An Efficient Hybrid LUT/SOP Reconfigurable Architecture

Po-Yang Hsu Ping-Chuan Lu Yi-Yu Liu
Department of Computer Science and Engineering
Yuan Ze University, Chungli, Taiwan 320, R.O.C.
s976024@mail.yzu.edu.tw s966027@mail.yzu.edu.tw yyluu@saturn.yzu.edu.tw

Abstract

With the increasing NRE cost of advanced process technologies, reconfigurable devices receive great attention in small and medium volume IC designs. However, lower logic utilization and slower timing performance limit the efficacy of FPGA and CPLD. In this paper, we propose an efficient hybrid LUT/SOP reconfigurable design style to exploit both the advantages of LUT-cell and SOP-cell for circuit design. After that, architectural evaluations are performed in order to have the best cell mixture ratio. Furthermore, three logic optimization techniques, cell collapsing, phase flipping, and phase duplication, are proposed for hybrid LUT/SOP FPGA synthesis from a pure LUT-based design. The experimental results demonstrate that our proposed hybrid LUT/SOP design style achieves 35% circuit performance improvement and 51% transistor count reduction as compared with the depth optimal 4-LUT-based FPGA.

I. INTRODUCTION

Reconfigurable design styles provide relatively low NRE cost as compared to either the standard cell or the full custom design styles. With the growth of VLSI technology, reconfigurable design styles are widely used for either pre-silicon hardware/software co-verification or for small and medium volume ASIC products. Field programmability enables fast re-spin turn around time and hence speeds up time to market. There are two major categories of reconfigurable devices, the field programmable gate array (FPGA) and the complex programmable logic device (CPLD). The FPGA utilizes lookup tables (LUT) to implement multi-level functions in order to maximize node sharing in a Boolean network. The CPLD employs macro-cells to implement two-level functions in either sum-of-product (SOP) form or product-of-sum (POS) form with maximized product-term sharing in a Boolean network. The drawbacks of the reconfigurable design styles are lower timing performance and lower area utilization as compared with the standard cell design style. Previous research optimizes either circuit delay or circuit area for FPGA and CPLD [1]–[7]. Hybrid FPGA architecture was proposed in last decade [8]. The hybrid FPGA utilizes both LUT-cells and macro-cells in order to take the advantages of FPGA and CPLD. Many research teams suggest different macro-cell configurations for performance and area optimization [8]–[11]. Most of the proposed hybrid FPGA architectures utilize large macro-cells in order to cluster more functions within a macro-cell. In this paper, we first investigate the logic utilization issue of homogeneous reconfigurable logic blocks. According to our profiling results, the logic utilizations of both LUT-cell and macro-cell are poorly low. This motivates us to propose a small reconfigurable cell (SOP-cell) for hybrid FPGA architecture. In order to achieve the best area saving, neither I/O phase complement nor product-term sharing is available in the SOP-cell. After that, architectural evaluations are performed to determine the best mixture ratio of SOP-cell to LUT-cell in each reconfigurable logic block. Furthermore, three logic optimization techniques, cell collapsing, phase flipping, and phase duplication, are proposed for hybrid LUT/SOP synthesis. The experimental results demonstrate that the simple SOP-cell, without phase complement and product-term sharing, is promising for both performance improvement and area reduction.

The rest of this paper is organized as follows. The reconfigurable architecture survey is given in Section II. Section III gives the motivation of this paper. In Section IV, we propose our SOP-cell and logic optimization techniques for hybrid LUT/SOP reconfigurable architecture. The experimental results are drawn in Section V. Finally, Section VI concludes this paper.

II. SURVEY OF RECONFIGURABLE ARCHITECTURES

In this section, we give a brief architecture survey for three reconfigurable design styles, FPGA, CPLD, and hybrid FPGA.

A. LUT-based FPGA

FPGA utilizes LUT-cell as a basic logic block. The LUT-cell is characterized by $k$ as a $k$-LUT, which consists of a $2^k$ input MUX with $k$ selection signals and $2^k$ SRAM bits. The SRAM is used to store the truth table of a $k$-input function. Hence, a $k$-LUT is capable of realizing any $k$-input Boolean function by properly assigning all the SRAM bits selected by $k$ MUX selection signals. Due to the exponential growth nature of the SRAM bits with respect to $k$, LUT-cell input size $k$ is usually less than 7 in most of commercialized LUT-based FPGA [12], [13]. Taking circuit delay into account, Ahmed and Rose suggest 4-LUT with the best area efficiency [1]. Many research teams investigate area and delay trade-off in order to make FPGA competitive [2]–[4].

