Lab 1: Logic Synthesis with Design Compiler

Objective

Learn how to compile the RTL design using Design Compiler GUI and command line interfaces.

Introduction

Using Design Compiler, it is possible to:

- Produce fast, area-efficient ASIC designs by employing user-specified or standard-cell libraries
- Translate designs from one technology to another
- Explore design tradeoffs involving design constraints such as timing, area, and power under various loading, temperature, and voltage conditions
- Synthesize and optimize finite state machines
- Integrate netlist inputs and netlist or schematic outputs into third-party environments while still supporting delay information and place and route constraints
- Create and partition hierarchical schematics automatically.

Basic Synthesis Flow

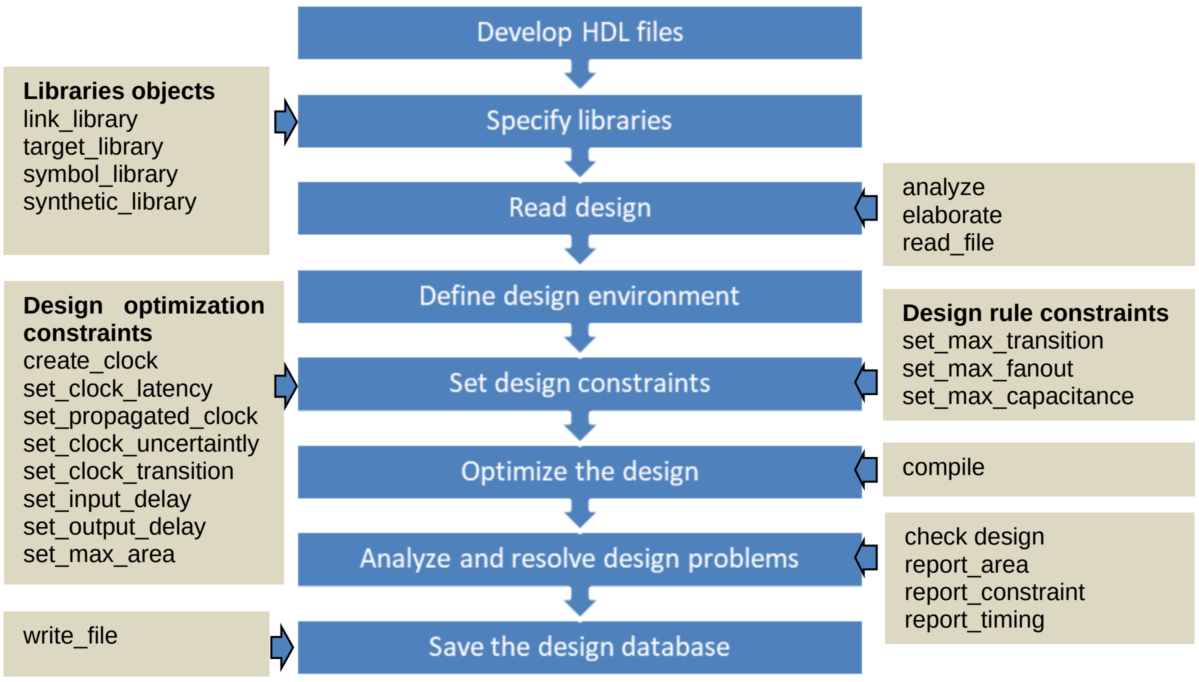


Fig. 1. Basic synthesis flow

Design Compiler input and output files

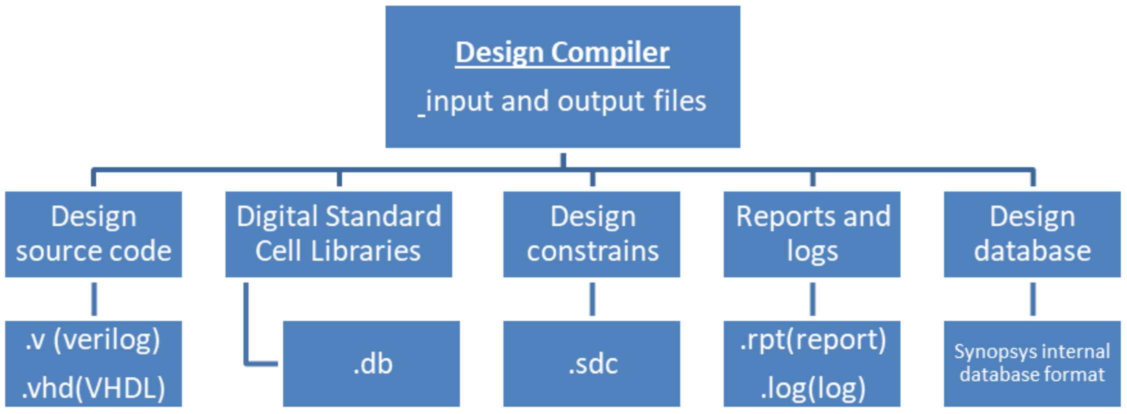


Fig. 2. Input and output files during synthesis.

