EN1640
Design of Computing Systems
Spring 2015

Lecture 7: FPGA Dataflow

February 6, 2015
Prof. R. Iris Bahar

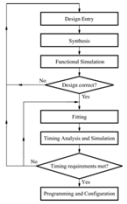
BROWN

Homework #1, Lab #1

- Lab #1 is posted on the webpage
  www.brown.edu/Departments/Engineering/Courses/engn1640
- Please show all your work and state all assumptions for the problems.
- Homework due today (hand in during class or by 5pm to the TAs in room 196)
- Lab assignment is due next Friday, Feb. 13 by 6pm

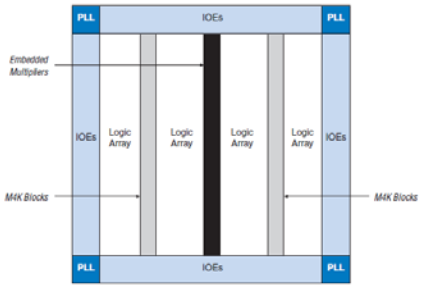
Topics

1. Programmable logic
2. Design Flow
3. Verilog --- A Hardware Description Language




```
module fulladd4(output reg[3:0] sum,
                                output reg c_out,
                                input [3:0] a, b,
                                input c_in);
_
_
endmodule
```

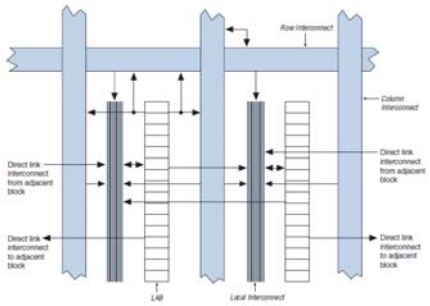

Our FPGA: Altera Cyclone II



- 2D array of Logic Array Blocks (LABs), with 16 Logic Elements (LEs) in each LAB
- EP2C35 (in DE2 board) has 2076 LABs configured in rows and columns for a total of 33216 LEs.
- Chip has 105 M4K memory blocks and 35 embedded multipliers
- Contains 4 PLLs to generate clock frequencies

BROWN

FPGA Cyclone II organization

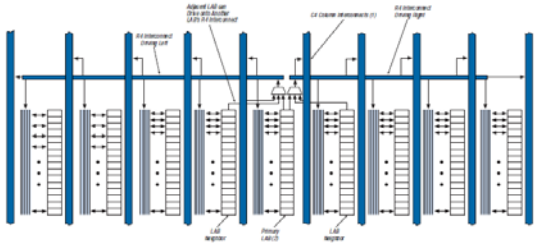


- Logical interconnects transfer signals between LEs in the same LAB and are driven by column and row interconnects and LE outputs within the same LAB
- Neighboring LABs, PLLs, M4K RAM and multipliers from the left and right can also drive any LAB's local interconnect directly
- Larger communication range can be achieved through R4, C16, and R24 links.

5

BROWN

FPGA Cyclone II organization

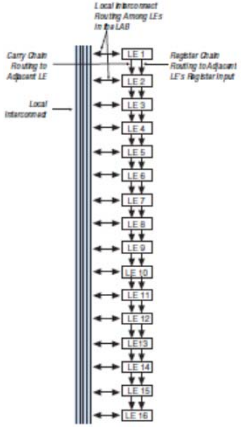


- Multitrack interconnect consists of row (directlink, R4, R24) and column (register chain, C4, C16)
- R4/C4 interconnects spans 4 blocks (right, left / top, down)
- R24/C16 spans 24/16 blocks and connects to R4/C4 interconnects
- R4/C4 can drive each other to extend their range

6

BROWN

FPGA Cyclone II organization



- Column interconnect allows creation of register chain within an LAB
- Register output of one LE connects directly to register input of next LE
- C4 interconnects spans 4 blocks (top, down)
- C16 allows for vertical routing through device. Drive to other row/column interconnects at every 4th LAB.

7

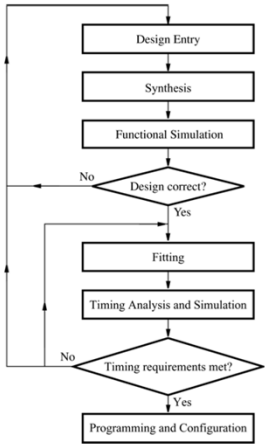
BROWN

More Details of the Cyclone II

- The course webpage includes links to more documentation on the Cyclone II devices
- Look under Assignments (Lab1)

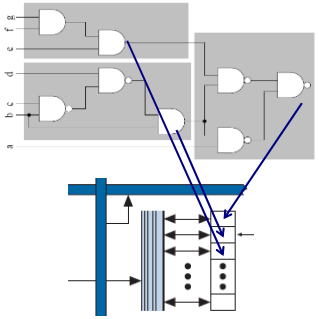
- Of particular interest:
  - Click on "Cyclone II device documentation"
  - Under "Cyclone II Device Handbook", Chapter 2 goes into detail regarding the architecture of the chip and the hardware implementation

2. Design Automation



9

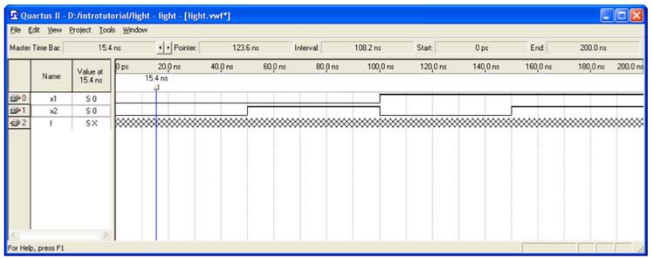
Synthesis



- Maps your design into a network of 4-input LEs
- Packs the LEs into logic array blocks (LABs) of at most 16 LEs