B. Macro-cell-based CPLD

CPLD utilizes macro-cell as a basic logic block. The macro-cell configuration is characterized by 3-tuple, $(k, m, p)$. There are $k$ inputs with both positive and negative phases, $m$ product-terms derived by the AND-plane, and $p$ outputs derived by the OR-plane. The advantages of macro-cell are circuit performance and delay predictability since its regular layout and flattened circuit level. Feedback paths are available in many commercialized products in order to extend the capability of implementing multi-level Boolean functions [12], [13]. Anderson and Brown propose an area-efficient multi-level synthesis approach for medium size macro-cell with $(10, 12, 4)$ constraints [5]. In addition, logic packing receives intensively study to cluster multiple output Boolean functions into a macro-cell [6], [7].

C. Combined LUT/Macro-cell-based Hybrid FPGA

Hybrid FPGA is composed of heterogeneous reconfigurable devices. The objective of using different types of logic blocks is to draw on the strength of each to offset the weakness of the other. Previous work utilizes large macro-cells in hybrid FPGA design [8]–[10]. Hu et al. propose a unique function extraction method by profiling the NPN-equivalence of LUT-cells. Four commonly used sub-circuits are extracted to make up a primitive macro-gate. However, once a certain sub-circuit is selected by the MUX in the macro-gate, the other 3 sub-circuits are wasted [11].

APEX-20K is a commercialized hybrid FPGA product. Both LUT-cell and macro-cell are integrated in the device [12]. The $(k, m, p)$ constraints of the macro-cell are large (36, 80, 16) in order to fit more functions. However, the product positioning of hybrid FPGA is in vague since either FPGA or CPLD is capable of replacing the hybrid FPGA. Besides, the most intractable issues are technology mapping and logic packing of heterogeneous cells. To improve the capability of hybrid FPGA, mapping and packing algorithms are required to deal with the following three issues simultaneously: (1) utilize less hardware resource, (2) achieve higher circuit performance, and (3) balance heterogeneous type of reconfigurable cells.

This work is supported in part by the National Science Council of Taiwan under Grant NSC-97-2221-E-155-071-MY2.
III. MOTIVATION

A. Profiling on LUT-cell

For more understanding about the logic utilization of LUT-cell, we perform a profiling experiment on the technology mapping results of delay-optimal 4-LUT FPGA [4]. Our profiling targets on analyzing the functionality of a LUT-cell in terms of the number of product-terms and the literal counts in each product-term. The detailed profiling procedure is drawn in Figure 1.

The profiling results of 20 benchmark circuits mapped by DAOMap are summarized in Figure 2. Since there are at most 8 product-terms in any 4-input function, the figure dimension is $4 \times 8$. For brevity, we subdivide 10 regions from 0 to the maximum matrix value. We use darker color to represent higher value and use lighter color to represent lower value in Figure 2. From the profiling result, we can see that the logic utilization is low in LUT-cell. Most LUT-cells are utilized to implement simple Boolean functions. Only a few LUT-cells implement Boolean functions with more than 4 product-terms.

B. Profiling on Macro-cell

To understand the macro-cell logic utilization, we use the technology mapping results of PLAmap [6] for profiling. Three profiling experiments are performed in this paper. The first profiling targets on analyzing the number of product-terms in a macro-cell and the literal counts in each product-term. The second profiling targets on analyzing the number of product-terms and the output counts in each macro-cell. The last profiling focuses on analyzing product-term sharing for different outputs. Our first profiling procedure is the same as that in Figure 1. Our second and third profiling procedures are summarized in Figure 3.

The profiling results of 20 benchmark circuits mapped by PLAmap under $(10, 12, 4)$ constraints are summarized in Figure 4. For brevity, we use darker color to represent higher matrix value and use lighter color to represent lower matrix value similar to the color scheme in Figure 2. From Figure 4(a), there are a few product-terms with more than four literals. From Figure 4(b), most outputs utilize only a few product-terms. Those unused product-terms are wasted. From Figure 4(c), we can see that product-term sharing is not a majority phenomenon. Most product-terms are used for single output. Hence, we summarize that the logic utilization is low in macro-cell according to our profiling.

From the profiling results, we notice that the logic utilizations of both LUT-cell and macro-cell are poorly low. This phenomenon motivates us to propose compact reconfigurable cells for frequently used functions.

