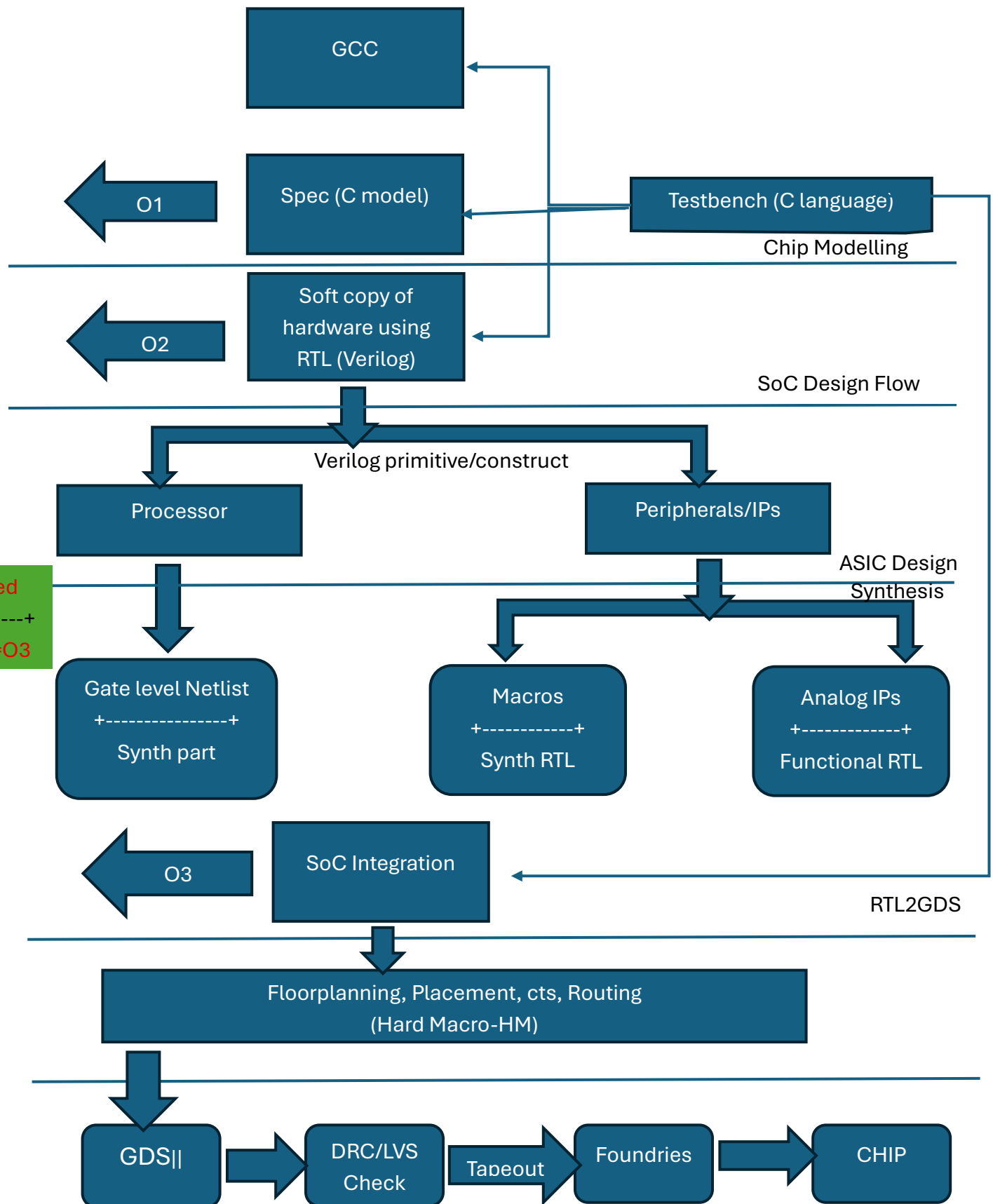


## WORK FLOW OF DIGITAL VLSI SOC DESIGN AND PLANNING:



**BLOCK DIAGRAM EXPLANATION:** Our end goal is that particular application must run on the chip.

1. **GCC** - Acts as the compiler front end, turning C-based specifications into executable artifacts like testbenches and initial chip models.

2. **Spec (C model)** - Defines the chip's intended functionality in C, serving as the golden reference for both modeling and verification.

**Outputs: • O1:** here we are checking that our application itself is correct or not. After this my spec are freeze and our very first step done.

