



# Intro To Verilog - Installation & Execute Instructions

21.08.2021

—

## Introduction:

- IVerilog is a command-line compiler for Verilog HDL. Generally, Verilog codes are written with some modules for each component, for example, let's take a 2x1 MUX. Each of its inputs, outputs, function, etc., is written in this module and this module can be used in all further code places, wherever needed.
- Testbench is a set of examples (test cases) for testing your code. In general, it is all  $2^n$  possible options if you are dealing with  $n$  input bits, if that turns out to be very high, you can specify your own cases.
- For multiple files, there is usually a top-level module that uses all the other modules and a testbench is defined for the top-level module (Let's say we are making a 4x1 MUX using 2 2x1 MUXes, then we write the testbench for the 4x1 MUX which automatically tests working of 2x1 MUX).
- GTKWave is a waveform visualizer that lets you examine the input and output signal states at various time instants.

## Installation:

- To install iverilog:  

```
sudo apt-get install iverilog
```
- To install GTKWave:  

```
sudo apt-get install gtkwave
```

## File Formats:

- Modules and testbenches must have the extension (.v).
- Dumpfiles should have the extension (.vcd)

## Execute:

- To compile the written code, just type  
`iverilog <fileName> <names of any included modules>`
- To execute the compiled code, as usual run  
`./a.out`
- To check the waveforms on GTKWave, use the command  
`gtkwave <dumpfileName>`
- To observe waveforms on GTKWave, select the module name in the left side section, then drag whichever variable you want to observe.

## Sample Codes:

- Clone the repository - Run this command on Ubuntu Terminal  
`git clone https://github.com/Sasanka-GRS/Verilog-Session`
- Open the files in the directory, make necessary changes and run them