Making Defect Avoidance Nearly Invisible to the User in Wafer Scale
Field Programmable Gate Arrays

by G.H. Chapman and Benoit Dufort
Simon Fraser University
School of Engineering Science
Burnaby, B.C. Canada V5A 1S6
PHONE: (604)-291-3814; FAX: (604)-291-4951; EMAIL: glenn@cs.sfu.ca

Abstract

Field Programmable Gate Arrays have the main features required for
interesting wafer scale systems: high flexibility with potential large num-er of applications, a repeatable cell, and a built in need for switchable
flexible routing. Wafer scale work must involve routing around defective
cells to build the large system. However it is important to minimize signal
delays so the bypassing of the defective cells is invisible. Experiments on
a small test FPGA shows defect avoidance routing using laser link struc-
tures produces delays which are about half those produced by the active
switches required for the FPGA’s operation.

1. Introduction

Wafer Scale Integration designs usually consist of dividing a system into blocks of
repeatable cells surrounded by signal buses. Then after testing the working cells are
interconnected using a switching method. One problem arises with this procedure: the
designs are limited to systems having repeatable unites. However a very flexible system
can be produced if a Wafer Scale Field Programmable Gate array is designed. In
FPGA’s the basic cells are programmable logic blocks which are interconnected by pro-
grammable routing buses. Their logic programmability enables a FPGA to emulate any
digital circuit within its gate and speed limit. The current largest FPGA’s have a 40,000
equivalent gate count, hence limiting their application. By extending this to a wafer
scale FPGA with current 0.35 micron CMOS technology a 200 mm wafer would have a
14 million equivalent gate count, assuming a 75% yield. Using smaller 35x35 mm
slices FPGA’s with 870,000 gate equivalents could be build. The concept of wafer scale
FPGA’s was first proposed by McDonald[1].

Thus FPGA’s have three features wanted for wafer scale applications: a highly flexi-
ble device with potentially a wide range of applications, a repeatable cell, and a built in
need for switchable flexible routing. One important point is that routing is a key ele-
ment in any FPGA designs, involving 70-90% of the chip area[2]. In wafer scale
Figure 1: FPGA basic cell: Logic Block (LUT: Look Up Table; D: D Flip Flop)

Figure 2: FPGA Cell Block Diagram
FPGA’s both cells and signal lines may have defects making the routing problem more difficult. Recent papers have considered this problem from a simulation point of view. One study used Monte Carlo simulation of defects[3] and suggested that using medium sized blocks of cells for substitution was the best result. Another simulation paper considered the use of electrically alterable connections (antifuses and fuses) or active switches in the bus lines[4].

The current paper uses experimental prototypes to investigate and measure the routing problems in large area/wafer scale FPGA’s. The long term concept is to build the wafer scale FPGA, test the cells after fabrication and determine the working logic blocks. Then a permanent defect avoidance routing would be done to hook together only the working cells and avoid the defective ones. The distribution of working FPGA cells is random. Thus an important question is how to minimize the defect avoidance signal delays so to a user the bypassing of the defective cells is invisible. This paper experimentally investigates this question using a test vehicle for a wafer scale field programmable gate array developed previously[5].

2. Test Vehicle FPGA Design

The FPGA cell, along with its switching sections, forms the basic building block of any FPGA system. For the test vehicle a very simple cell was built but one which has all the characteristics of the more complex FPGA cells expected in wafer scale designs. Figure 1 shows the test vehicle basic cell consisting of a 3 input lookup table (LUT) and a D flip-flop[5]. This configuration is used because, while simple, it can be used for both combinational and sequential circuits. Also the number of input/output lines is low so that smaller channel densities could be used in the test system. Figure 2 shows the FPGA cell I/O connections and the bus switch matrix or switch box.

The general cell and bus layout is given in Figure 3 showing the arrangement for the redundancy routing. Note there are two switchboxes in the column/row intersections. The regular switch box consists of active pass transistors that are controlled from the FPGA cell. These are what are used to interconnect the cell signals as part of the FPGA routing. Beside them are a set of laser switch boxes. These contain laser links[6] which enable modest resistance permanent connections and bus line cutting to be done after fabrication. The laser links consist of two implanted lines separated by minimum spacing gap, and connected to adjacent metal buses by contact cuts and vias. These initially act as two back to back diodes. When the gap is hit with a laser pulse (typically from an Argon laser, a 50 microsec, 2 W, pulse focused to 1.2 microns) the dopant flows across the gap forming a permanent connection with a typical resistance of 100 ohms. The same laser can be used to cut the bus lines. This work is done after fabrication on a table which automatically moves the desired link under laser beam, makes the connection, and tests it.

