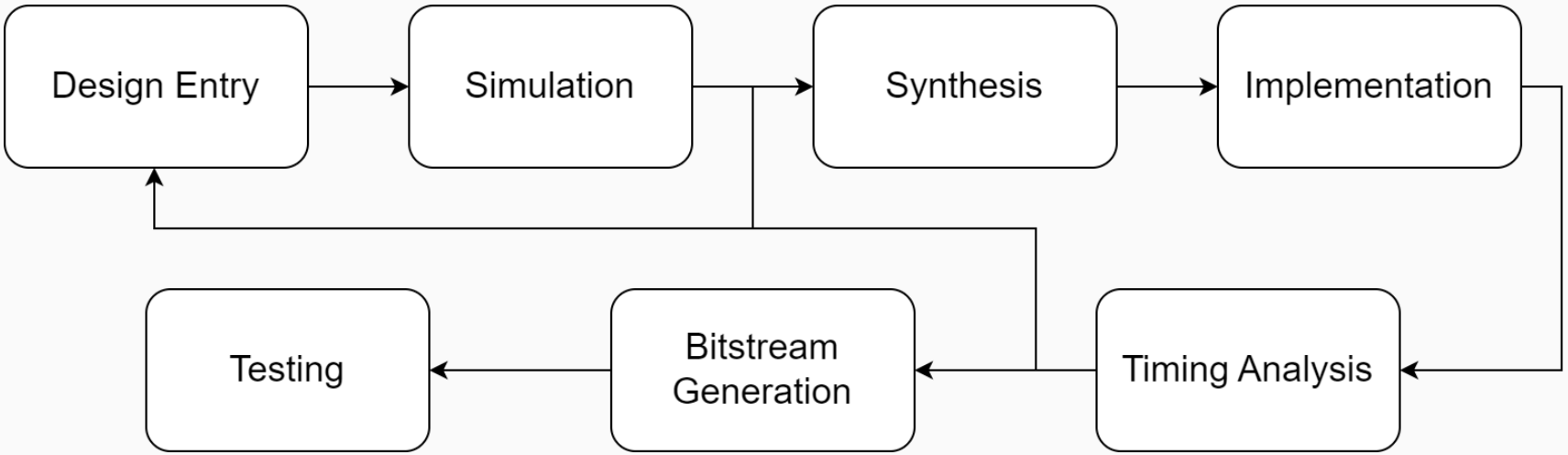
The FPGA Design Process

Danijela Lazarevic

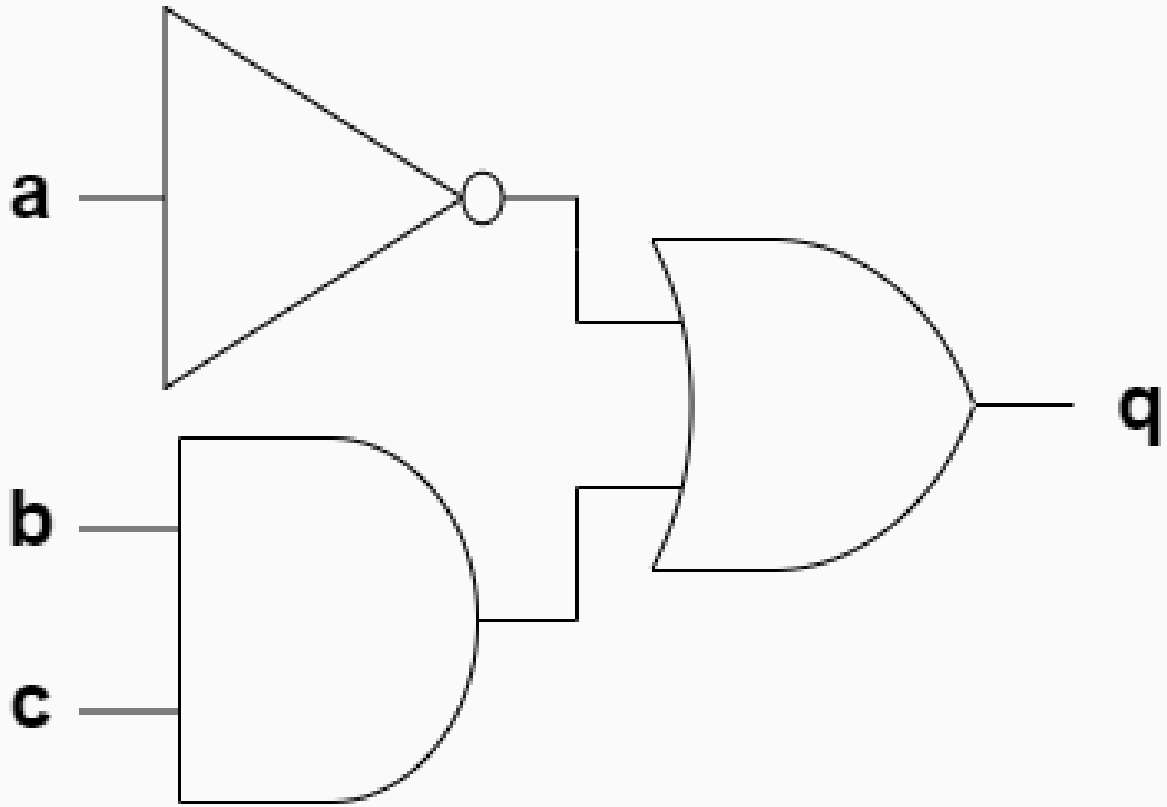
November 24, 2022

Overview



