GTK Wave Installation and Usage Manual

CS623: CAD for VLSI Design

Dept. of Computer Science and Engineering

Indian Institute of Technology Madras

Document prepared by:

Malika Shaik

Reconfigurable and Intelligent Systems Engineering Group

Department of Computer Science and Engineering,

Indian Institute of Technology Madras, Chennai - 600036

GTK Wave Installation and Usage Manual

Contents

1 Introduction

This document aims to set the environment for using Electronic Waveform viewer GTK Wave and to give you an idea on how to use this tool for viewing waveforms. This document is just a basic one and by no means a complete user guide.

2 What is GTKWave?

GTKWave is a electronic waveform viewer built using GTK + toolkit, which reads vcd (value change dump file) and displays waveforms. A VCD file is an ASCII file that contains header information, variable definitions, and the value changes for all variables specified in the task calls. Such a file is typically generated using simulation tools(iverilog in our case).

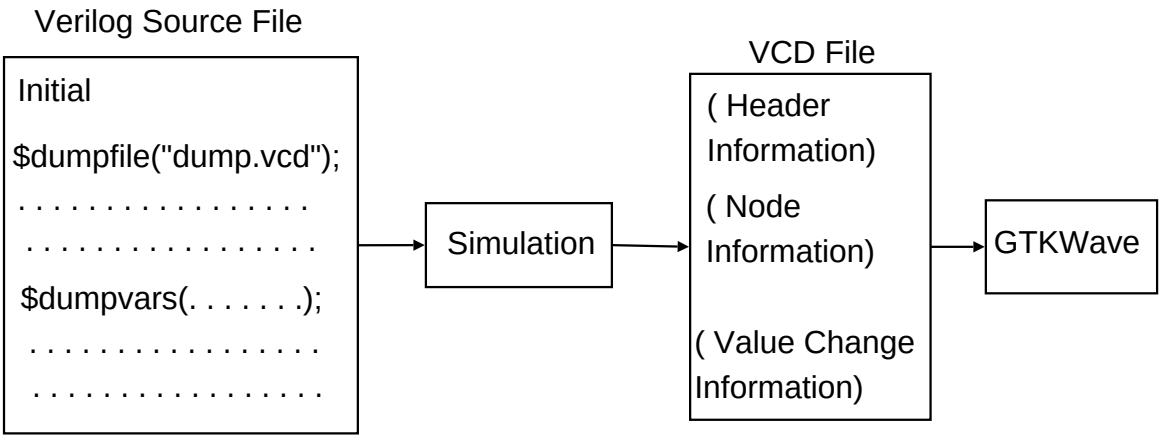

Figure 1:

3 Where can I download it?

Linux version:
http://www.geda.seul.org/download.html
Windows version:
http://www.geocities.com/SiliconValley/Campus/3216/GTKWave/gtkwave-win32.html

4 How to Install?

4.1 Using Source

To install the GTKWave from tarball execute the following commands.

- Untar gzipped tarball using the command
  tar -xzvf gtkwave - 1.3.59.tar.gz
  Now one directory called *gtkwave-1.3.59* is created with all the files required for the installation.
- *./configure −−prefix=directorypath*
  where *directorypath* is the directory path to install GTKWave.
- *make*
- *make install* (as root)

The tarball also has a $.gtkwave$ file that you must copy to your home directory or to your current VCD directory. It contains the configurations for the GTKWave.

5 Environment Customization

Customizing environment in GTKWave is quite easy with the $.gtkwave$ file. The search path for the file is first the current working directory and then the effective user's home directory. It is simply a series of variable definitions, one per line, which modify the behavior for specific features in GTKWave. The format for each line is:
variable value
Lines that start with # are considered comment lines and are ignored. Make sure that you copy the $.gtkwave$ file to your home directory or your VCD project directory. The contents of a sample $.gtkwave$ file can be found in Appendix A.

6 How to use GTK Wave ?

Now GTK wave is ready for use .

