talk can cause a significant delay variation in a wire depending on the state of neighboring wires. Also, it can cause the logic value of a wire to be incorrectly interpreted depending on the state of neighboring aggressor wires, resulting in a loss of a signal integrity. With the decreasing minimum feature size of VLSI fabrication processes, this problem is becoming increasingly common.

4.2 Solution

For this methodology, the cross-talk problem is eliminated by design. This is done by imposing a fixed pattern of wires on the IC die, on all metal layers. In particular, this repeating pattern henceforth referred to as the Dense Wiring Fabric (DWF) is …VSGSVSGS…, where V represents a VDD wire, G represents a GND wire, and S represents a signal wire. This ensures that adjacent signal wires are always capacitively shielded from each other. Metal wires on any layer run perpendicular to those on layers above and below it. This layout arrangement is shown below.

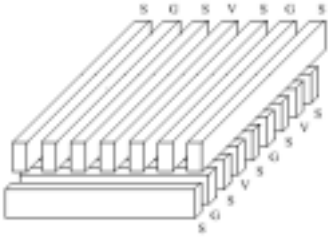


Figure 13: Arrangement of conductors as in [18]

As reported in [18], the use of this fabric has some compelling advantages. First, with this choice of layout fabric, the cross-coupling capacitance between signal wires drops by one to two orders of magnitude, thereby all but eliminating the delay variation and signal integrity problems due to cross-talk. The uniformity of inductive and capacitive parasitic, which results from the regularity of the DWF, is a feature that CAD tools can exploit. Also, by suitably introducing vias whenever VDD or GND wires intersect on adjacent metal layers, a power and ground distribution network of low and highly uniform resistance is created. Finally, the DWF enables us to easily generate a low-skew global clocking network due to the low and predictable parasitic. The main disadvantage of the scheme of [18] was an increased area requirement compared to the standard cell methodology.

In this part a new design flow will be introduced, which retains the best features of the scheme of [18], with an extremely low area penalty, and significant delay improvement compared to a standard cell implementation. In this scheme, a logic network is implemented as a Network of Programmable Logic Arrays (PLAs). With a network of PLAs, there is a more direct relationship between the cost function being optimized for during synthesis, and the actual PLA implementation, since there is no inverting technology mapping step. As a result, multi-level logic synthesis is tightly coupled with logic implementation in design flow.

4.3 Approach

In this scheme, the circuit as a network of PLAs is implemented. Each PLA is a multi-output structure; lays out in a crosstalk-immune manner. Each PLA implements its logic functionality with high density and speed. The routing region between PLAs is organized using the DWF, giving rise to highly predictable, crosstalk-immune routes. Metal layers that are not utilized in the layout of the PLA are gridded maximally throughout the die, using the technique of [18]. Also intersecting VDD or GND wires on adjacent layers are connected by vias. This gives rise to a highly efficient power and ground distribution network throughout the die. When PLAs are placed, local breaks occur in the power and ground gridding structure of Metal1 and Metal2.

4.4 PLAs in DSM VLSI Design

The pre-charged NOR-NOR PLA is used in this design. The schematic view of the PLA core is shown in Figure 14.

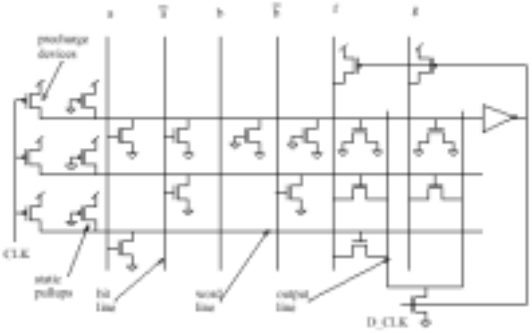


Figure 14: Schematic view of the PLA core [8]

By using a pre-charged NOR-NOR PLA as the layout building block, no extra area penalty is incurred, either in the horizontal and vertical direction. At the same time, the PLA structure is crosstalk immune, which makes it an ideal choice. Figure 15 shows the layout of the PLA core (implemented using two metal layers). The horizontal word lines are implements in METAL2. The width of the PLA core is 4.n+2.m tracks, since the each input requires 4 vertical tracks, and each output requires 2.