FPGA design flow diagram

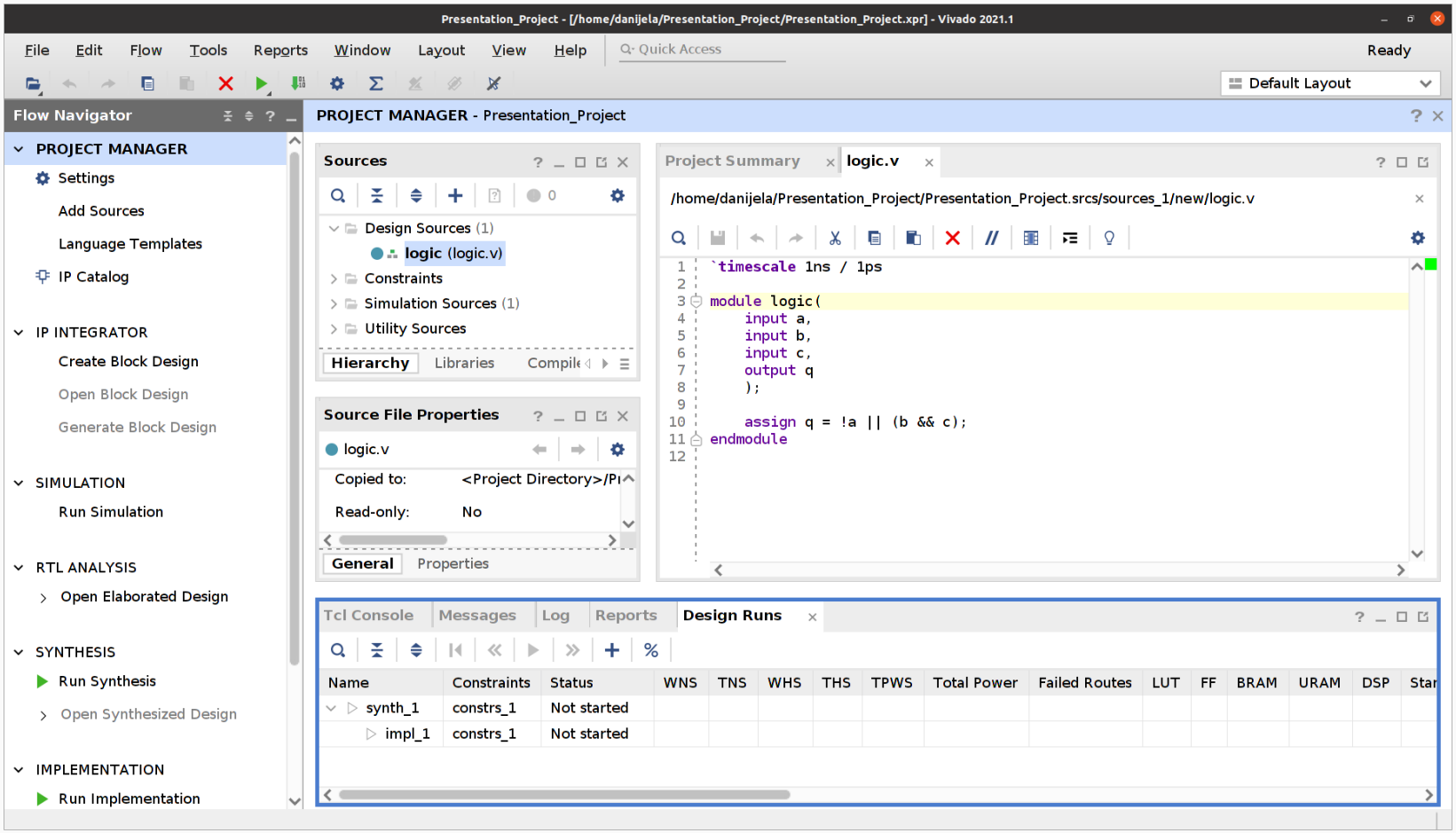
Demonstration Circuit



Design Entry

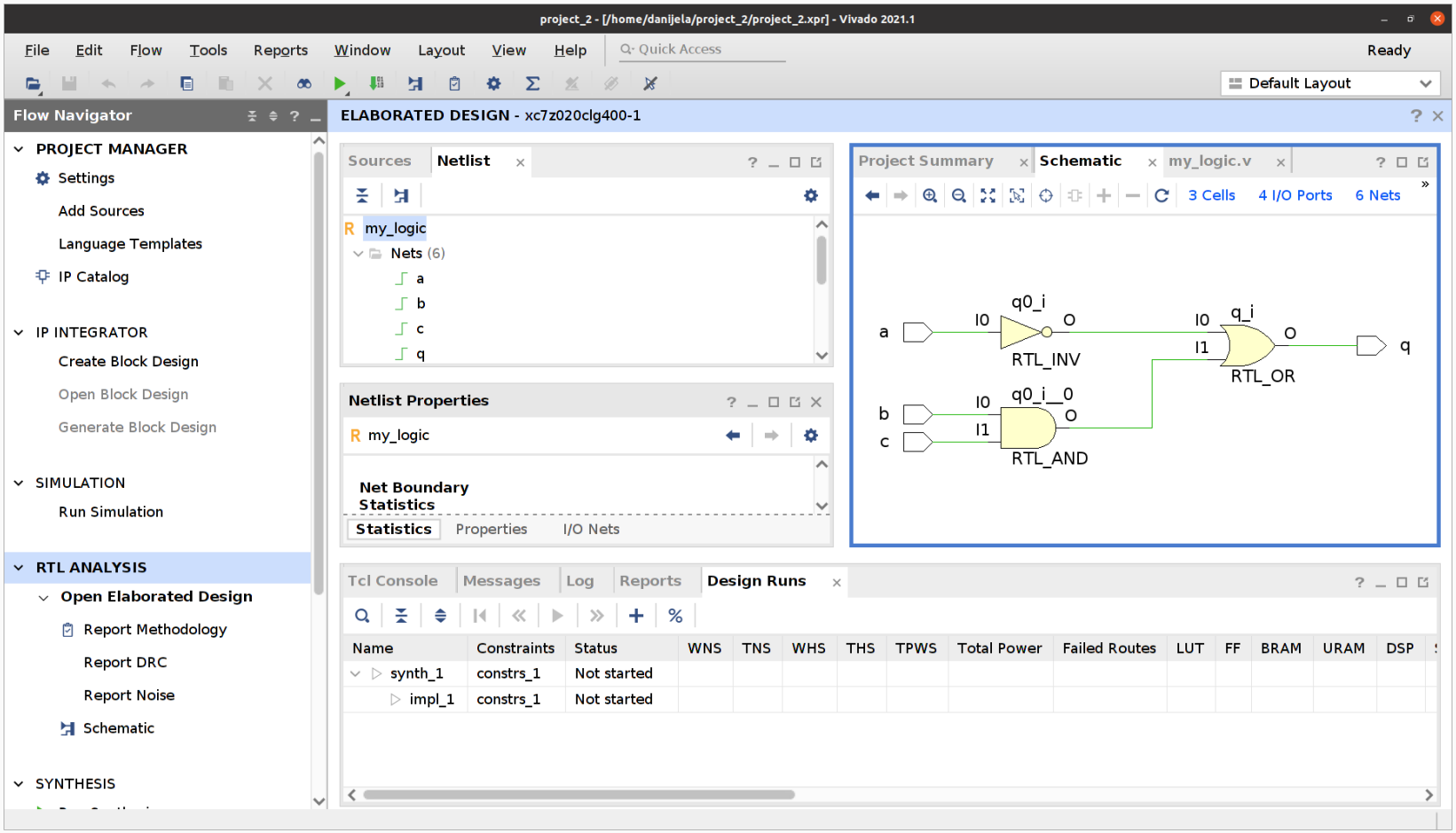
- RTL Design
- Implement each block using HDL
- Integration/Development of IPs
- Constraints:
    - Connection to physical pins
    - Signal delays
    - Clock frequency
    - Power Consumption