1. GTKWave expects a .vcd file that is generated from a simulation run as input. Here, we are going to assume that the input is *fibonacci.vcd* that is generated by compiling and simulating "fibonacci.v" file that is provided in the Appendix B.
2. Invoke GTKWave :
   bash-2.05a $ gtkwave filename.vcd

   A sample trace is given below. Online help is available for every menu function in GTKWave. In order to access online help, select Help-Wave Help from the menu and then select any menu option in order to see the menu option's description. Figure 2 shows the snapshot of the GTKWave window.
3. Select the option SEARCH → signal Search Tree. This will open a new window as shown in Figure 2. It provides as easy means of adding traces to the display in a text based treelike

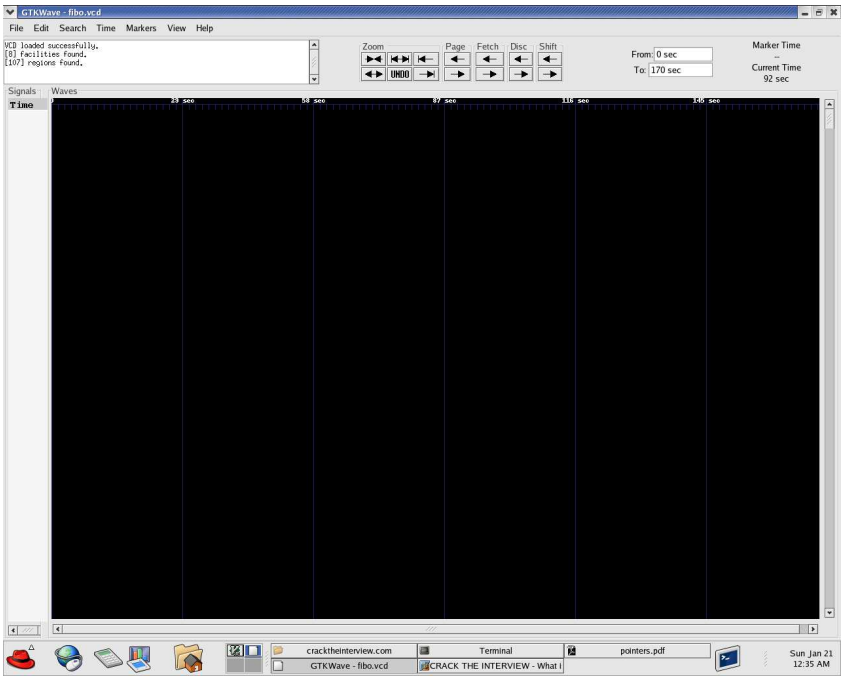

Figure 2: GTKWave Window

fashion. The main focus here is to add the signals to the display for viewing (Refer to snapshot shown in Figure 2.)

Signal Search tree provides an easy means of adding traces to the display. Various functions are provided in the Signal Search Tree that allow searching a treelike hierarchy. Now select the signals from the signal search tree and click the append button. Corresponding signals are added to the display. Similarly, we can insert or replace signals from the display using insert or replace tab (Refer to snapshot in Figure 3).

4. We are now ready for viewing and analyzing the waveforms of our design. The display window is divided into parts namely signals window and waves window. We can observe the values of the signals at any particular instant of the time clicking the mouse at that point in the signal window.

What follows below is the listing and description for the commonly used menu items

1. To print the result into a file use the option FILE → Print to File, that will ask for the name of encapsulated postscript to generate the current main window displays contents.

2. To set the display data format use the option EDIT→ Data Format. It will show a set options like Hex, decimal , binary, etc. Hex options will step through all highlighted traces and ensure that vector with this qualifiers will be displayed with hexadecimal values. Similarly we can chose other options.

