Chapter 5: Project Planning and Designing Tools

Credit to : Dr. MD Rizal Othman
Faculty of Electrical & Electronics Engineering
Universiti Malaysia Pahang

Ext: 6036

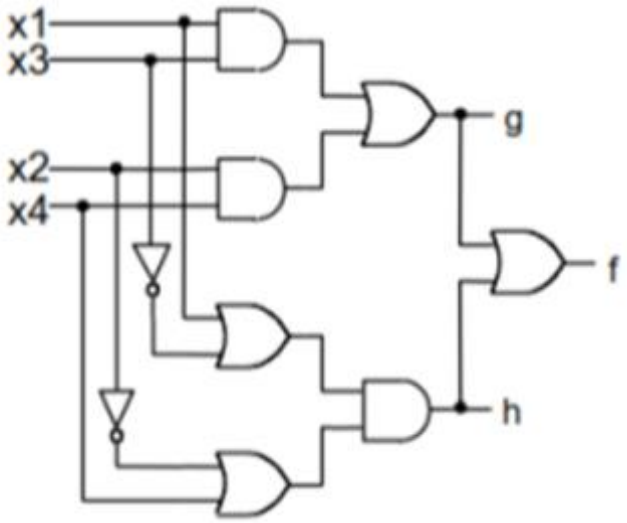
FPGA Generic Design Flow

- Design Entry
  - Create design :- Schematic or HDL
- Design Implementation
  - Partition
  - Place
  - Route

  } create bit stream file
- Design Verification
  - Use simulator to check functionality
  - Check operation after loading

FPGA Generic Design Flow

- Design Entry
  - Create design :- Schematic or HDL




```
// Example 4

module example4 (x1, x2, x3, x4, f, g, h);
    input x1, x2, x3, x4;
    output f, g, h;

    assign g = (x1 & x3) | (x2 & x4);
    assign h = (x1 | ~x3) & (~x2 | x4);
    assign f = g | h;

endmodule
```

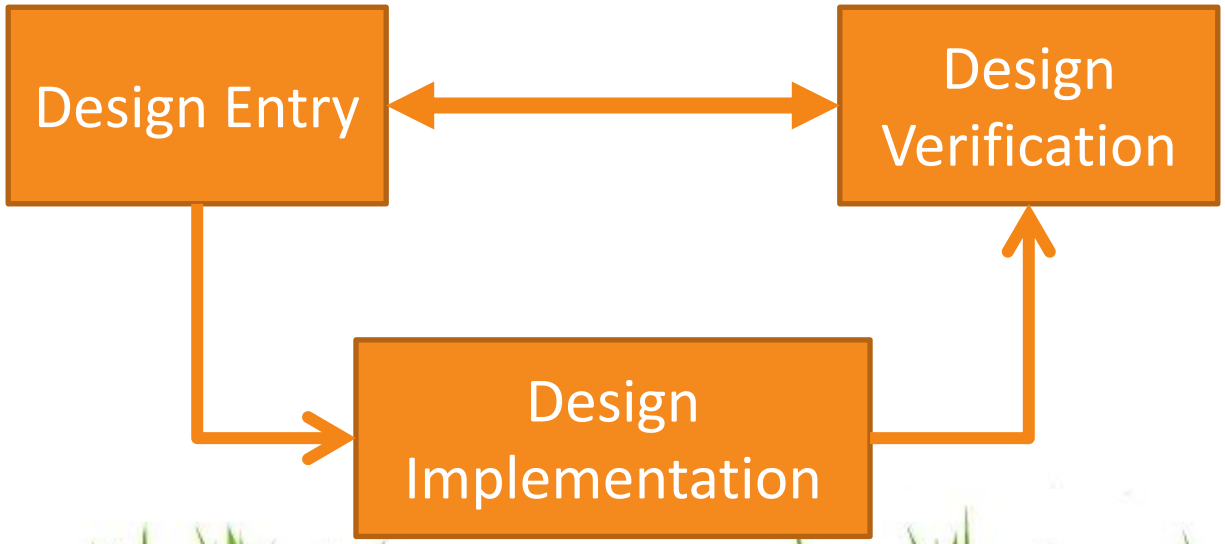
FPGA Generic Design Flow

- Design Implementation
  - Implementing HDL to FPGA (something that fpga can use)

