Estimation and Reduction of Delay in Deep Submicron FPGA using LUT

Gaurav Chandil¹, Satay Prakash Singh²

¹Assistant Professor (Electronics), Department of Computer Science Engineering, NITRA Technical Campus, Ghaziabad, India, gaurav.iitm18@gmail.com
²Assistant Professor, Department of Electronics & communication Engineering, KIET Group of Institutions, Ghaziabad, India, er.satayprakash@gmail.com

Abstract

With the advancement of future technologies, there is a need of high speed or high performance logic style with low cost. FPGA has the advantage of low cost but need some improvements in the performance in terms of speed. The purpose of this work is to propose a high speed deep submicron FPGA by reducing the delay of 4-input look up table (LUT) used in devices like Spartan 180nm. In this paper the logic delay of 4-Input LUT is reduced by increasing the size of pass transistor in 180nm process technology. Delay is reduced by 1.05% by using a proper size of 3µm of NMOS pass transistor in 180nm technology. A decoder (10 × 8) benchmark circuit is implemented on Spartan-3 (180nm) FPGA device using the proposed high speed LUT. The performance analysis of FPGA is done by comparing the proposed LUT with the conventional LUT. In this paper work, all the circuits are designed on cadence tool by using gpdk180 technology files for 180nm technology. By using this technique/approach a high speed LUT can be designed for a high speed FPGA which then can be used in many high speed applications like mobile applications and wireless applications.

Keywords: Field programmable gate array (FPGA), look-up table (LUT), delay, deep submicron, Spartan-3.

1. INTRODUCTION

Field-Programmable Gate Arrays (FPGAs) have experienced tremendous growth in recent years and have become a multi-billion dollar industry. Shrinking device geometries resulting in larger gate capacity have provided for greater functionality. The instant programmability gives systems built with these devices a significant time-to-market advantage. However, this programmability comes at a price, since FPGAs are at least three times slower and demand more than ten times the silicon area when implementing the same function on a chip when compared to Standard Cells or Masked-Programmable Gate Arrays. This happens because Standard Cells use simple wires to make interconnections between logic gates but in FPGAs, gates are connected with programmable switches. These switches have much larger resistance and capacitance and hence are slower than the wires in full-fabrication chips. Ideally, to improve the performance of an FPGA we would like to use as few switches as possible for any given circuit. In general, the three main factors affecting overall FPGA performance are the architecture of the FPGA, the quality of the CAD tools, and the electrical transistor level design of the FPGA.

The reference number should be shown in square bracket [1]. However the authors name can be used along with the reference number in the running text. The order of reference in the running text should match with the list of references at the end of the paper.

There are some facts which are motivating us to change the design style from ASICs to FPGA.

- IC costs are rising aggressively.
- ASIC complexity has bolstered development time and costs.
- R&D resources and headcount is decreasing.
- Revenue losses for slow time-to-market are increasing.
- Financial constraints in a poor economy are driving low-cost technologies.

1.1 Field Programmable Gate Array (FPGA)

Field-Programmable Gate Arrays (FPGAs) are user programmable digital devices that provide efficient, yet flexible, implementations of digital circuits [1]. An FPGA consists of an array of programmable logic blocks interconnected by programmable routing resources. The flexibility of FPGAs allows them to be used for a variety of digital applications from small finite state machines to large complex systems. Field Programmable gate arrays (FPGAs) are increasingly used for several applications. The programmability of FPGAs helps in achieving a short design cycle and low development costs, as well as a reduced time to market.

A FPGA chip consists of configurable logic blocks (CLB’s), input-output blocks (IOB’s), and programmable interconnection resources. A CLB contains flip flops and lookup tables which can be configured to perform combinational or sequential logical functions. The architecture of static RAM (SRAM)-based FPGAs consists of two-dimensional (2-D) arrays of logic blocks and programmable interconnection network, surrounded by programmable input/output blocks on the periphery. In
SRAM-based FPGAs, different designs can be mapped into the same device different configuration bitstreams.