With current typical yields of 75% a reasonable restructuring method is first to locate the bad cells. Then using the Gupta algorithm[7] the nearest cell is substituted to create either a good row (or column) of cells from the external users point of view. The column/row length is set to less than the physical maximum. This process creates delay paths. In our designation the number of defect avoidance path routings (delays) is given
Figure 3: FPGA Symmetrical Restructurable Architecture. The Switch box contains both laser and active switches.

Figure 4: Test vehicle FPGA with 1x12 cells (15 mm x 15 mm)
by the term $N_R$. For all good cells no bypassing occurs and $N_R=0$ while for a single row/column bypass creates a delay $N_R=1$ etc. Since row and column delays are usually different, due to different cell and line lengths in each direction, the delays will be different in different directions.

The actual test vehicle consisted of 1206 x 650 basic FPGA cell[5]. In the first version a 1x12 row of the cells formed a 15 mm x 1.5 mm test chip. This contained several spare bus rows to enable bypass test (see Figure 4). A second chip with a 2x5 matrix of smaller FPGA was also created to test both row and column bypassing.

3. FPGA Delay Approximations

The major drawback of FPGA’s is circuit speed. For a given circuit a custom implementation is much faster than the FPGA because of the large delays in the routing circuitry. This section looks at a model of the effect of redundancy on this delay.

The total delay ($D_{tot}$) of the critical path in a defect free FPGA is approximated by[8]:

$$D_{tot} = N_L(D_{LB} + D_R)$$

where

$N_L$ is the number of logic block in the critical path
$D_{LB}$ is the delay of the logic block
$D_R$ the delay of the routing between two blocks.

The delay of the logic block can be easily calculated but the delay of the routing is much more difficult to approximate. It depends on a large number of factors, like the fanout and the length of the connections[8]. A calculated value is used to give an idea of the delay but it should be noted that this value can vary a lot, even just by remapping the circuit.

For the wafer scale FPGA circuit new delays must be included in the calculations of the total delay $D_{tot}$ because of the extra routing and the physical restructuring of the array. There are two extra delays:

$D_{OH}$ the delay of the overhead restructuring circuitry in each cell
$D_{REC}$ the delay of a restructured channel.

The total delay of the critical path becomes:

$$D_{tot} = N_L(D_{LB} + D_R + D_{OH}) + N_RD_{REC}$$

where $N_R$ is again the number of restructuring channels in the path. This number is hard to estimate in advance because it depends on the restructuring algorithm.

4. FPGA Restructuring Delay Tests

Probably the fastest circuit that can be built by with a FPGA is a n'th order ring oscillator, where an odd number of inverters output is feed back to the input inverter. Thus in one experimental series ring oscillators were programmed using the test vehicle.
FPGA to verify the impact of the restructuring on the maximum frequency of operation of a FPGA circuit. To achieve best results an odd number of cells were programmed to simulate inverters, while a final cell acted as an output buffer, so the capacitance of the oscilloscope would not affect the frequency. The test chip has a long row (1x12 cells) and was used to measure the impact of a column substitution. In this test the number of switches (laser links or active devices) is relatively small, but the bypass path lengths are long; of the cell length (1206 microns).

The idea of the test is to simulate the oscillator first without restructuring, then with one restructured path and so on, until the maximum number of restructured paths possible with the chips was obtained (N_R = 4) Hspice simulations using a delay model where also calculated. The experimental results were obtained as follows: the ring oscillator was incrementally restructured and the frequency measured for each value of N_R from 0 (no restructuring) to 4. The experiment was done as shown in Figure 5 where section (a) shows the laser restructuring and (b) the active cell bypass.

Results of both simulation and the experiments are shown in Table 1 for the active switching rerouting while Table 2 shows those for laser link bypassing. Both illustrate the expected trends of decreasing frequency with increasing N_R. These results are also shown in graphically in Figure 6. The same experiment was repeated on a second chip and the results were similar. Note how the ring oscillator frequency decreases for laser linking about 5% for each restructuring bypass N_R value, while it decline grows to about 10% for active devices. The effect of this is that with laser linking N_R=4 has the same effect as N_R=2 with active switching. Indeed as this reflects on the fastest speed the FPGA is able to run the percentage effect is even smaller for more complicated circuits.

These restructuring experiments showed it is possible to use the laser link and get much better performance then with active switching. Furthermore the case with N_R=4 is one where a very low cell yield is obtained and would rarely be encountered on a real wafer. The simulation was repeated with wider pass transistors, double the size of those designed (7.0 microns instead of 3.5). The active switches showed an 18.3% decrease in their delay time, so that the ring oscillator time deceased only 8.3% per level, still much worse than of the laser links at 5%. Furthermore the area of the pass transistors increased that this point to be the same as the laser links, which means it would begin to dominate the size of the switches in the systems.