1. Partition
2. Place
3. Route

FPGA Generic Design Flow

- Design Verification
  - Use simulator to check functionality
  - Check operation after loading



Verification Plan

Types of verification

- Simulation
  - simulate the HDL code with no timing information
  - simulate the synthesized with no timing information
- Static timing analysis (STA)
  - Compares the implemented design to specified timing constrains

Verification Plan

- Bench Testing
  - Test a physical system (prototype) in a controlled environment (lab)
- Field testing
  - Test the final product in a real-world environment

FPGA Generic Design Flow

- Combinational Logic must be covered by 4-input and 1-output “gate” – done in translation process.
- FF from the circuit must map to FPGA FF.
- Placement in general attempts at wiring minimization.
  - Wiring increases → delay increase → speed decreases

Introduction to Xilinx ISE

- Tool to configure FPGA provided by Xilinx
- Integrated Software Environment
  - Integrated collection of tools with GUI (user friendly). Eg: XST, PACE, CoreGen, Constraint Editor, Impact
- Support all the steps required to complete the design

Xilinx development flow

1. Design Entry
   - .sch, .v , .vhd
2. Synthesis : use XST (Xilinx Synthesis Tool)
   - Produces a netlist file starting from an HDL/Schematic description
   - Converts .sch, .v , .vhd → .ngc (netlist file)
3. Translate : NGD Build
   - Reads all input design netlists and then write the results in a single merged file, that describes logic circuit constrains.

Xilinx development flow

3. Translate : NGD Build
   - NGD is Native Generic Database: describes the logic design reduced to xilinx primitives
4. Mapping
   - Maps the logic on device components
   - Takes the netlist and group the logical element into CLBs and IOBs
   - Generate NCD and PCF
   - NCD (Native Circuit Description)
     - Represents the physical circuit description of the input design as applied to a specific device → device dependent

Xilinx development flow

5. Place and Route
   - Determines the placement of the cells and the routing between the cells
   - PCF (Physical Constrains File)
     - Contains physical constrains
     - Info about which CLB can be used and which CLB cannot be used (increase delay)
6. Bit Stream Generation
   - A bit stream is a stream of data that contains the location information for logic on a device
7. Configuration/Programming
   - Download a bit file into FPGA using JTAG port