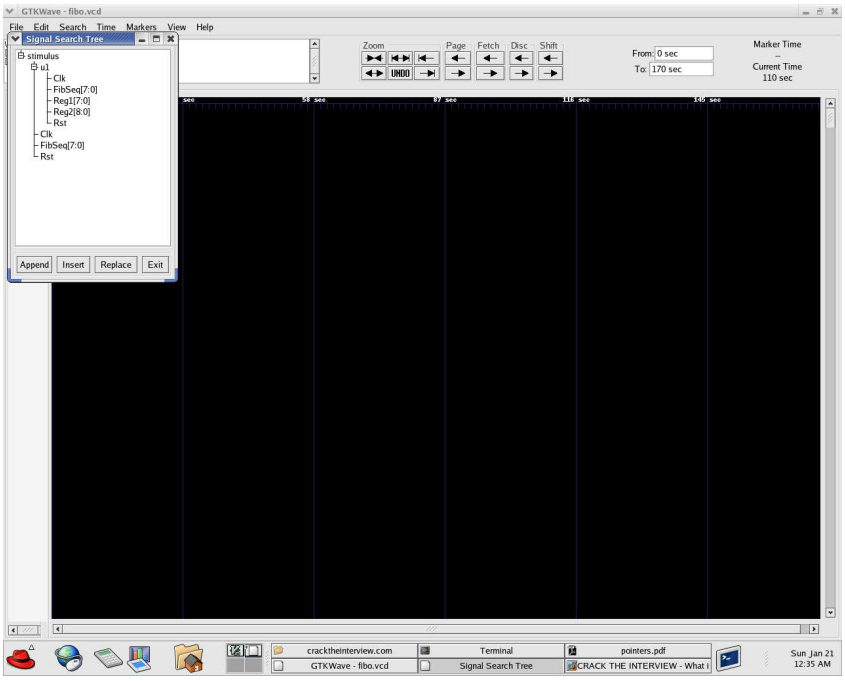

Figure 3: Adding Signals to GTKWave

3. We can search for waveforms using the option SEARCH → pattern Search. Pattern search only works when at least one trace is highlighted (by clicking the signal). A request will appear that lists all the selected traces and allow various criteria to be specified for each trace. Search can go forward of backward from the primary unnamed marker (Refer to snapshot shown in Figure 2).