3. **Soft copy of hardware using RTL** - Translates the C model into a synthesizable hardware description, capturing data paths and control logic at the register-transfer level.

**Outputs: • O2:** output from RTL architect and O1 must be equal to O2 then functionality retain.

4. **processor** - The central compute engine, responsible for executing instructions and orchestrating data movement within the SoC.

**Outputs: • Gate level Netlist** - The synthesized netlist of standard cells and interconnects, ready for timing analysis and placement.

5. **Peripherals/IPs** - pre-designed functional blocks (e.g., UART, SPI, timers) that interface the processor with external components.

**Outputs: • Macros (Synthesizable RTL):** Larger pre-characterized blocks (e.g., memory, DSP units) integrated at the RTL level and synthesized alongside other logic.

• **Analog Ips (Functional RTL):** Translates analog building blocks into behavioral RTL models for inclusion in the digital design flow.

6. **SoC Integration** - The stage where all digital macros, analog IPs, and the processor-peripheral subsystem are assembled onto a single system-on-chip, with bus fabrics and power/clock domains defined.

**Outputs: • O3:** We want  $O1=O2=O3$ ; Denotes equivalence or alignment checkpoints between specification, RTL, and SoC integration stage.

7. **Hard Macro (HM): Floorplanning, Placement, CTS, Routing** - Physical implementation steps for large, fixed-layout blocks: floorplanning and placement determine macro locations; clock tree synthesis (CTS) and routing establish timing-constrained interconnects.

**8. GDSII** - The final database format representing the chip's physical layout, containing polygons and layers that describe every transistor, wire, and via for mask generation.

**9. DRC / LVS Check Design Rule Check** - ensures layout complies with foundry manufacturing constraints; Layout Versus Schematic verifies the physical layout matches the intended netlist connectivity.

**10. Tape out** - The process of finalizing and submitting the GDSII files to the foundry, marking the transition from digital design to physical mask production.

**11. Foundries & CHIP** - Fabrication facilities manufacture wafers based on the submitted masks; the resulting silicon is packaged, tested, and delivered as the finished ASIC product.