Folder & file structure

```
lab01
├── inputs
│   └── johnson.sdc
├── ref
│   └── db_nldm
│       ├── saed14rvt_tt0p8v25c.db
│       └── saed14rvt_tt0p8v25c.lib
├── src
│   ├── johnson_dft.v
│   └── johnson.v
└── syn
    ├── Makefile
    ├── scripts
    │   ├── compile_dft.tcl
    │   └── compile.tcl
    └── .synopsys_dc.setup
```

- `src` contains the HDL designs
- `ref` contains the library for synthesis
- `inputs` contains input files for synthesis and some other steps
- `syn` contains the scripts for synthesis

Before you begin

- Copy lab01 and change to lab01/syn

```
$ mkdir $HOME/icdesign/m3
$ cp -a /home/tools/synopsys/m3/lab01 $HOME/icdesign/m3
$ cd $HOME/icdesign/m3/lab01/syn
$ source /home/tools/synopsys/env.sh
```

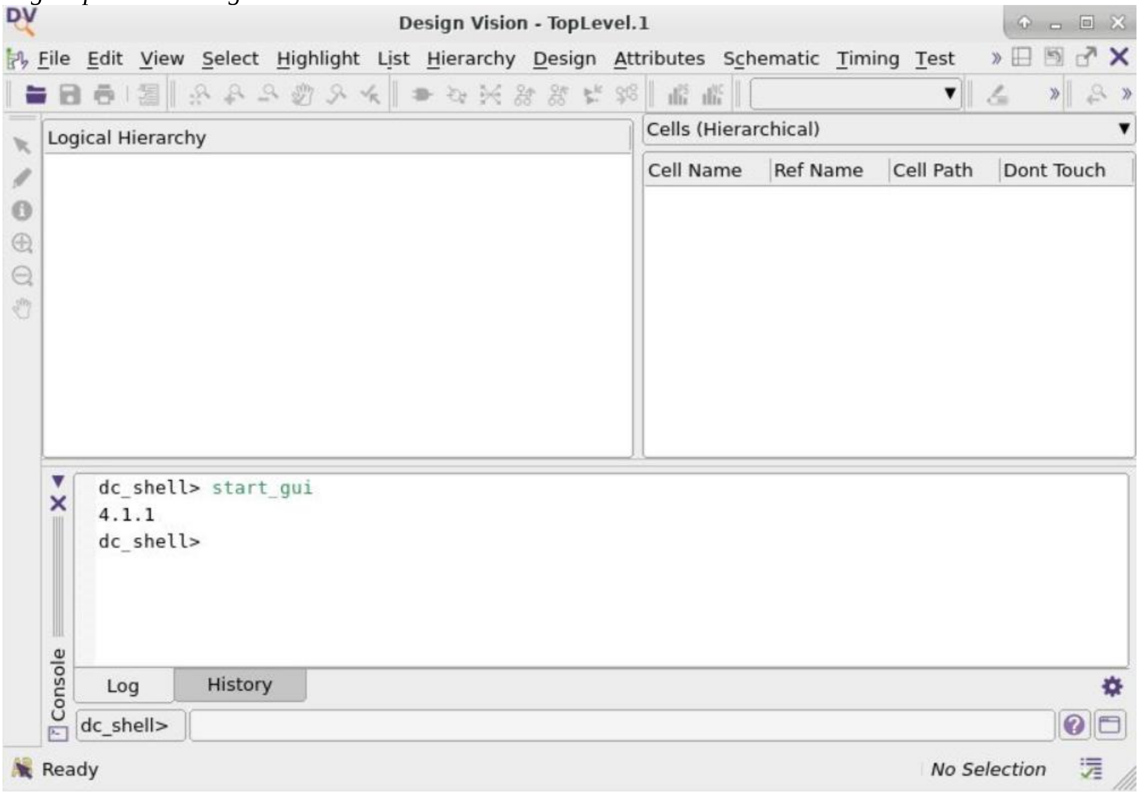
1. Start Design Compiler graphical user interface (GUI) from the **syn** directory. To start it use the following command:

```
$ dc_shell
dc_shell> start_gui
```

Hoặc

```
$ dc_shell -gui
```

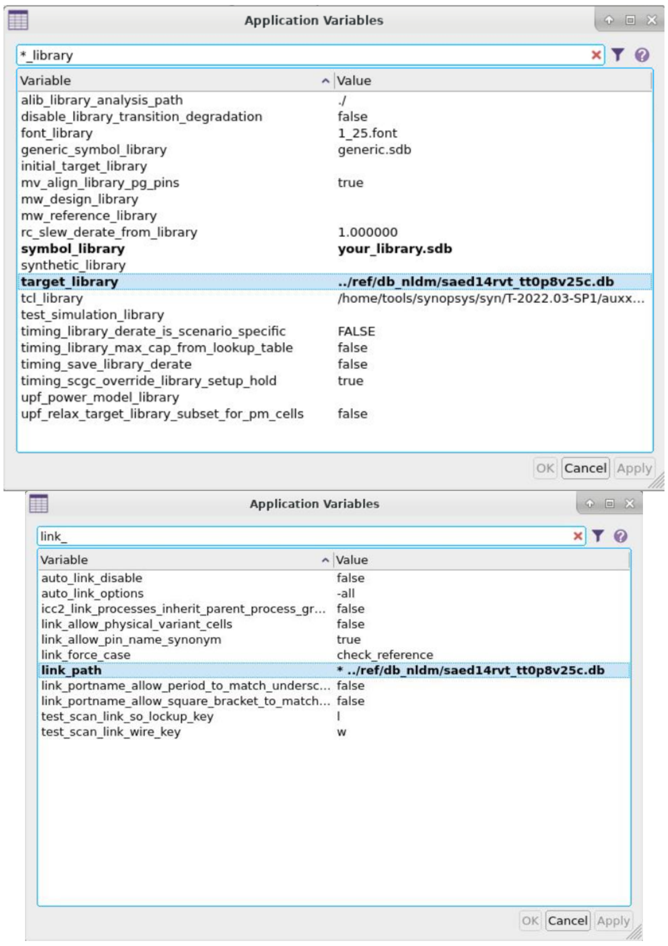
This opens the Design Compiler top-level GUI window (Design Vision). (Fig. 3)