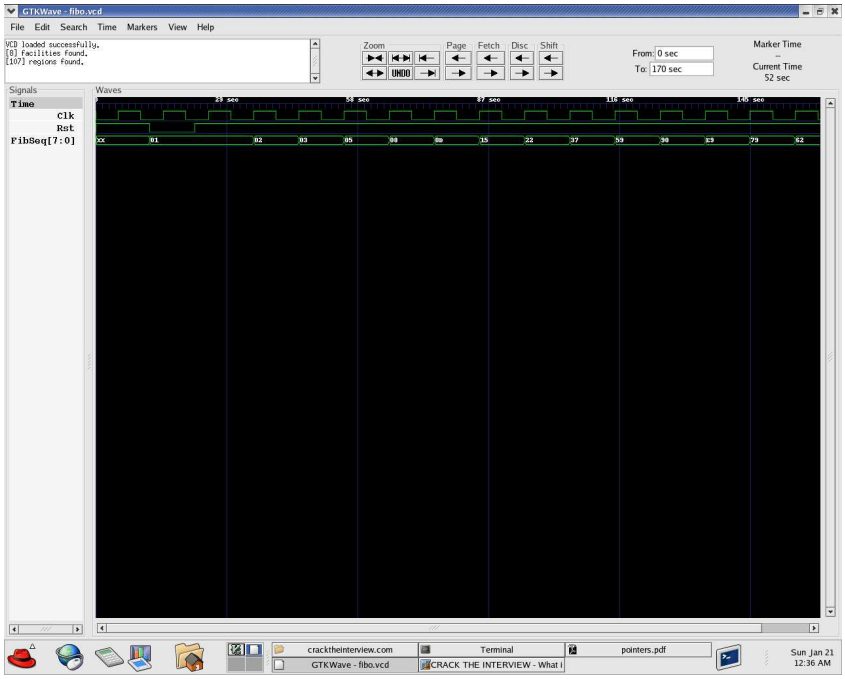

Figure 4: Insert or Replace Signals

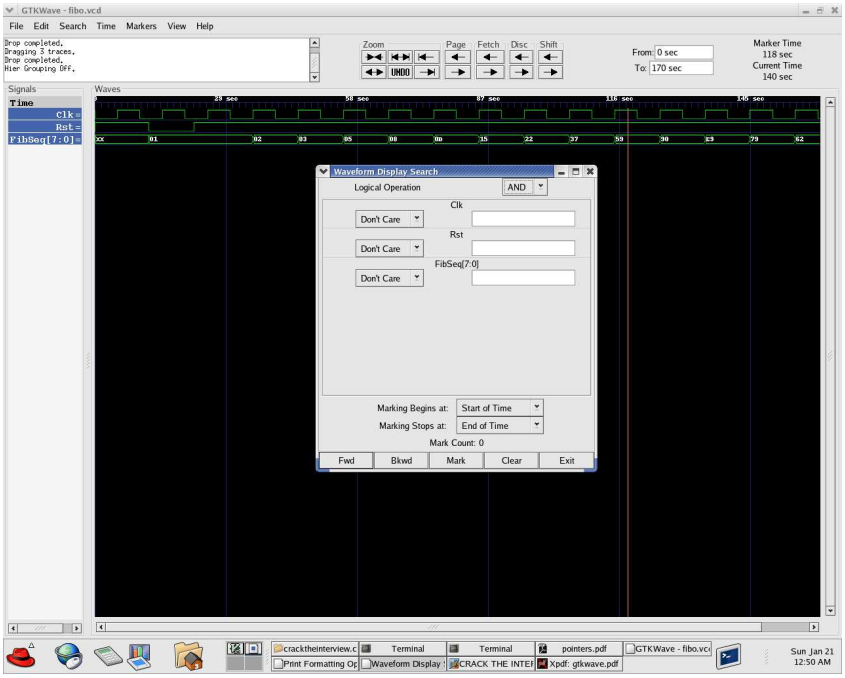

Figure 5: Searching in GTKWave

7 Appendix A - .gtkwaverc file

```
#
# sample rc file
#
hier_max_level 1
force_toolbars 0
#cursor_snap 8

dynamic_resizing 1
hpane_pack 1
use_aet 0
use_vcd 0
#initial_window_x 700
#initial_window_y 400
#initial_window_xpos 50
#initial_window_ypos 50

use_maxtime_display 0

enable_vcd_autosave 0
use_roundcaps 1

use_nonprop_fonts yes
enable_horiz_grid yes
use_big_fonts no
constant_marker_update yes
show_grid yes
show_base_symbols no
use_roundcaps yes

atomic_vectors yes
vcd_explicit_zero_subscripts no

#
# color additions
#
color_back   000000
color_grid   202070
color_high   00ff00
color_low    008000
```

```
color_trans  00c000
color_mid    c0c000

color_value  ffffff
color_vbox   00ff00
color_vtrans 00c000

color_x      00ff00
color_xfill  004000

color_umark  ff8080
color_mark   ffff80

color_time   ffffff
color_timeb  000000
```

8 Appendix B - fibonacci.v, fibonacci_tb.v

The source code of the files fibonacci.v and fibonacci_tb.v (test bench) can be found on
http://www.cse.iitm.ac.in/∼shankar/teaching/cs623/download.html

9 Appendix C - Simulation Results

The file *fibonacci.v* is compiled and executed by the following commands.
bash-2.05b$ iverilog fibonacci_tb.v
The command above will generate two files, fibonacci.vcd and a.out. **a.out** is an executable file. The Value Change Dump file for fibonacci.v is fibonacci.vcd (as mentioned in the test bench). This command will display the output on your screen as follows.

VCD info: dumpfile fibonacci.vcd opened for output.
0 : FibSeq = x
12 : FibSeq = 0
25 : FibSeq = 1
45 : FibSeq = 2
55 : FibSeq = 3
65 : FibSeq = 5
75 : FibSeq = 8
85 : FibSeq = 13
95 : FibSeq = 21
105 : FibSeq = 34
115 : FibSeq = 55
125 : FibSeq = 89
135 : FibSeq = 144
145 : FibSeq = 233
out of range ERROR
155 : FibSeq = 121
165 : FibSeq = 98