IV. AN EFFICIENT HYBRID LUT/SOP ARCHITECTURE

A. Efficient SOP-cell

From the profiling results of a 4-LUT architecture, we notice that most of LUT-cells are used to implement simple Boolean functions within 4 product-terms. In addition, the profiling results of a $(10, 12, 4)$ CPLD architecture indicate that either the product-term size or the product-term sharing is not significant. Accordingly, we propose an efficient sum-of-product-cell (SOP-cell) without product-term sharing in this paper. The SOP-cell is a two level design with 5 4-input NAND gates. There are 4 NAND gates in the first level and each of the 4 NAND gates fanouts to the second level NAND gate. The SOP-cell is also known as the AO4444. Therefore, there are total 16 inputs and 1 output in the SOP-cell. Neither I/O phase complement nor product-term sharing is available in our SOP-cell in order to reduce transistor count. Figure 5 draws the schematic of our SOP-cell implementing a Boolean function $O = ab + ac + e$. In Figure 5, we disable the rightmost NAND gate and the unused inputs by applying controlling value (GND) and non-controlling value (VDD), respectively. After that, we perform SPICE simulations for both 4-LUT-cell and SOP-cell to characterize their timing. The target technology is TSMC 0.18μm. The transistor count and SPICE simulation delay of SOP-cell and LUT-cell are listed in Table I. In Table I, columns Transistor, Rise Delay, and Fall Delay are the transistor count, rise delay, and fall delay. Column Delay Ratio represents the maximum delay ratio of different cell types taking 4-LUT-cell as the baseline. From Table I, the transistor count and delay of our proposed SOP-cell reduce by 75% and 48% as compared with the LUT-cell, respectively.

B. Architectural Evaluations for Hybrid LUT/SOP FPGA

For a hybrid LUT/SOP reconfigurable design style, both the LUT-cells and SOP-cells are pre-fabricated. If the ratio of SOP-cell number
to LUT-cell number is not balanced, some of the unused cells are forced to be wasted. Hence, we conduct architectural evaluations for the hybrid LUT/SOP FPGA. We use 20 benchmark circuits mapped by DAOMap as our baseline. Then, we simply transform LUT-cell into SOP-cell if the LUT-cell satisfies both literal and product-term constraints of a SOP-cell. In order to understand the best mixture ratio of SOP-cells to LUT-cells in each reconfigurable logic block, we further perform similar transformations under given cell ratio constraints.

There are total 6 architectural configurations, the transformation without cell ratio constraint and the transformations with cell ratio constraints from 0.5 to 4. The evaluation results are summarized in Figure 6. We notice that the parameters listed in Table I for transistor count and circuit delay evaluations. The “Optimal” configuration represents that a SOP-cell is along with 2 LUT-cells, the “Ratio = 2” configuration represents that a LUT-cell is along with 2 SOP-cells, and so on and so forth. From Figure 6, our proposed SOP-cell is efficient for both area reduction and performance improvement. Taking both area and performance into account, we conclude that the best ratio of SOP-cell number to LUT-cell number is about 2 to 3 in each logic block.

C. Synthesis for Hybrid LUT/SOP FPGA

In this subsection, we propose our synthesis techniques and overall synthesis flow for hybrid LUT/SOP design. The initial input is a pure LUT-based design. We transform some LUT-cells into SOP-cells for area and delay optimization. According to the SOP-cell characteristics discussed in Section IV.A, phase complement is not available in our design. Hence, our proposed synthesis techniques maximize the number of positive unate LUT-cells in order to perform SOP-cell mapping.

1) Cell Collapsing: The collapsing operation is to transform a multi-level Boolean network into a two-level representation. In cell collapsing, only positive unate cells are eligible for collapsing into SOP-cells. Figure 7 illustrates cell collapsing. Two cascaded LUT-cells, α and w, are collapsed into one SOP-cell under given product-term and literal constraints.

2) Phase Flipping: Phase flipping is to flip the phase of a negative unate input to a positive unate input. By using phase flipping technique, we are capable of transforming a function with both positive unate inputs and negative unate inputs into a pure positive unate function. Figure 8 illustrates phase flipping. We complement the phase of LUT-cell w in order to form a positive unate LUT-cell α2.

3) Phase Duplication: Phase duplication is to clone a new LUT-cell with complemented phase from the original cell. Since both positive and negative phases are available, we can easily transform binate inputs in LUT-cells into positive unate inputs. Figure 9 illustrates phase duplication. In Figure 9, a complemented LUT-cell w′ is duplicated in order to remove the binate input in LUT-cell α1.

4) Overall Synthesis Flow: Our overall synthesis flow is summarized into three main steps.

Step 1: Cell collapsing is performed on critical cells for timing optimization if both the product-term and literal constraints are satisfied. We repeat the procedure until no further critical cell collapsing is possible.

Step 2: We further improve circuit performance by one-to-one transforming critical LUT-cells into SOP-cells. The transformation procedure is performed repeatedly until there is no possible critical LUT-cell transformation.