The screenshot shows a terminal window on a Linux system. The title bar indicates the date and time as 'Sep 20 00:53'. The terminal prompt is 'root@Ubuntu: ~/yosys'. The user has run 'yosys --help' and 'yosys --license', which displays the copyright information and license for Yosys. The license is a standard 'as is' disclaimer. The user has also run 'yosys --help' again, which shows the usage instructions. The terminal output is as follows:

```
root@Ubuntu: ~/yosys
yosys --help
Copyright (C) 2012 - 2025 Claire Xenia Wolf <claire@yosyshq.com>
Distributed under an ISC-like license, type 'license' to see terms
Yosys 0.57+148 (git sha1 259bdfb3, g++ 9.4.0-1ubuntu1-20.04.2 -fPIC -O3)

yosys> exit
End of script. Logfile hash: da39a3ee5e. CPU: user 0.00s system 0.00s, MEM: 14.39 MB peak
Yosys 0.57+148 (git sha1 259bdfb3, g++ 9.4.0-1ubuntu1-20.04.2 -fPIC -O3)
Time spent: no commands executed
root@Ubuntu: ~/yosys# license
Command 'license' not found, did you mean:
command 'licensee' from deb ruby-licensee (8.9.2-1)
Try: apt install <deb name>

root@Ubuntu: ~/yosys# yosys
yosys -- Yosys Open SYnthesis Suite
Copyright (C) 2012 - 2025 Claire Xenia Wolf <claire@yosyshq.com>
Distributed under an ISC-like license, type 'license' to see terms
Yosys 0.57+148 (git sha1 259bdfb3, g++ 9.4.0-1ubuntu1-20.04.2 -fPIC -O3)

yosys> license
yosys -- Yosys Open SYnthesis Suite
Copyright (C) 2012 - 2025 Claire Xenia Wolf <claire@yosyshq.com>

Permission to use, copy, modify, and/or distribute this software for any
purpose with or without fee is hereby granted, provided that the above
copyright notice and this permission notice appear in all copies.

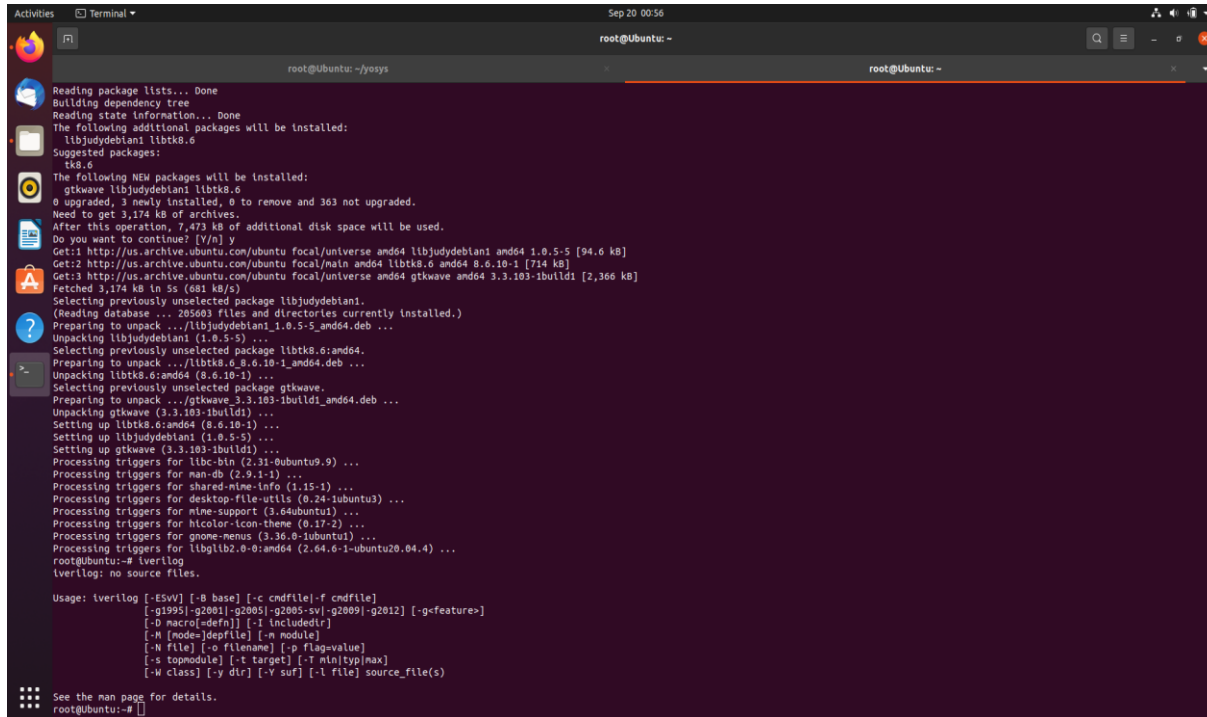
THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR
ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN
ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF
OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.
```