Design Vision - TopLevel.1
File Edit View Select Highlight List Hierarchy Design Attributes Schematic Timing Test
Logical Hierarchy
Cells (Hierarchical)
Cell Name
Ref Name
Cell Path
Dont Touch
Console
Log
History
Ready
No Selection

```
dc_shell> start_gui
4.1.1
dc_shell>
```

Fig. 3. Design Compiler Top-level GUI window

2. When Design Compiler starts it automatically reads “.synopsys_dc.setup” file from the current directory. In this lab this file is located in the **syn** directory. Start Design Compiler from that directory to automatically read the file and set up the libraries. To check whether the libraries were set open **File> Setup** window (Fig. 4).



*_library

Variable	Value
alib_library_analysis_path	./
disable_library_transition_degradation	false
font_library	1_25.font
generic_symbol_library	generic.sdb
initial_target_library	
mv_align_library_pg_pins	true
mw_design_library	
mw_reference_library	
rc_slew_derate_from_library	1.000000
symbol_library	**your_library.sdb**
synthetic_library	
target_library	**../ref/db_nldm/saed14rvt_tt0p8v25c.db**
tcl_library	/home/tools/synopsys/syn/T-2022.03-SP1/auxx...
test_simulation_library	
timing_library_derate_is_scenario_specific	FALSE
timing_library_max_cap_from_lookup_table	false
timing_save_library_derate	false
timing_scgc_override_library_setup_hold	true
upf_power_model_library	
upf_relax_target_library_subset_for_pm_cells	false

link_

Variable	Value
auto_link_disable	false
auto_link_options	-all
icc2_link_processes_inherit_parent_process_gr...	false
link_allow_physical_variant_cells	false
link_allow_pin_name_synonym	true
link_force_case	check_reference
link_path	*** ../ref/db_nldm/saed14rvt_tt0p8v25c.db**
link_portname_allow_period_to_match_undersc...	false
link_portname_allow_square_bracket_to_match...	false
test_scan_link_so_lockup_key	l
test_scan_link_wire_key	w

Fig. 4. Application setup window.

3. Use the script created to carry out complex commands. The script is in the **syn/scripts** directory. The name of this script: **compile.tcl.** To source the script, write:

```
dc_shell> source scripts/compile.tcl
```

Now show all the steps of the **compile.tcl** script.

3.1. Read design commands are derived from the read_file -format VHDL or read_file -format Verilog commands (Fig. 5). In the design, format Verilog was used (Fig.4).

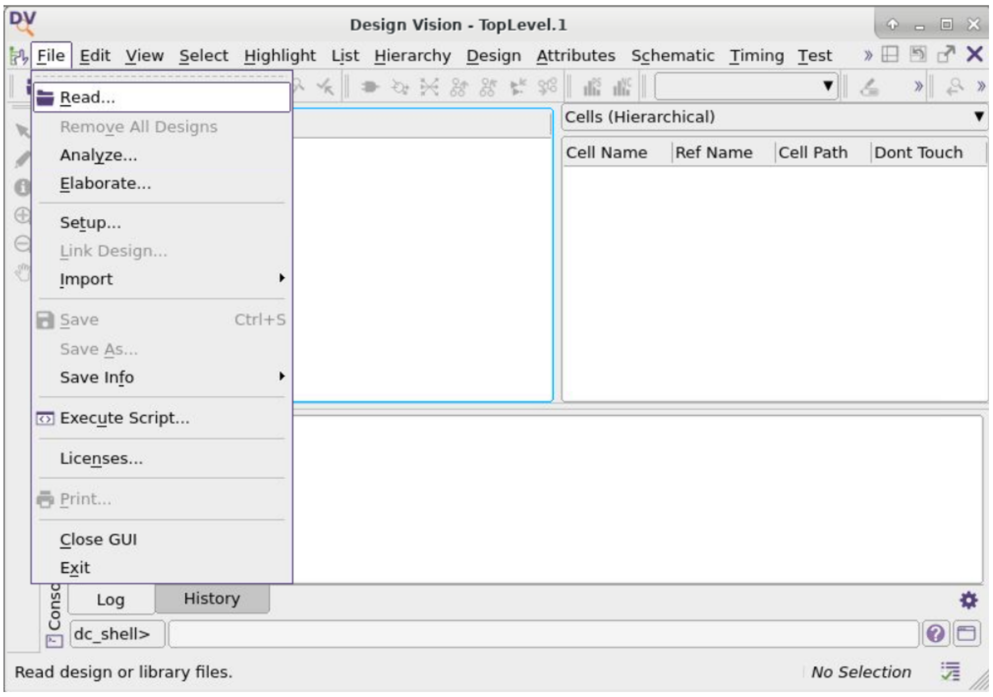


Fig. 5. Read design.