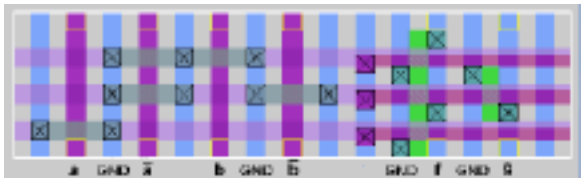


Figure 15: Layout of PLA core [8]

Fgure16 shows the relative orientation of the pre-charge devices, muxes and drivers in the layout of each PLA.

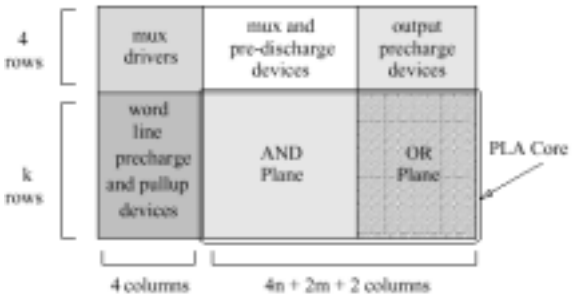


Figure 16: Layout Floorplan of PLA [8]

Due to the regularity of the PLA structure, a simple delay formula can be used to estimate the worst-case delay of PLA.

4.5 PLA Characterization

Figure 17 shows the pattern of wires occurring within the core of these PLAs.

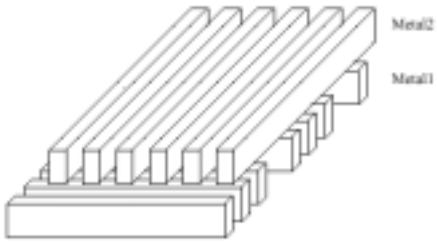


Figure 17: Arrangement of conductors in the PLA core [8]

The reasons why PLAs result in very favorable area and delay characteristics compared to a standard cell layout are the following:

i. First, PLAs implement their logic function in 2-level form, which results in superior delay characteristics as long as k is bounded. (k is the number of PLA in the PLA Network.) On the other hand, in a standard cell implementation, considerable delay is incurred in traversing the different levels of the design.

ii. In the PLA implementation scheme, local wiring is collapsed into a compact 2 level-core, which is naturally crosstalk immune. Hence local wiring delays are reduced.

iii. In DSM processes, it is often stated that a large part of a signal's delay is attributable to global wiring. Methods to tackle this include up-sizing of drivers, and buffer insertion.

- Sizing of PLA output drivers is easily done with no PLA area penalty. This is because output drivers are placed in the routing area.
- A signal can be buffered in the DWF with no penalty, since the VDD and GND signals required constructing a buffer available on either side of the signal.

iv. Devices in the PLA core are minimum sized, giving rise to extremely compact layouts. Such is not the case for standard cell layouts.

v. In this PLA core, NMOS devices are used exclusively. As a result devices can, be placed extremely close together. However, in a standard cell, both PMOS and NMOS devices are present in each cell, and the PMOS- to-NMOS diffusion spacing requirement results in a loss of layout density.

4.6 Advantages of Presented Method

✓ High Speed. Each PLA is shown to be on average 2.1* faster than its corresponding standard cell based circuit implementation. Also, the network of

PLA is about % 15 faster than the standard cell implementation of the same netlist.

✓ Low area overhead. This scheme has an area overhead of about 3 %. This is in spite of the fact that the DWF is used in the routing area between PLAs, but not used in the standard cell case. Each individual PLA is shown to be 2.17 % smaller than its corresponding standard cell based circuit implementation.

✓ Area overheads and timing improvements are shown to be largely independent of the number of routing layers utilized to route the design.

✓ Elimination of the crosstalk and signal integrity problems that are common in DSM design. The resulting implementation is highly reliable.

✓ Power and ground routing is done implicitly, and not in a separate step in the design methodology. Power and ground resistances are very low and vary much less compared to the power and ground distribution used in the standard cell methodology.

✓ Variations in delay of a signal wire due to switching activity on its neighboring signal wires is less than 1.02:1, compared to a 2.47:1 variation using conventional layout techniques.

✓ Smaller and uniform inductances for all wires on the chip, compared to larger and unpredictable values using the existing layout styles.

✓ Rapid design turn-around time due to highly regular structures and regular parasitic.