**Iverilog:** # sudo apt-get update

# sudo apt-get install iverilog

**GTKWave:** # sudo apt-get update

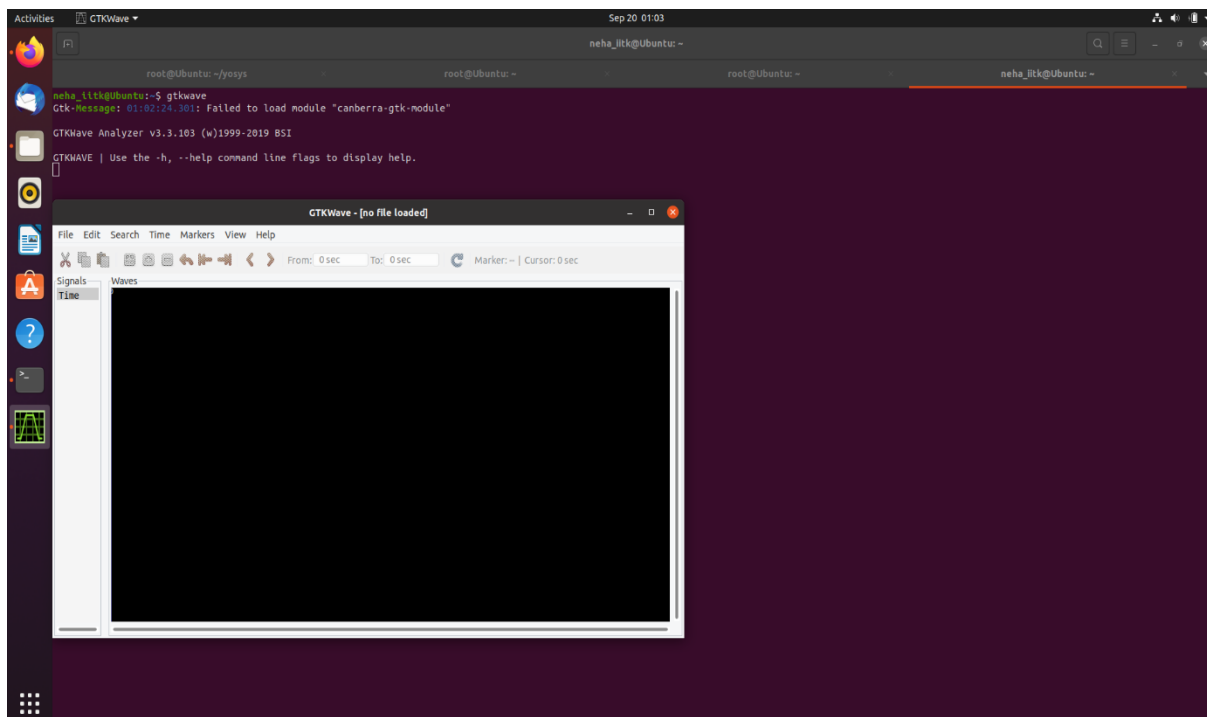
# sudo apt install gtkwave



```
root@Ubuntu: ~# sudo apt-get update
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  libjudydebian1 libtk8.6
Suggested packages:
  tk8.6
The following NEW packages will be installed:
  gtkwave libjudydebian1 libtk8.6
0 upgraded, 3 newly installed, 0 to remove and 363 not upgraded.
Need to get 3,174 kB of archives.
After this operation, 7,473 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 libjudydebian1 amd64 1.0.5-5 [94.6 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu focal/main amd64 libtk8.6 amd64 8.6.10-1 [714 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu focal/universe amd64 gtkwave amd64 3.3.103-1build1 [2,366 kB]
Fetched 3,174 kB in 5s (661 kB/s)
Selecting previously unselected package libjudydebian1.
(Reading database ... 205603 files and directories currently installed.)
Preparing to unpack .../libjudydebian1_1.0.5-5_amd64.deb ...
Unpacking libjudydebian1 (1.0.5-5) ...
Selecting previously unselected package libtk8.6:amd64.
Preparing to unpack .../libtk8.6_8.6.10-1_amd64.deb ...
Unpacking libtk8.6:amd64 (8.6.10-1) ...
Selecting previously unselected package gtkwave.
Preparing to unpack .../gtkwave_3.3.103-1build1_amd64.deb ...
Unpacking gtkwave (3.3.103-1build1) ...
Setting up libtk8.6:amd64 (8.6.10-1) ...
Setting up libjudydebian1 (1.0.5-5) ...
Setting up gtkwave (3.3.103-1build1) ...
Processing triggers for libc-bin (2.31-0ubuntu9.9) ...
Processing triggers for man-db (2.9.1-1) ...
Processing triggers for shared-mime-info (1.15-1) ...
Processing triggers for desktop-file-utils (0.24-1ubuntu3) ...
Processing triggers for mime-support (3.64ubuntu1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for gnome-menus (3.36.0-1ubuntu1) ...
Processing triggers for libgl1:amd64 (2.64.6-1-ubuntu20.04.4) ...
root@Ubuntu:~# iverilog
iverilog: no source files.

Usage: iverilog [-B base] [-c cmdfile] [-f cmdfile]
             [-g1995] [-g2001] [-g2005] [-g2005-sv] [-g2009] [-g2012] [-g<feature>]
             [-O macro[=defn]] [-I includedir]
             [-M [nodes]depfile] [-m module]
             [-N file] [-o filename] [-p flag=value]
             [-s toplevel] [-t target] [-T min|typ|max]
             [-W class] [-y dir] [-Y suf] [-l file] source_file(s)

See the man page for details.
root@Ubuntu:~#
```



```
neha_ltk@Ubuntu:~$ gtkwave
Gtk+Message: 01:02:24.301: Failed to load module "canberra-gtk-module"
GTKWave Analyzer v3.3.103 (w)1999-2019 BSI
GTKWave | Use the -h, --help command line flags to display help.

GTKWave - [no file loaded]
File Edit Search Time Markers View Help
From: 0 sec To: 0 sec Marker: - Cursor: 0 sec
```