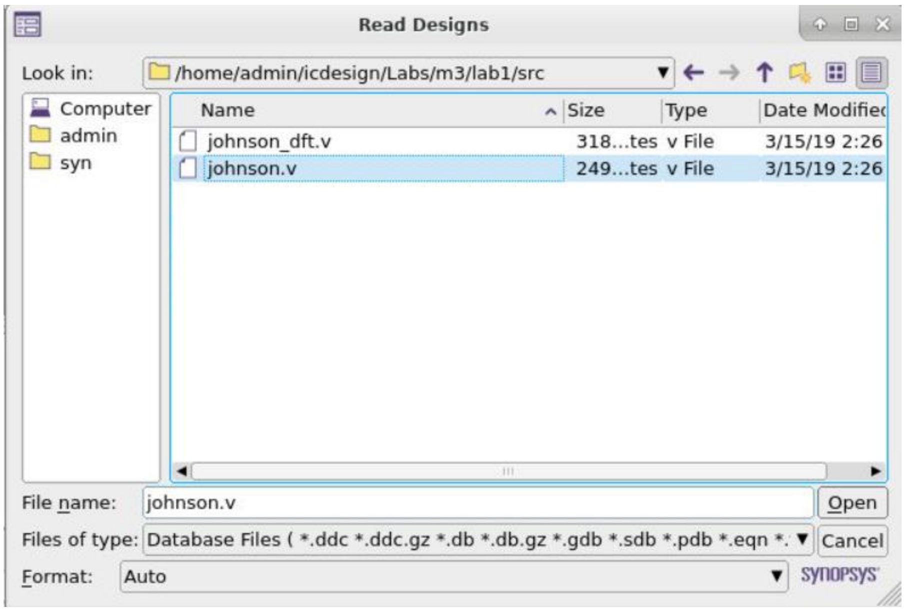

Read Designs
Look in: /home/admin/icdesign/Labs/m3/lab1/src
Computer
admin
syn
Name
Size
Type
Date Modified
johnson_dft.v 318...tes v File 3/15/19 2:26
johnson.v 249...tes v File 3/15/19 2:26
File name: johnson.v
Open
Files of type: Database Files (*.ddc *.ddc.gz *.db *.db.gz *.gdb *.sdb *.pdb *.eqn *.
Cancel
Format: Auto

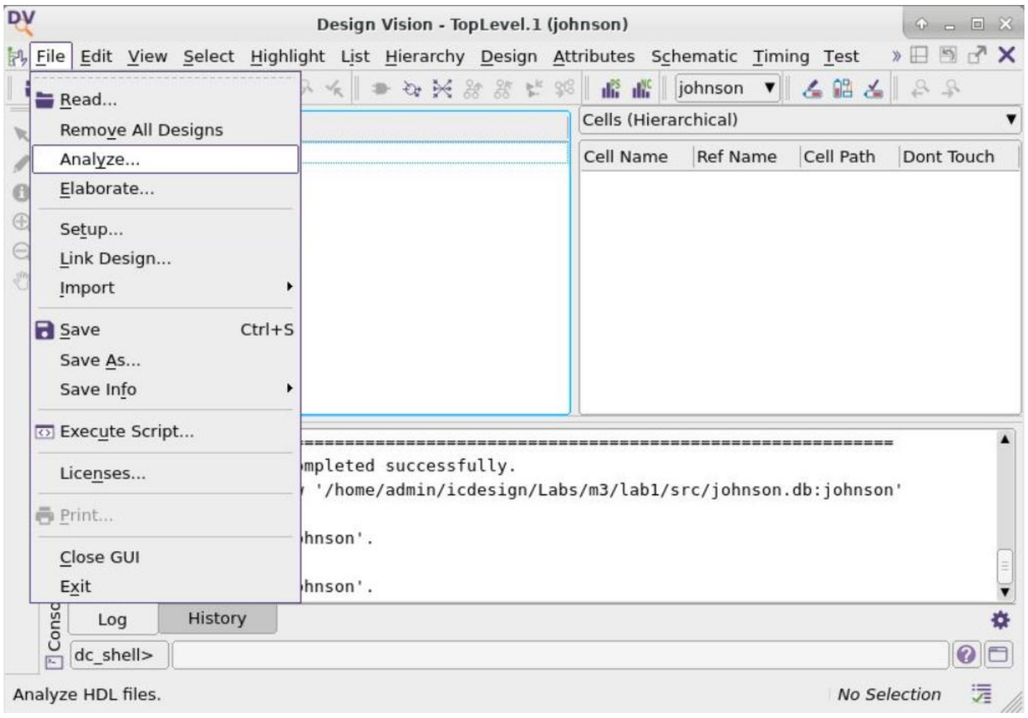
Fig. 6. Read designs box.

3.2. Then use the **analyze** and **elaborate** commands.
The analyze command does the following:
- Reads an HDL source file
- Checks it for errors (without building generic logic for the design)
- Creates HDL library objects in an HDL-independent intermediate format
- Stores the intermediate files in a defined location

If the analyze command reports errors, fix them in the HDL source file and run analyze again. After a design is analyzed, reanalyze it only when it changes (Fig. 7 and Fig. 8).