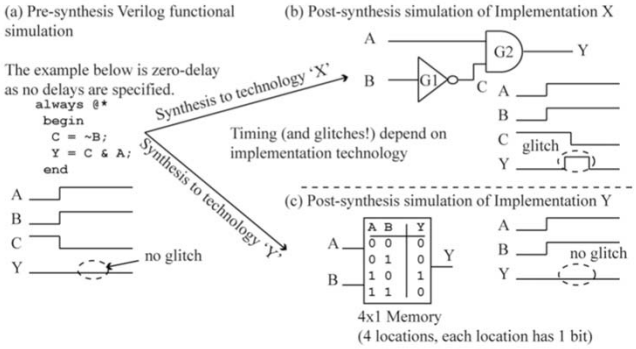
10

Functional Simulation



11

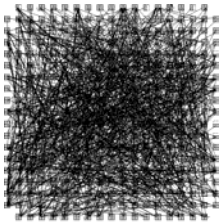
Pre- vs. post-synthesis simulation



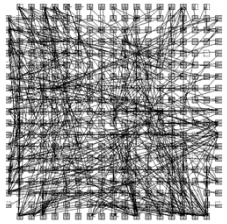
[Example from Thornton & Reese]

12

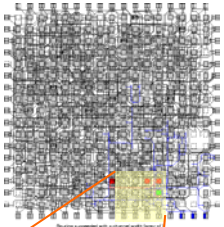
Fitting (i.e., Placement & Routing)



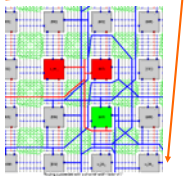
[random fitting/placement]



[after placement]



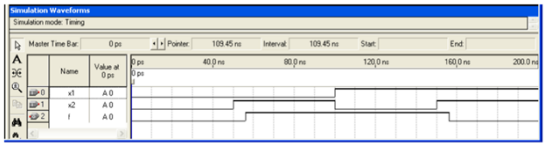
[after routing]



- Fitting Objectives:
  - Reduce routing resources (to fit into the FPGA device)
  - Meet timing requirements

13

Timing simulation (post fitting)

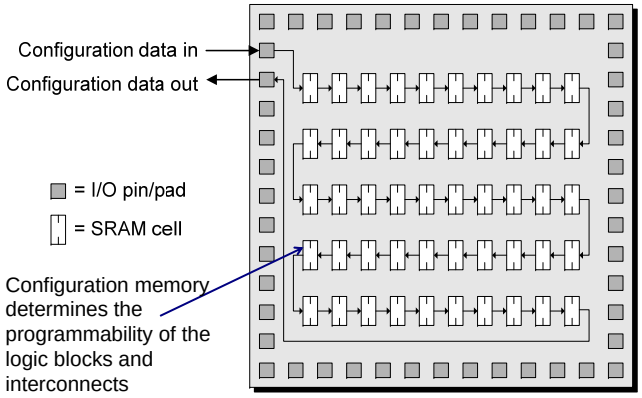


- The path with the largest delay between flip-flops determines the circuit clock frequency
- Smaller worst case delay → faster processor clock frequency

14

BROWN

Programming the FPGA



15

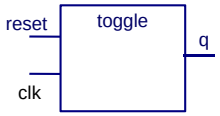
BROWN

3. Introduction to Verilog

- Hardware description language (HDL) vs. software languages:
  - Concurrency
  - Propagation of time
  - Signal dependency or sensitivity
- Verilog:
  - Case sensitive, with syntax similar to C
  - Comments designated by // to end of line or by /* to */ across several lines
  - Textbooks:
    - Introduction to Logic Synthesis using Verilog HDL
    - Verilog Quickstart
    - Verilog Digital System Design
    - The Verilog Hardware Description Language

16

Verilog modules

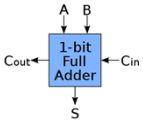


- The functionality of each module can be defined with 3 modeling levels:
  - Structural (or gate level)
  - Dataflow level
  - Behavioral (or algorithmic level)
- Verilog allows different levels of abstraction to be mixed in the same module.

17

Modules and ports

```
module FA(A, B, Cin, S, Cout);
input A, B, Cin;
output S, Cout;
...
endmodule
```



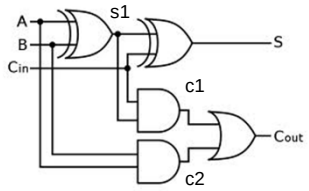
- All port declarations (input, output, inout) are implicitly declared as `wire`.
- If an input should hold its value, it must be declared as `reg`.

```
module FA(input A, input B, input Cin, output S, output Cout);
...
endmodule
```

18

Verilog Part 1. Structural modeling

- **Data types: bits**
- Nets represent connections between hardware elements.
- Continuously driven by the output of connected devices.
- Nets are declared using the keyword `wire`.



- `wire s1;`
- `wire c1, c2;`
- `wire d=0;`

19

Gate level modeling

```
wire Z, Z1, OUT, OUT1, OUT2, IN1, IN2;

and a1(OUT1, IN1, IN2);
nand na1(OUT2, IN1, IN2);
xor x1(OUT, OUT1, OUT2);
not (Z, OUT);
buf final (Z1, Z);
```

- Describes the topology of a circuit
- All instances are executed concurrently just as in hardware
- Instance name is not necessary
- The first terminal in the list is an output; others are inputs
- Not the most interesting modeling technique for this class

20