5. Effect of Changing the Cell Size

An important question, simulated in Howard’s paper[3], is what is the effect of block size change on the routing delays. In this section the experimental results are used in simulations to extrapolate the effects of changing the this cell size. The logic block is optimized for speed in these simulations, so higher frequencies are obtained than in the fabricated design. In these a comparison is made of the delays between defect bypassing with a direct wired connections (the ideal wire path delays), routing switches using the laser links, and those of active switches. Cells of 1x, 2x and 5 times linear size (1x, 4x, 25 area) were then simulated. The same restructuring scheme as in the section 4’s experiments were employed. Table 3 shows the resulting ring oscillator frequencies for the direct, laser link and active cases. In these 1x size means the experimental design
Figure 5: Ring Oscillator Restructuring Experiment

Figure 6: FPGA Ring Oscillator test delay vs Number of delay paths $N_R$
size was used (1206 x 650 μm), in 2x are 2412 x 1300 μm cells and 5x means a 6032 x 3250 μm cell. The total length of structure would change from 1.5 cm with the 1x, to 3 cm on 2x and 7.5 cm in the 5x case.

These values indicate that the laser link is even better for very large cells, going from 1.6 times faster for the small cells to two times better than active cells for the 5x case. Note these simulations are comparing the results for the length of the routing architectures, and do not take into account changes in the performance of the logic block. Clearly with larger cells, more complex circuits can be done with each block. Hence using the ring oscillator as a test example does make the larger cells appear to be at a speed disadvantage without taking into account the routing delays resulting that result from hooking together the simpler blocks for more complex systems.

6. Conclusions

The operation speed of wafer scale FPGAs has been shown to be affected by the type of defect avoidance routing used. The experimental results show laser interconnections methods can increase the operation frequency of FPGAs with defects by between 1.6 and 2 times depending on the size of the cell. Such savings are clearly important in such large systems.

7. References


<table>
<thead>
<tr>
<th>Number restructured paths ((N_r))</th>
<th>Simulated Frequency ((\text{kHz}))</th>
<th>Frequency change from non-defective path ((N_r=0))</th>
<th>Experimental Frequency ((\text{kHz}))</th>
<th>Frequency change from non-defective path ((N_r=0))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>943</td>
<td></td>
<td>956±2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>835</td>
<td>-9.3%</td>
<td>859±2</td>
<td>-10.1%</td>
</tr>
<tr>
<td>2</td>
<td>787</td>
<td>-16.5%</td>
<td>727±2</td>
<td>-24.0%</td>
</tr>
<tr>
<td>3</td>
<td>690</td>
<td>-26.8%</td>
<td>629±2</td>
<td>-34.2%</td>
</tr>
<tr>
<td>4</td>
<td>549</td>
<td>-41.8%</td>
<td>543±2</td>
<td>-43.2%</td>
</tr>
</tbody>
</table>

Table 1: FPGA ring oscillator, active switching

<table>
<thead>
<tr>
<th>Number restructured paths ((N_r))</th>
<th>Simulated Frequency ((\text{kHz}))</th>
<th>Frequency change from non-defective path ((N_r=0))</th>
<th>Experimental Frequency ((\text{kHz}))</th>
<th>Frequency change from non-defective path ((N_r=0))</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>943</td>
<td></td>
<td>956±2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>885</td>
<td>-6.2%</td>
<td>906±2</td>
<td>-5.2%</td>
</tr>
<tr>
<td>2</td>
<td>833</td>
<td>-11.7%</td>
<td>820±2</td>
<td>-14.2%</td>
</tr>
<tr>
<td>3</td>
<td>787</td>
<td>-16.5%</td>
<td>761±2</td>
<td>-20.4%</td>
</tr>
<tr>
<td>4</td>
<td>740</td>
<td>-20.9%</td>
<td>723±2</td>
<td>-24.5%</td>
</tr>
</tbody>
</table>

Table 2: FPGA ring oscillator, laser linking

<table>
<thead>
<tr>
<th>(N_r)</th>
<th>Direct connection</th>
<th>Laser Linking</th>
<th>Active switching</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1x</td>
<td>2x</td>
<td>5x</td>
</tr>
<tr>
<td>0</td>
<td>4080</td>
<td>2750</td>
<td>1400</td>
</tr>
<tr>
<td>1</td>
<td>3720</td>
<td>2520</td>
<td>1220</td>
</tr>
<tr>
<td>2</td>
<td>3400</td>
<td>2230</td>
<td>1050</td>
</tr>
<tr>
<td>3</td>
<td>3140</td>
<td>2000</td>
<td>943</td>
</tr>
<tr>
<td>4</td>
<td>2920</td>
<td>1800</td>
<td>826</td>
</tr>
</tbody>
</table>

Table 3: FPGA Ring Oscillator, comparison of cell size