Design Entry



HDL design implementation

Design Entry



Circuit representing the design

Design Entry

my_logic.v | my_logic_tb.v | **my_logic_constraint.xdc** | Schematic (2)

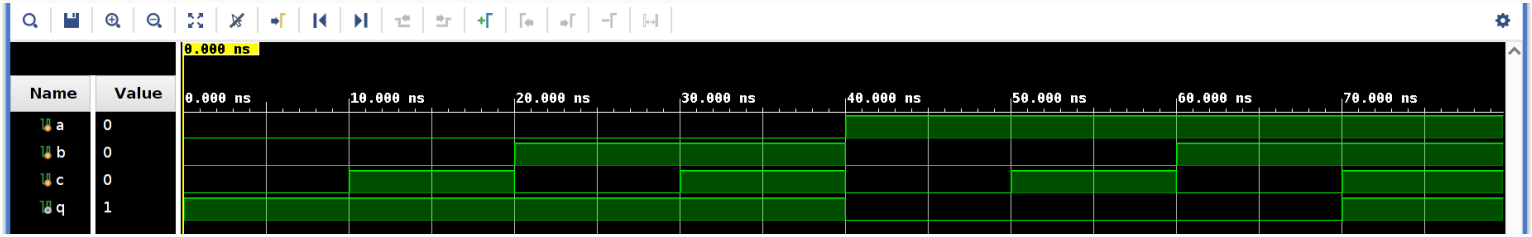
/home/danijela/project_2/project_2.srcs/constrs_1/new/my_logic_constraint.xdc

```
1  create_property INIT cell -type string
2  set_property PACKAGE_PIN Y19 [get_ports a]
3  set_property PACKAGE_PIN W19 [get_ports b]
4  set_property PACKAGE_PIN W20 [get_ports c]
5  set_property PACKAGE_PIN W18 [get_ports q]
6  set_property IOSTANDARD LVCMOS18 [get_ports a]
7  set_property IOSTANDARD LVCMOS18 [get_ports b]
8  set_property IOSTANDARD LVCMOS18 [get_ports c]
9  set_property IOSTANDARD LVCMOS18 [get_ports q]
10 set_max_delay -from [get_ports {a b c}] -to [get_ports q] 2.000
11
12
13
```

Constraints file

Simulation

- Verification of RTL source code
- Behavioral simulation
- Fast compared to later simulations
- No information about delays



Behavioral simulation

Synthesis

- Translation to technology-specific gate level netlist
- Input:
  - RTL Design
  - Standard cell libraries
  - Constraints
  - Technology information
- Source code mapped to standard cell libraries and optimized for target technology.

Synthesis

- Syntax Analysis: Check syntax for correctness
- Library Definition:
  - Standard cell libraries loaded.
  - Contain information on technology, cells and IPs.
- Elaboration and Binding:
  - RTL converted to boolean structure.
  - Optimized according to boolean algebra.
  - Cells bound to standard cell libraries.

