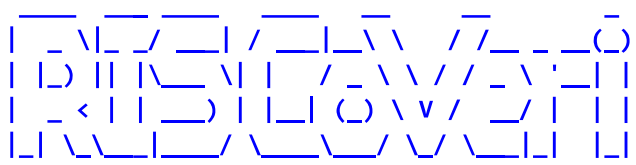

A comprehensive guide to Open-Source RTL-GDSII toolflow

Contributors:

- Sai Govardhan <saigov14@gmail.com> PES University, Bangalore
- Nandeesha Swamy <ns08360836@gmail.com> PES University, Bangalore
- Pooja R <pooj5046@gmail.com> PES University, Bangalore

Table of Contents:

Simulation
  Verilator
  Symbiosis
  cocotb
Synthesis
  Yosys
RISCoVeri
Making RISCoVeri
  Python script execute on command line:

Simulation

Simulation is a vital tool used to validate and refine electronic circuit designs. It involves creating virtual models of integrated circuits and subjecting them to various tests to access functionality, timing, and power consumption. By accurately emulating circuit behavior, simulations help identify potential issues and optimize design parameters leading to high quality and efficient VLSI designs.

Verilator

On your linux terminal, first install verilator, iverilog, gtkwave and dependencies using the apt package manage.

Verilator: Verilator is an open source hardware description language(HDL) simulator that converts Verilog code into high speed C++ or SystemC models for efficient and fast simulation of digital circuits.

```
sudo apt install verilator
```

Dependencies for verilator:

```
sudo apt-get install git help2man perl python3 make
sudo apt-get install g++  # Alternatively, clang
sudo apt-get install libgz  # Non-Ubuntu (ignore if gives error)
sudo apt-get install libfl2  # Ubuntu only (ignore if gives error)
sudo apt-get install libfl-dev  # Ubuntu only (ignore if gives error)
sudo apt-get install zlibc zlib1g zlib1g-dev  # Ubuntu only (ignore if gives error)
```

iVerilog - iVerilog is an open-source Verilog simulation and synthesis tool that is widely used for testing and validating digital designs.

```
sudo apt install iverilog
```

GTKWave - GTKWave is an open-source waveform viewer that allows users to visualize and analyze simulation results from various digital design tools.

```
sudo apt install gtkwave
```

Given the Design Files and Testbench Files, perform simulation
Note: Use the $dumpfile(“<file_name.vcd>”);$dumpvars(); commands in the testbench to capture the value change dump file, later used for viewing waveform on GTKWave.

Compiling the design and testbench to an outfile

```
iverilog -o <outfile name with .out> <design file> <testbench file>
```

Eg. iverilog -o counter_tb.out counter.v counter_tb.v

Invoking the VVP run engine:

```
vvp <outfile>
```

Eg. vvp counter_tb.out

Viewing the waveform:

```
gtkwave <.vcd file generated by the testbench>
```

Eg. gtkwave dump_counter.vcd
<Example waveform of GTKWave for a counter, with labels

Synthesis

Yosys

Yosys: Yosys is a free and open-source framework for Verilog RTL synthesis, which translates high-level hardware descriptions into low-level gate-level representations.

Graphviz Viewer:

```
sudo apt install yosys
sudo apt install gv
```

Invoking yosys

```
yosys
```

Reading the Verilog file:

```
read_verilog <design file>
```

Elaborate the design heirarchy:

```
hierarchy -top <module name>
```

Writing the design to console in yosys internal format:

```
write_ilang
```

Convert process(always blocks)to netlist elements and some simple optimizations:

```
proc; opt
```

Translating netlist to gate logic and perform some simple optimizations:

```
techmap; opt
```

Alternatively:
Instead of proc; opt; techmap;opt we can use ‘synth’

Display design netlist using xdot:

```
show -format ps -viewer gv
```

Write design netlist to a new verilog file:

```
write_verilog <verilog file>
```

Verification

SymbiYosys

SymbiYosys is an advanced open-source formal verification tool designed for RTL designs. It utilizes Yosys synthesis capabilities and supports SystemVerilog Assertions (SVA) for checking design properties, finding bugs, and ensuring correctness.

CAD Suite:

Download free OSS CAD suite

```
https://github.com/YosysHQ/oss-cad-suite-build/releases
```

Extract the archive and set the environment

```
source <extracted_location>/oss-cad-suite/environment
```

Prerequisites:

```
sudo apt-get install build-essential clang bison flex \
                     libreadline-dev gawk tcl-dev libffi-dev git \
                     graphviz xdot pkg-config python3 zlib1g-dev
```

Required Components:

Yosys Installation:

```
git clone https://github.com/YosysHQ/yosys
cd yosys
make -j$(nproc)
sudo make install
```

Sby Installation:

```
git clone https://github.com/YosysHQ/sby
cd sby
sudo make install
```

Reading the system verilog file and sby file:

```
sby <filename>.sby
```

To check for test fail or pass:

```
sby -f fifo.sby nofullskip
```

To examine the error trace:

```
gtkwave fifo_nofullskip/engine_0/trace.vcd noskip.gtkw
```