Analyze command in Design Compiler shell:

```
dc_shell> analyze -library WORK -format verilog {../src/johnson.v}
```

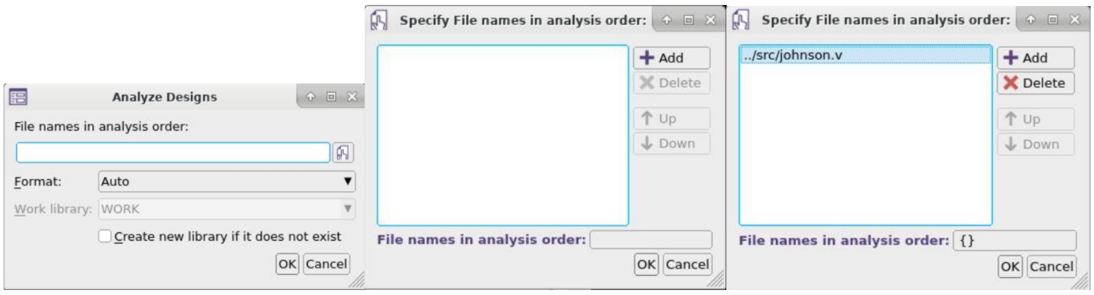


```
==============================================================
mpleted successfully.
 '/home/admin/icdesign/Labs/m3/lab1/src/johnson.db:johnson'

hnson'.

hnson'.
```

Fig. 7. Analyze designs


Analyze Designs
File names in analysis order:
Format:
Work library:
Create new library if it does not exist
OK
Cancel
Specify File names in analysis order:
Add
Delete
Up
Down
File names in analysis order:
OK
Cancel
Specify File names in analysis order:
Add
Delete
Up
Down
File names in analysis order:
OK
Cancel

Format: Auto
Work library: WORK
../src/johnson.v
File names in analysis order: {}

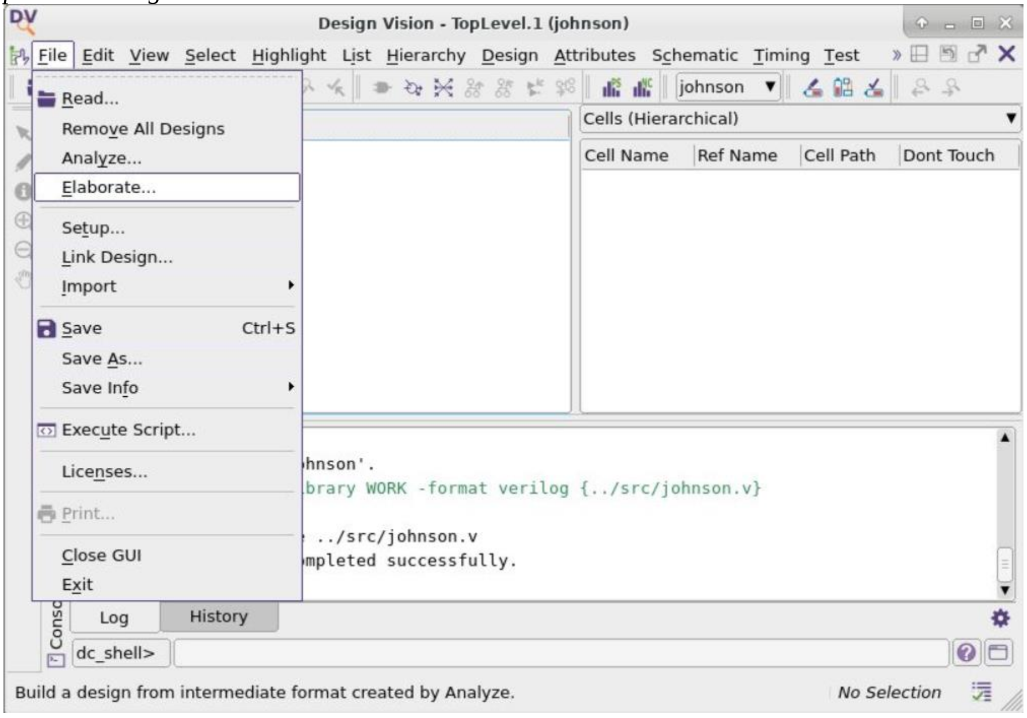
Fig. 8. Analyze designs box

3.3. The **elaborate** command does the following:

- Translates the design into a technology-independent design (GTECH) from the intermediate files produced during analysis
- Allows changing of parameter values defined in the source code
- Allows VHDL architecture selection
- Replaces the HDL arithmetic operators in the code with DesignWare components
- Automatically executes the link command, which resolves design references
- Use options to the elaborate command as in Fig. 9 and Fig. 10

Elaborate command in design compiler shell:

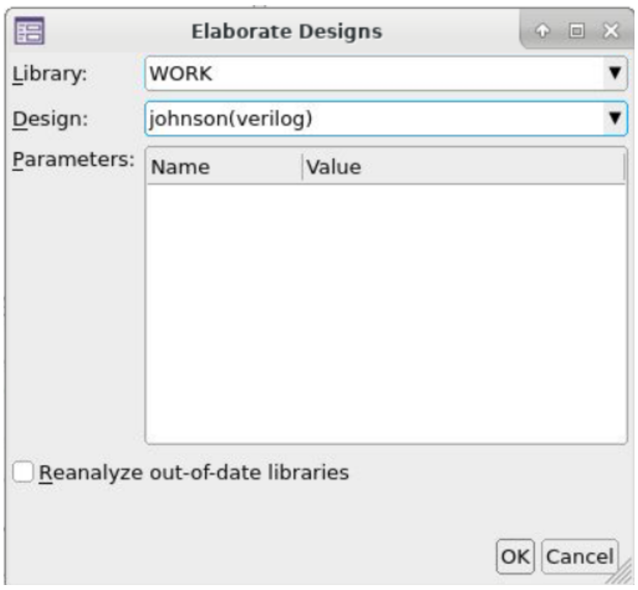
```
dc_shell> elaborate johnson -architecture verilog -library WORK
```



```
hnson'.
brary WORK -format verilog {../src/johnson.v}

 ../src/johnson.v
mpleted successfully.
```