Synthesis

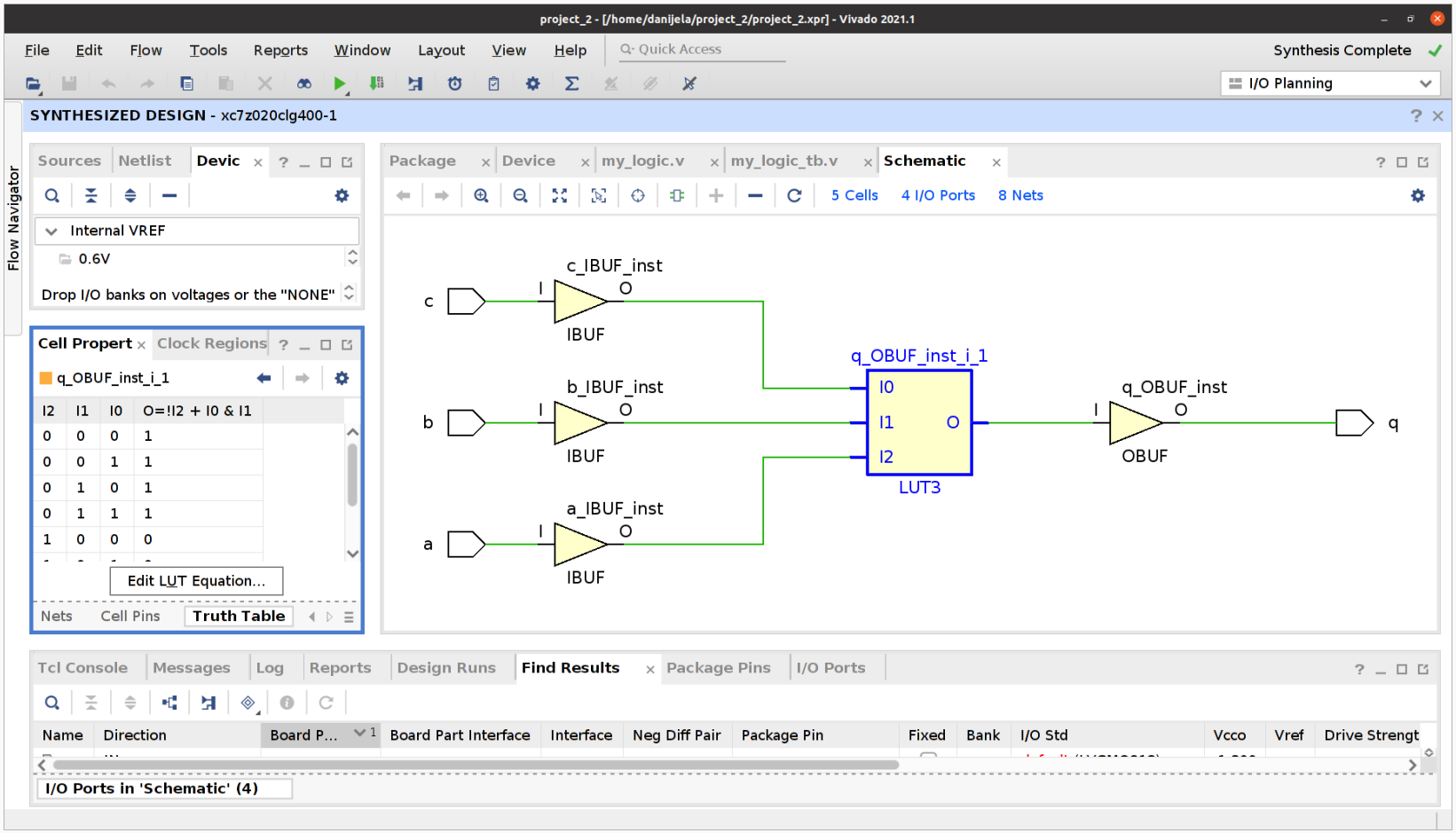
- Constraint Definition: constraint file is loaded.
- Technology Mapping: Move from a generic standard cell library to technology dependent library.
  - Provides additional information about cells like delays
- Post-Mapping Optimization: iteratively applying transforms to improve cost function
  - Resize cells
  - Buffer/Clone to reduce load on critical nets
  - Decompose large cells
  - Swap connections on commutative pins or among equivalent nets.