HDL compiler → XST

Simulator → Xilinx ISEsim

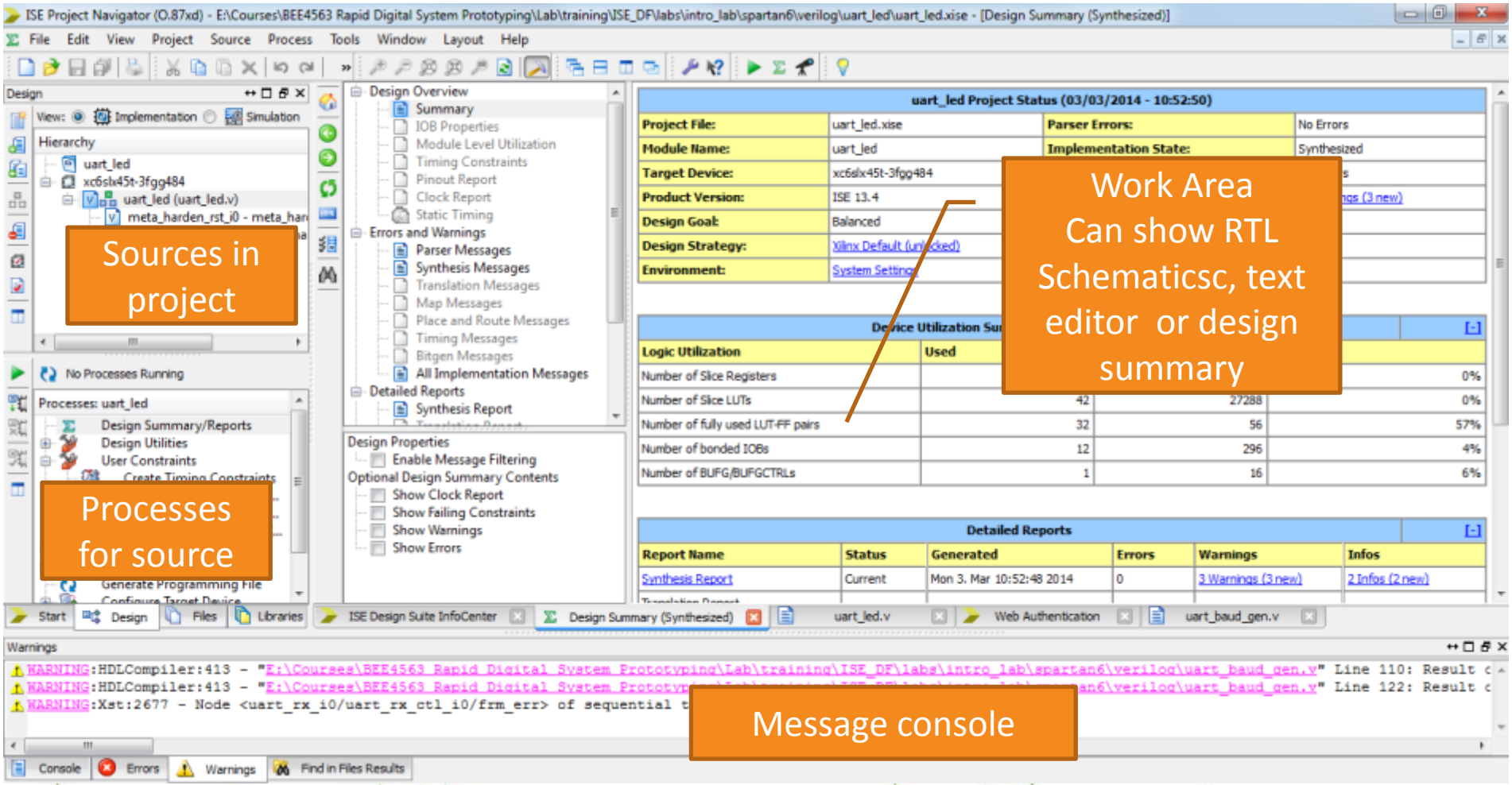
Core Generator & Arch. Wizard → CoreGen

Pinout and Area Constrain Editor → PACE

Implementation → Translate/MAP/P&R

Device Configuration → IMPACT

Project Navigator



Foundation Series ISE Software

- ISE®Design Suite
  - Integrated Synthesis Environment
  - Project Navigator
  - ChipScope™ Pro Tool
  - Embedded Development Kit
- WebPACK™ software

Xilinx Design Process

- Step 1: Design Entry
  - HDL (Verilog or VHDL)
- Step 2: Synthesis
  - Translate HDL files into a netlist
- Step 3: Implementation
  - Translate, Map, Place & Route
- Step 4: Configuration
- Simulation can occur after Steps 1, 2, or 3

Xilinx Design Process

- Step 1: Design Entry
  - HDL (Verilog or VHDL)
- Step 2: Synthesis
  - Translate HDL files into a netlist
- Step 3: Implementation
  - Translate, Map, Place & Route
- Step 4: Configuration
- Simulation can occur after Steps 1, 2, or 3

Tool and Processes

Design entry

Synthesis

Implementation

Configuration

Additional Tool