Fig. 9. Elaborate designs



Library: WORK
Design: johnson(verilog)
Parameters:
☐ Reanalyze out-of-date libraries

Fig. 10. Elaborate designs box

3.4. For a design to be complete, it needs to be connected to all library components and designs it references. So to perform a name-based resolution of design references for the current design, use the **link** command.

```
dc_shell> link
```

3.5. The **check_design** command checks the internal representation of the current Synopsys Design Constraints for consistency, and issues error and warning messages as appropriate.

```
dc_shell> check_design
```

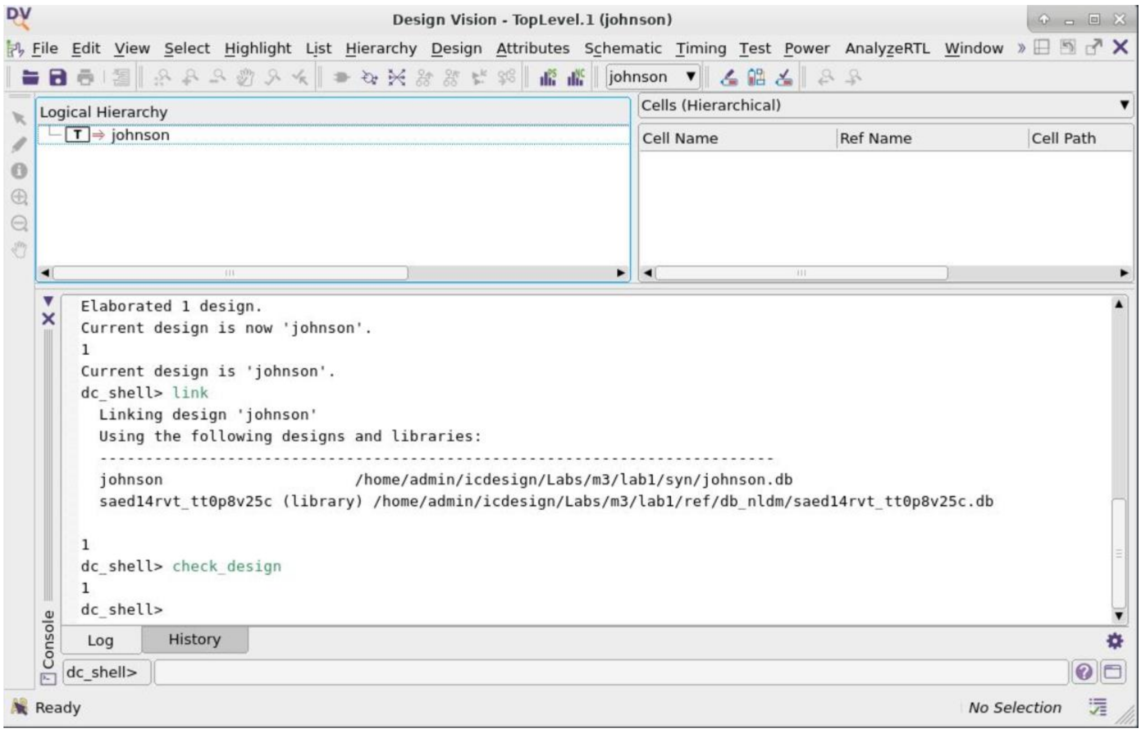
3.6. Set design optimization constraints, read johnson.sdc.
The optimization constraints comprise:

- Timing constraints (performance and speed)
- Input and output delays (synchronous paths)
- Minimum and maximum delay (asynchronous paths)
- Maximum area (number of gates)

To read constraints with .sdc (Synopsys Design Constraints) format:

```
dc_shell> read_sdc ../inputs/johnson.sdc
```

After all the above mentioned steps, the design view from Design Compiler is shown in Fig. 11.


Design Vision - TopLevel.1 (johnson)
File Edit View Select Highlight List Hierarchy Design Attributes Schematic Timing Test Power AnalyzeRTL Window
johnson
Logical Hierarchy
johnson
Cells (Hierarchical)
Cell Name
Ref Name
Cell Path
Log
History
Console
Ready
No Selection

```
Elaborated 1 design.
Current design is now 'johnson'.
1
Current design is 'johnson'.
dc_shell> link
  Linking design 'johnson'
  Using the following designs and libraries:
  --------------------------------------------------------------------------
  johnson                       /home/admin/icdesign/Labs/m3/lab1/syn/johnson.db
  saed14rvt_tt0p8v25c (library) /home/admin/icdesign/Labs/m3/lab1/ref/db_nldm/saed14rvt_tt0p8v25c.db

1
dc_shell> check_design
1
dc_shell>
```