✓ With a network of PLAs, there is a direct relationship between the cost function being optimized for during synthesis, and the PLA implementation, since there is no intervening technology mapping step. This helps ensure that benefits due to synthesis optimizations are not lost in the implementation step.

5-Summary

The reasons of 'PLA is becoming an attractive and satisfied choice for high performance and high reliable circuit design' are presented. And several very new papers that are published on Timing Closure and Cross-talked problem, which included the sophisticated solutions with the usage of PLA, were introduced.

References:

[1] C. Mead and L. Conway, Introduction to VLSI Systems. Reading, MA: Addison-Wesley, 1980.

[2] "The International Technology Roadmap for Semiconductors," SIA, Seamatech, Inc., 1999

[3] E. D. Fabricius, Introduction to VLSI Design. New York: Mc-Graw-Hill, 1990

[4] S. Posluszny et al., "Design methodology for a 1 GHz microprocessor," in Proc. Int. Conf. On Computer Design: VLSI in Computers and Processor, 1998, pp. 17-23.

[5] C.C. Wang, C.F. Wu, R.T. Hwang, and C.H. Kao, " A low-power and high-speed dynamic PLA circuit configuration for single-clock CMOS", IEEE Trans. Circuits Syst. I, vol. 46, pp. 857-861, July 1999.

[6] G.M Blair, "PLA design for single-clock CMOS", IEEE J. Solid-State Circuits, vol 27, pp. 1211-1213, Aug. 1992

[7] Y.B Dhong and C.P. Tsang, "High Speed CMOS POS PLA using predischarged OR array and charge sharing AND array", IEEE Trans. Circuits Syst. II, vol. 39, pp. 557-564, Aug. 1992

[8] S.P. Khatri, R. K. Brayton, and A. Sangiovanni-Vincentelli, " Cross-talk Immune VLSI Design using a Network of PLAs Embedded in a Regular Layout Fabric," in Proc. IEEE Int. Conf. Computer-Aided Design, Nov. 2000

[9] O. Takahashi, N. Aoki, J. Silberman, and S. Dhong, "A 1-Ghz logic circuit family with sense amplifiers," IEEE J. Solid-State Circuits, vol.34, no. 5, pp. 616-622, May 1999

[10] H. Yamaoka, M. Ikeda, K. Asada, " A High-Speed PLA using Dynamic Array Logic Circuits with Latch Sense Amplifiers," IEICE Trans. Electron, vol E84-C, no.1, January 2001

[11] J.S. Wang, C.R. Chang, C. Yeh, "Analysis and Design of High-Speed and Low-Power CMOS PLAs" IEEE Journal of Solid-State Circuits, vol. 36, no.8, August 2001

[12] H. Yoshida, M. Ikeda, and K. Asada, " Logic Synthesis for AND-XOR-OR type Sense-Amplifying PLA," IEEE Int. Conf on VLSI Design & Asia South Pacific Design Automation Conf., Jan. 2002 (to appear)

[13] S. Posluszny et al., " Timing Closure by Design" in ACM, ISBN: 1-58113-187-9, pp 712-717, 2000

[14] P. Hofstee, N. Aoki, D. Boerstler, P. Coulman, S. Dhong, B. Flachs, N. Kojima, O. Kwon, K. T. Lee, D. Meltzer, K. Nowka, J. Park, J. Peter, S. Posluszny, M. Shapiro, J. Silberman, O. Takahashi, and B. Weinberger, "A 1 GHz Single-Issue 64b PowerPC Processor," *ISSCC* Digest of Technical Papers, 2000, pp. 9293.

[15] Silberman, J., et al., " A 1.0 GHz Single-Issue 64b PowerPC Integer Processor", ISSCC Digest of Technical Papers, p.230, Feb. 1998

[16] Hofstee, P., et al., "Designing for a Gigahertz", IEEE Micro, May-June 1998

[17] R. Brayton, G.Hachtel, C. McMullen, and A. Sangiovanni-Vincentelli, "Logic Minimization Algorithms for VLSI Synthesis" Kluwer Academic Publishers, Boston, MA 1984.

[18] S. Khatri, A. Mehrotra, R. Brayton, A. Sangiovanni-Vincentelli, and R. Otten, "A novel VLSI layout fabric for deep sub-micron applications, " in Proceedings of the Design Automation Conference, pp. 122-127, June 1999.

[19] http://www.research.ibm.com/ journal.com/ rd446/ allen.html