2. 4-INPUT LOOK UP TABLE

FPGAs consist of an array of programmable logic blocks that are connected through a programmable interconnection network. Most commercial FPGAs use 4-input look-up table (LUT) as the combinational logic element in their logic blocks for the 4-input LUT architecture have the best power delay sensitivity. Each 4-input LUT is generally coupled with a flip-flop for implementing sequential logic. Logic blocks contain clusters of 4-input LUTs and flip-flops. For example, the primary tile in the Xilinx Vertex-4 is called a configurable logic block (CLB) tile. A CLB contains 4 SLICEs each comprised of two 4-input LUTs and two flip-flops. One of the most popular 4-input look-up table circuits is shown in Fig.2 [3].

![Fig - 1: Island style FPGA [4].](image)

As LUT is the main element in FPGA so reducing the delay of LUT can improve the performance of FPGA. The generally used transistor level architecture of LUT is shown in Fig.3. The total delay of LUT is the logic gate delay and the wire delay. As shown in figure LUT consists of buffers, inverters, NMOS transistors, and keeper. To reduce the delay of LUT proper designing of these circuits is required. In deep submicron FPGAs, the effect of wire resistance and capacitance becomes more prevalent. The wire delay depends on the wire resistance and capacitance. Reduction of wire delay is also required to improve the performance of LUT and also the FPGA.

![Fig - 3: 4-Input Look-up Table circuit [4].](image)

3. THE EFFECT OF LUT SIZE AND CLUSTER SIZE ON DEEP SUBMICRON FPGA PERFORMANCE

The performance of FPGA also depends on the LUT size and cluster size. A paper “The Effect of LUT and Cluster Size on Deep-Submicron FPGA Performance and Density” [2] given by Elias Ahmed and Jonathan Rose shows the relation between the delay of FPGA with cluster size and LUT size. They performed many experiments by taking a set of benchmark circuits. The benchmark circuits were synthesized into different cluster-based logic block architectures, which contain groups of LUTs and flip-flops. Across all architectures with LUT sizes in the range of 2 to 7 inputs, and cluster size from 1 to 10 LUTs, they experimentally determined the relationship between the number of inputs required for a cluster as a function of the LUT size (K) and cluster size (N). The result shows that the performance of FPGAs with these small LUT sizes is significantly worse (by almost a factor of 2) than larger LUTs. According to this paper if the area delay product is considered as main criteria, then the use of clusters of between 3-10, and LUT sizes of 4-6 will produce the best overall results. The work showed that
a look-up table (LUT) size of 5 to 6 gave the best performance in a nonclustered context.

Increasing either LUT size (K) or cluster (N) increases the functionality of the logic block, which has two positive effects:

- It decreases the total number of logic blocks needed to implement a given function.
- It decreases the number of such blocks on the critical path, typically improving performance.

As shown above as the cluster size increases, the buffers shown in Fig. 4 must be sized larger because of larger loading from the internal muxes, which results in an increase in the basic BLE delay. This is shown in Table 3.1 which gives the logic delays as the cluster size increases for the paths indicated in Fig. 4 for a BLE based on a four-input LUT.

### Table - 2: LUT Delay using 0.18µm CMOS process

<table>
<thead>
<tr>
<th>LUT Size (K)</th>
<th>C to D (ps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>199</td>
</tr>
<tr>
<td>3</td>
<td>283</td>
</tr>
<tr>
<td>4</td>
<td>401</td>
</tr>
<tr>
<td>5</td>
<td>534</td>
</tr>
<tr>
<td>6</td>
<td>662</td>
</tr>
<tr>
<td>7</td>
<td>816</td>
</tr>
</tbody>
</table>

Similarly, the design of the larger LUTs must be done carefully, with proper buffer sizing and, in some cases, insertion of buffers within the tree of pass-transistors. Table 3.2 represents the LUT delay as a function of the LUT size. The delay is increasing as the LUT size is increasing.

### 4. SIMULATION RESULTS

#### 4.1 Implementation of Benchmark Circuit on FPGA Device

A decoder (10 × 8) benchmark circuit [5] is implemented on Spartan 3 (3s200pq208-5) 90nm FPGA Device using Xilinx ISE 11.1 for the performance analysis of LUT. It is synthesizable VHDL’93. It accepts a 10-bit parallel encoded character consisting of bits A1, B1, C1, D1, E1, F1, G1, H1, J1 where A1 is the least significant bit. An associated clock, RBYTECLK, latches the 10-bit character on the falling edge. The 8b10b_dec module decodes the 10-bit code into an unencoded 8-bit character along with an associated “KO” bit to indicate if the character output is one of the 12 allowable control, or “K”, characters. After implementation, the device utilization summary is shown below.