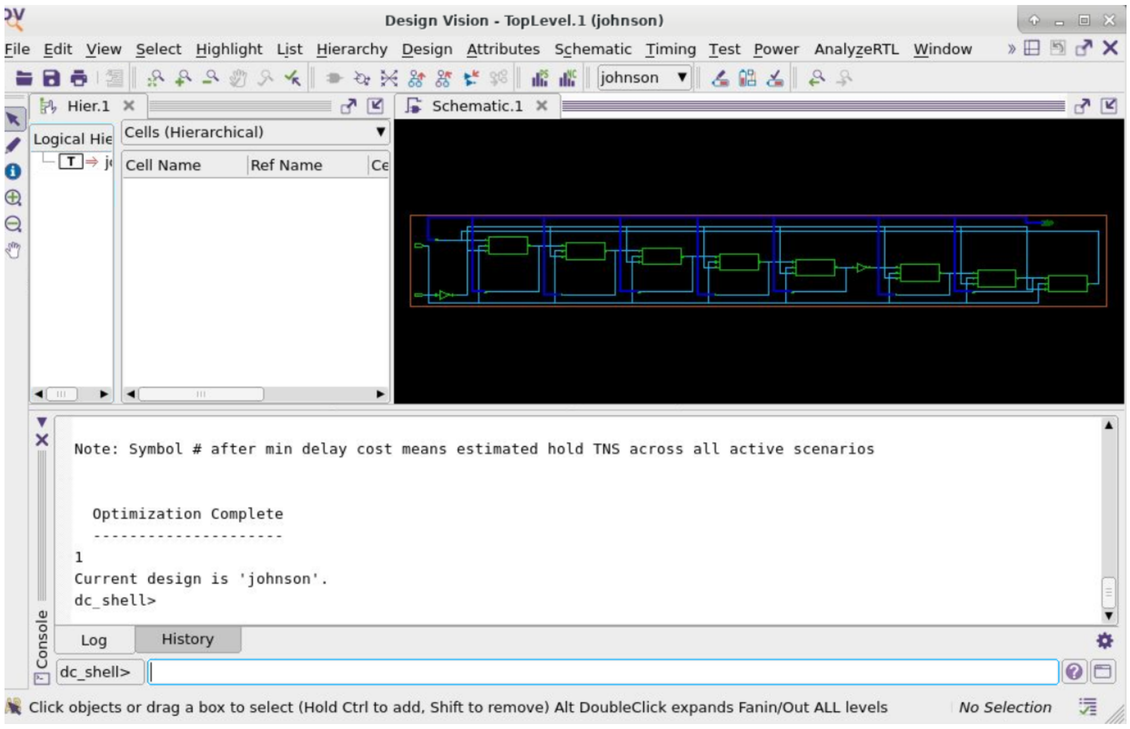
Fig. 11. Design view from Design Compiler

3.7. The **compile** command performs logic and gate-level synthesis and optimization on the current design. Optimization is controlled by user specified constraints on the design. To realize this, use the following command:

```
dc_shell> compile
```

This specifies that the sequential elements in the final design must exactly match the descriptions specified in the HDL.

Compile the design to produce a gate-level description (Fig.10). You can view the schematic by right-click on the selected design and select **Schematic view**. After that, you can double-click on Jonson design.


Design Vision - TopLevel.1 (johnson)
File Edit View Select Highlight List Hierarchy Design Attributes Schematic Timing Test Power AnalyzeRTL Window
johnson
Hier.1
Schematic.1
Logical Hie
Cells (Hierarchical)
Cell Name
Ref Name
Log
History
Console
Click objects or drag a box to select (Hold Ctrl to add, Shift to remove) Alt DoubleClick expands Fanin/Out ALL levels
No Selection

```
Note: Symbol # after min delay cost means estimated hold TNS across all active scenarios


  Optimization Complete
  ---------------------
1
Current design is 'johnson'.
dc_shell>
```

Fig.10. Design view from Design Compiler after compile

3.8. Design Compiler can generate numerous reports on the results of design synthesis and optimization. Reports are used to analyze and resolve any design problems or to improve synthesis results.

Get reports from Design Compiler

Area reports

```
dc_shell> file mkdir results
dc_shell> report_area -hierarchy > results/rpt.area.report
```

Output:

```
****************************************
Report : area
Design : johnson
Version: T-2022.03-SP1
Date   : Mon Apr 29 10:15:54 2024
****************************************

Information: Updating design information... (UID-85)
Library(s) Used:

    saed14rvt_tt0p8v25c                                                                   (File:
/home/admin/icdesign/Labs/m3/lab1/ref/db_nldm/saed14rvt_tt0p8v25c.db)

Number of ports:                           10
Number of nets:                            12
Number of cells:                           10
Number of combinational cells:              2
Number of sequential cells:                 8
Number of macros/black boxes:               0
Number of buf/inv:                          2
Number of references:                       3

Combinational area:                  0.355200
Buf/Inv area:                        0.355200
Noncombinational area:               8.524800
Macro/Black Box area:                0.000000
Net Interconnect area:               3.550286

Total cell area:                     8.880000
Total area:                         12.430287
1
```