Synthesis

```
1 module my_logic(a, b, c, q);
2   input a;
3   input b;
4   input c;
5   output q;
6
7   wire a;
8   wire a_IBUF;
9   wire b;
10  wire b_IBUF;
11  wire c;
12  wire c_IBUF;
13  wire q;
14  wire q_OBUF;
15
16  IBUF a_IBUF_inst(.I(a), .O(a_IBUF));
17  IBUF b_IBUF_inst(.I(b), .O(b_IBUF));
18  IBUF c_IBUF_inst(.I(c), .O(c_IBUF));
19  OBUF q_OBUF_inst(.I(q_OBUF), .O(q));
20  LUT3 #(.INIT(8'h8F)) q_OBUF_inst_i_1(.I0(c_IBUF), .I1(b_IBUF), .I2(a_IBUF), .O(q_OBUF));
21 endmodule
```

Synthesized netlist of the example circuit

Synthesis



Schematic showing the synthesized netlist of the example circuit

Implementation

Consists of several steps where the netlist elements are physically placed and mapped to the physical resources:

- Logic Optimization (opt_design)
- Power Optimization (power_opt_design)
- Placement (place_design)
- Physical Synthesis (phys_opt_design)
- Routing (route_design)

Result: Map of netlist elements to FPGA blocks/Configuration file

Implementation

Logic Optimization

- Retargeting: conversion of cell types to aid downstream optimization (Ex.: MUX to LUT)
- Constant Propagation:
  - Elimination: AND with constant 0 logic
  - Reduction: 3-input AND with constant 1 input reduced to 2-input AND
  - Redundancy: 2-input OR with constant 0 reduced to wire
- Sweeping: removes cells with no loads
- etc.

Implementation

Power Optimization

- Restructuring of logic to reduce dynamic power consumption. (Clock Gating)

Placement

- Optimal cell placement in device considering timing, wire length and congestion.