Step 3: We transform LUT-cells into SOP-cells in order to balance the ratio of SOP-cell number to LUT-cell number.
TABLE II
RESULT OF AREA REDUCTION AND PERFORMANCE IMPROVEMENT

<table>
<thead>
<tr>
<th>Circuit</th>
<th>Original</th>
<th>Simple</th>
<th>HLS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LUT</td>
<td>Transistor</td>
<td>Delay</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(%)</td>
</tr>
<tr>
<td>s14</td>
<td>1123</td>
<td>184172</td>
<td>8.0</td>
</tr>
<tr>
<td>s14</td>
<td>1128</td>
<td>183024</td>
<td>8.0</td>
</tr>
<tr>
<td>hkey</td>
<td>1376</td>
<td>225664</td>
<td>4.0</td>
</tr>
<tr>
<td>clima</td>
<td>5400</td>
<td>885660</td>
<td>19.0</td>
</tr>
<tr>
<td>des</td>
<td>1274</td>
<td>208596</td>
<td>7.0</td>
</tr>
<tr>
<td>dflip</td>
<td>1007</td>
<td>165124</td>
<td>19.1</td>
</tr>
<tr>
<td>s14p</td>
<td>1150</td>
<td>188680</td>
<td>4.0</td>
</tr>
<tr>
<td>s14p</td>
<td>1106</td>
<td>241120</td>
<td>18.1</td>
</tr>
<tr>
<td>s3841</td>
<td>1333</td>
<td>656565</td>
<td>10.0</td>
</tr>
<tr>
<td>s5p</td>
<td>902</td>
<td>147928</td>
<td>8.0</td>
</tr>
<tr>
<td>fuse</td>
<td>2354</td>
<td>386056</td>
<td>22.0</td>
</tr>
<tr>
<td>musex3</td>
<td>1113</td>
<td>182552</td>
<td>8.0</td>
</tr>
<tr>
<td>pck</td>
<td>2792</td>
<td>457888</td>
<td>10.0</td>
</tr>
<tr>
<td>c298</td>
<td>908</td>
<td>258752</td>
<td>20.0</td>
</tr>
<tr>
<td>c3841</td>
<td>1905</td>
<td>652080</td>
<td>11.0</td>
</tr>
<tr>
<td>scq</td>
<td>1747</td>
<td>634508</td>
<td>11.0</td>
</tr>
<tr>
<td>sgp</td>
<td>1360</td>
<td>206604</td>
<td>8.0</td>
</tr>
<tr>
<td>tseng</td>
<td>789</td>
<td>129396</td>
<td>14.0</td>
</tr>
<tr>
<td>average</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 10. SOP-cell profiling result of 20 circuits

V. EXPERIMENTAL RESULTS
The synthesis techniques proposed in Section IV are implemented as a HLS package (Hybrid LUT/SOP Synthesis). Logic synthesis tool, SIS, is used as our development platform [14]. Our experiment is performed on IBM X3550 2.5GHz Xeon Server with 26 GB memory. MCNC benchmark suite is used in our experiments. All circuits are first technology independently optimized by using rugged script and then technology mapped by using DAOnmap as our initial inputs. We use AO4444 as our SOP-cell, which has at most 4 product-terms and each product-term has at most 4 literals. We use the parameters listed in Table I in our experiment. The run-time of our HLS for each circuit is less than 2 seconds.

According to the architectural evaluation results, we set the cell mixture ratio to 2. The synthesis result is drawn in Table II. Columns Original, Simple, HLS represent the results of DAOnmap circuits, simple LUT-cell to SOP-cell transformation, and our proposed hybrid LUT/SOP synthesis, respectively. Columns LUT, SOP, Transistor, Delay are the number of LUT-cells, the number of SOP-cells, the number of transistors, and the circuit delay, respectively. Columns Tmap and Dmap represents the transistor count reduction and delay improvement, respectively. From Table II, our hybrid LUT/SOP architecture reduces 51.7% transistor counts and improves 35.1% circuit performance as compared with the delay optimal LUT-based FPGA.

Finally, we perform profiling in order to analyze the logic utilization of our proposed SOP-cell. The profiling procedure is the same as the aforementioned procedure drawn in Figure 1. Figure 10 draws the profiling results. As compared with Figure 2 and Figure 4, the logic utilization of our SOP-cell is flattened. Consequently, a compact SOP-cell can be fully utilized to implement various functions.

VI. CONCLUSION
We have proposed an efficient hybrid LUT/SOP reconfigurable design style, which exploits both the advantages of LUT-cell and SOP-cell. The compact SOP-cell requires less hardware resource and achieves higher circuit performance as well. Architectural evaluation indicates best cell mixture ratio to 2. After that, we propose an incremental logic synthesis design flow to synthesize a pure LUT-based design into a hybrid LUT/SOP design. The experimental results demonstrate that our proposed hybrid design style outperforms the LUT-based FPGA in terms of delay, area, and logic utilization. By using the AO4444 SOP-cell, our hybrid design style achieves 35% circuit performance improvement and 51% transistor count reduction as compared with the delay optimal 4-LUT-based FPGA.

REFERENCES