Timing reports

```
dc_shell> report_timing > results/rpt.timing.report
```

Output:

```

****************************************
Report : timing
        -path full
        -delay max
        -max_paths 1
Design : johnson
Version: T-2022.03-SP1
Date   : Mon Apr 29 10:15:54 2024
****************************************

Operating Conditions: tt0p8v25c   Library: saed14rvt_tt0p8v25c
Wire Load Model Mode: top

  Startpoint: out_reg[7] (rising edge-triggered flip-flop clocked by clk)
  Endpoint: out[7] (output port clocked by clk)
  Path Group: clk
  Path Type: max

  Des/Clust/Port     Wire Load Model       Library
  ------------------------------------------------
  johnson            ForQA                 saed14rvt_tt0p8v25c

  Point                                                  Incr       Path
  --------------------------------------------------------------------------
  clock clk (rise edge)                                  0.00       0.00
  clock network delay (ideal)                            0.00       0.00
  out_reg[7]/CK (SAEDRVT14_FDPRBQ_V2_0P5)                0.00       0.00 r
  out_reg[7]/Q (SAEDRVT14_FDPRBQ_V2_0P5)                 0.04       0.04 r
  out[7] (out)                                           0.00       0.04 r
  data arrival time                                                 0.04

  clock clk (rise edge)                                  2.00       2.00
  clock network delay (ideal)                            0.00       2.00
  clock uncertainty                                     -0.30       1.70
  output external delay                                 -1.20       0.50
  data required time                                                0.50
  --------------------------------------------------------------------------
  data required time                                                0.50
  data arrival time                                                -0.04
  --------------------------------------------------------------------------
  slack (MET)                                                       0.46
1
```

Quality of Results (QOR) report

```
dc_shell> report_qor > results/rpt.qor.report
```

Output:

```

****************************************
Report : qor
Design : johnson
Version: T-2022.03-SP1
Date   : Mon Apr 29 10:15:54 2024
****************************************


  Timing Path Group’ clk’
  -----------------------------------
  Levels of Logic:               0.00
  Critical Path Length:          0.04
  Critical Path Slack:           0.46
  Critical Path Clk Period:      2.00
  Total Negative Slack:          0.00
  No. of Violating Paths:        0.00
  Worst Hold Violation:         -0.27
  Total Hold Violation:         -2.12
  No. of Hold Violations:        8.00
  -----------------------------------


  Cell Count
  -----------------------------------
  Hierarchical Cell Count:          0
  Hierarchical Port Count:          0
  Leaf Cell Count:                 10
  Buf/Inv Cell Count:               2
  Buf Cell Count:                   0
  Inv Cell Count:                   2
  CT Buf/Inv Cell Count:            0
  Combinational Cell Count:         2
  Sequential Cell Count:            8
  Macro Count:                      0
  -----------------------------------


  Area
  -----------------------------------
  Combinational Area:        0.355200
  Noncombinational Area:     8.524800
  Buf/Inv Area:              0.355200
  Total Buffer Area:             0.00
  Total Inverter Area:           0.36
  Macro/Black Box Area:      0.000000
  Net Area:                  3.550286
  -----------------------------------
  Cell Area:                 8.880000
  Design Area:              12.430287


  Design Rules
  -----------------------------------
  Total Number of Nets:            12
  Nets With Violations:             0
  Max Trans Violations:             0
  Max Cap Violations:               0
  -----------------------------------


  Hostname: vku-truongsa

  Compile CPU Statistics
  -----------------------------------------
  Resource Sharing:                    0.00
  Logic Optimization:                  3.77
  Mapping Optimization:                0.05
  -----------------------------------------
  Overall Compile Time:               10.44
  Overall Compile Wall Clock Time:    11.52

  --------------------------------------------------------------------

  Design  WNS: 0.00  TNS: 0.00  Number of Violating Paths: 0


  Design (Hold)  WNS: 0.27  TNS: 2.12  Number of Violating Paths: 8

  --------------------------------------------------------------------


1
```

Constraint reports

```
dc_shell> report_constraints > results/rpt.constraints.report
```

Output:

```
****************************************
Report : constraint
Design : johnson
Version: T-2022.03-SP1
```

```
Date   : Mon Apr 29 10:15:54 2024
****************************************



                                                   Weighted
    Group (max_delay/setup)      Cost     Weight     Cost
    -----------------------------------------------------
    clk                          0.00      1.00      0.00
    default                      0.00      1.00      0.00
    -----------------------------------------------------
    max_delay/setup                                  0.00

                              Total Neg  Critical
    Group (critical_range)      Slack    Endpoints   Cost
    -----------------------------------------------------
    clk                          0.00         0      0.00
    default                      0.00         0      0.00
    -----------------------------------------------------
    critical_range                                   0.00

                                                   Weighted
    Group (min_delay/hold)       Cost     Weight     Cost
    -----------------------------------------------------
    clk (no fix_hold)            0.00      1.00      0.00
    default                      0.00      1.00      0.00
    -----------------------------------------------------
    min_delay/hold                                   0.00


    Constraint                                       Cost
    -----------------------------------------------------
    min_capacitance                                  0.00 (MET)
    max_capacitance                                  0.00 (MET)
    max_delay/setup                                  0.00 (MET)
    sequential_clock_pulse_width                     0.00 (MET)
    critical_range                                   0.00 (MET)


1
```

3.9 Write a design from memory to .ddc (Synopsys internal database format) format as well as .v and .vhdl.

```
dc_shell> change_names -rule verilog
dc_shell> write -hier -format verilog -output results/johnson_mapped.v
dc_shell> write -hier -format ddc -output results/johnson_mapped.ddc
```

4. You can also reset the design to rerun the flow using the following command:

```
dc_shell> reset_design
```