Implementation

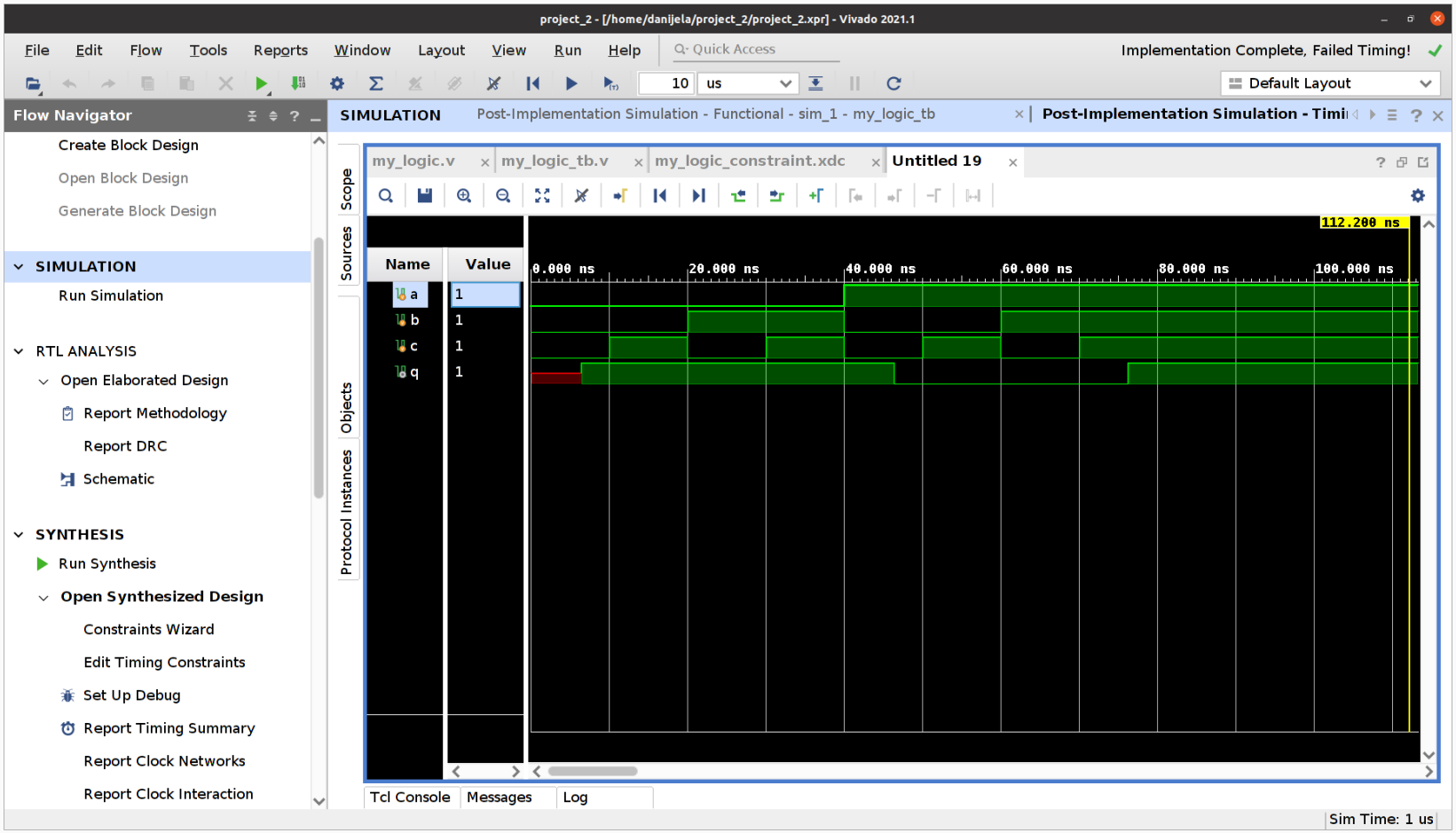
Physical Synthesis

- Further improvement of placement with physical optimizations. (e.g. logic replications to reduce fan out delay)

Routing

- Connections between logic blocks.
- Maps connections to routing resources.
- Programmable switches determined to connect the logic

Timing-Analysis



Timing simulation

Bitstream Generation

- Synthesis and implementation results are generated into a bitstream file.
- bitstream structure:
  1. Header packets to prepare the configuration process.
  2. The actual configuration bits.
  3. Header packets to clean up configuration process.
- bitstream file written into FPGA memory
- Result: .bit file that is used to program the device.
- On boot: FPGA loads configuration from memory and programs all cells

References i

[1] Yizhou Shan. *FPGA Bitstream Explained*. `http://lastweek.io/fpga/bitstream/`. Online; accessed 14 November 2022.

[2] XILINX. *Vivado Design Suite User Guide: Implementation*. `https://docs.xilinx.com/r/en-US/ug904-vivado-implementation/Implementing-the-Design`. Online; accessed 14 November 2022.

[3] FPGA Tutorial. *Introduction to the FPGA Build Process*. `https://fpgatutorial.com/fpga-build-process/`. Online; accessed 14 November 2022.

References ii

[4] Digilent inc. *Getting Started with FPGA Design 3: Basic FPGA Design Flow*. https://www.youtube.com/watch?v=KoljXGho5KY. Online; accessed 14 November 2022.

[5] Andreas Johansson. *Demonstration: FPGA design flow using Vivado*. https://www.youtube.com/watch?v=VYCWXaYLSWY&t=81s. Online; accessed 14 November 2022.

[6] vhdlwhiz. *VHDL AND FPGA Terminology*. https://vhdlwhiz.com/terminology/translate/. Online; accessed 14 November 2022.

References iii

[7] *Place and Route for FPGAs*. https://www.eng.uwo.ca/people/wwang/ece616a/616_extra/notes_web/5_dphysicaldesign.pdf. Online; accessed 15 November 2022.

[8] Barbara Gigerl, Rishub Nagpal *SoC Design Flow Tutorial*. https://www.iaik.tugraz.at/wp-content/uploads/2022/09/slides-2.pdf. Online; accessed 15 November 2022.

[9] Dr. Adam Teman *Digital VLSI Design - Lecture 3: Logic Synthesis Part 1* . https://www.eng.biu.ac.il/temanad/files/2018/11/Lecture-3-Synthesis-Part-1.pdf. Online; accessed 15 November 2022.

References iv

[10] Dr. Adam Teman *Digital VLSI Design - Lecture 4: Logic Synthesis Part 2*. https://www.eng.biu.ac.il/temanad/files/2018/11/Lecture-4-Synthesis-Part-2.pdf. Online; accessed 15 November 2022.