Device utilization summary:

Selected Device: 3s200pq208-5

Levels of logic: 8

Number of Slices: 23 out of 1920 1%

Number of 4 input LUTs: 43 out of 3840 1%

Number of IOs: 21

Number of bonded IOBs: 21 out of 141 14%

As the decoder circuit is using 8 logic levels so there would be approximately eight 4-input LUTs in the critical path delay. So eight 4-input LUT will affect the delay in the circuit.
4.2 Specifications for 4-Input LUT design simulation in 0.18µm technology
Simulation of a 4-input LUT is done on cadence with 0.18µm process technology. Some specifications for the simulation of 4-input LUT are as follows:

- Process technology: 0.18µm
- Technology File: gpdk180
- Supply voltage: 1.8V
- Initial pass transistor sizing W/L: 2µm/0.18µm
- Keeper sizing (PMOS transistor): 2µm/0.18µm
- Inverter sizing:
  - For NMOS W/L: 2µm/0.18µm
  - For PMOS W/L: 4µm/0.18µm
- Buffer:
- Stages: 2
- Scaling factor to next stage: 2
- Sizing:
  - For NMOS W/L: 2µm/0.18µm
  - For PMOS W/L: 4µm/0.18µm

4.3 Performance analysis of 4-input LUT design in 0.18µm technology
The 4-input LUT is designed using the above specification and simulation in cadence. As the size of pass transistor is increasing delay is decreasing but the power dissipation is increasing. Table 5.1 shows the results of delay and power dissipation of 4-Input LUT with the variation in pass transistor sizing.

Table 3: Delay and Power Dissipation of 4-input LUT using 0.18µm CMOS process Technology

<table>
<thead>
<tr>
<th>Pass transistor width W(µm)</th>
<th>Delay of 4-Input LUT (ns)</th>
<th>Power dissipation (µw)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>10.494</td>
<td>11.39</td>
</tr>
<tr>
<td>3</td>
<td>10.384</td>
<td>13.90</td>
</tr>
<tr>
<td>4</td>
<td>10.363</td>
<td>16.49</td>
</tr>
<tr>
<td>5</td>
<td>10.353</td>
<td>19.00</td>
</tr>
<tr>
<td>6</td>
<td>10.349</td>
<td>21.33</td>
</tr>
</tbody>
</table>

Chart 1 show a large variation of delay by varying the pass transistor sizing from 2µm to 3µm; if the pass transistor size is further increased variation in delay factor is less. This effect is due to self loading effect of transistor. Increasing the size of transistor also raises the internal capacitance (i.e the diffusion capacitance). In fact, once the intrinsic capacitance starts to dominate the extrinsic load formed by wiring and fan-out, increasing the gate size no longer helps in reducing the delay. It only makes the gate larger in area and also increases the power dissipation. So the optimum size of NMOS pass transistor is 3µm.

Delay of a 4-input LUT by using minimum width pass transistor is 10.494ns but by using proper size pass transistor of 3µm, the delay of 4-input LUT is 10.384ns. So it shows 0.11ns reduction in delay for a 4-input LUT. As a decoder benchmark circuit uses 8 LUTs in the critical Path, so the reduction of delay for decoder benchmark circuit on Spartan 3 device by using 3µm NMOS pass transistor size is

\[ 8 \times 0.11 = 0.88ns \]

5. CONCLUSIONS
In this paper, the work is done to reduce the delay of FPGA in deep submicron domain. The logic delay of 4-input LUT is reduced by using proper sizing technique. With the reduction of logic delay of 4-input LUT, the FPGA devices become faster and can be used in high speed applications like mobile applications and medical applications.

- A decoder (10 x 8) benchmark circuit is implemented on Spartan-3 FPGA Device to compare the results for performance analysis of FPGA.
- Delay of 4-input LUT is reduced by 1.05% by the use of 3µm width pass transistor as compared to minimum width pass transistor in 0.18 µm technology. 3µm is the optimum size of pass transistors for good performance with slightly increase in area and power dissipation.

6. REFERENCES


[9]. Deepak Kumar, Pankaj Kumar and Manisha Pattanaik, “Performance Analysis of 90nm Look-up Table (LUT) for Low Power Application” IEEE Digital System Design conference, DSD, September 2010, Lille, France.