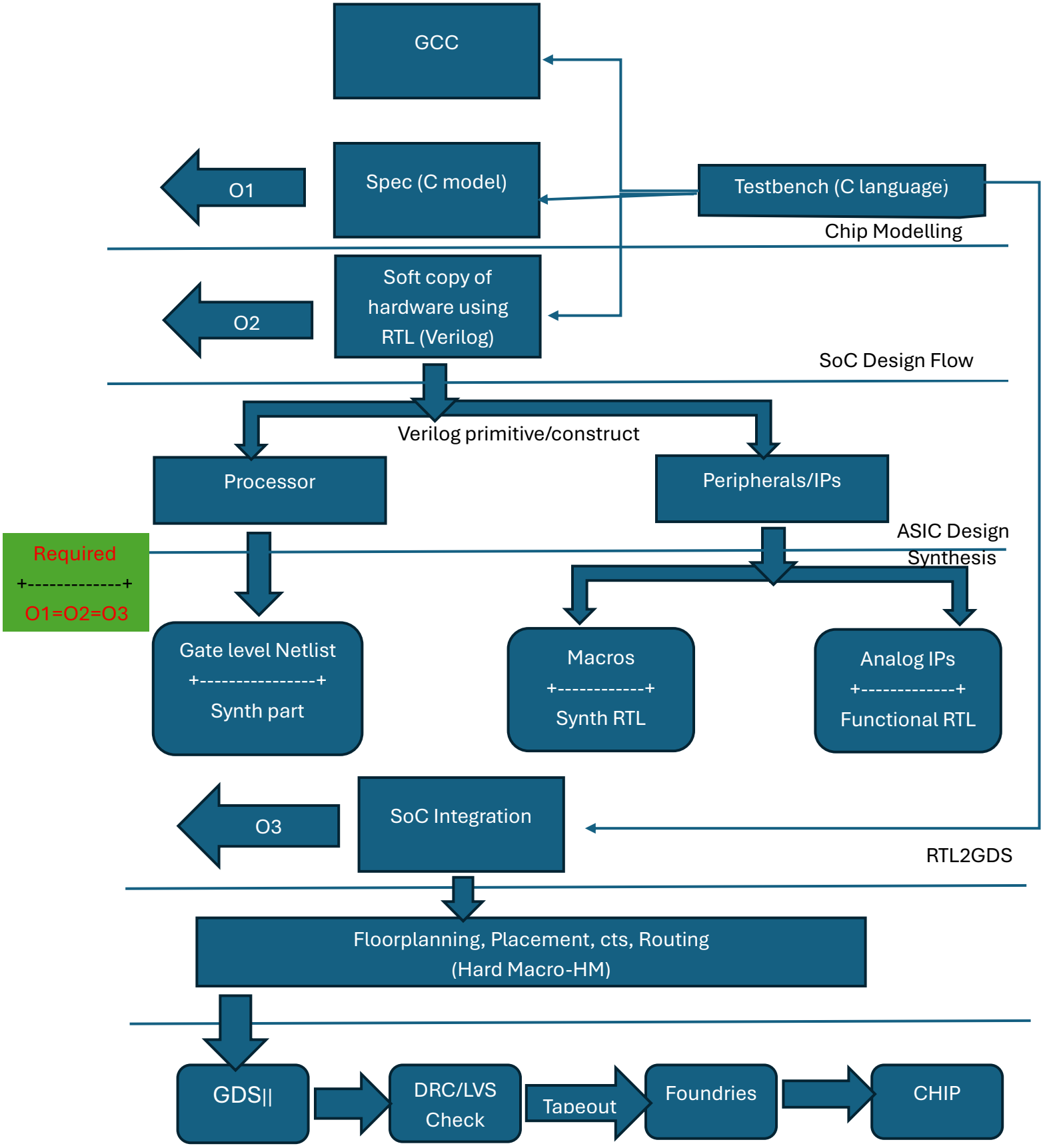
WORK FLOW OF DIGITAL VLSI SOC DESIGN AND PLANNING:



BLOCK DIAGRAM EXPLAINATION: 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.

TOOLS INSTALLATION:

YOSYS:

CODE USED FOR UBUNTU SYSTEM:

su – (for access as a root user)

sudo apt-get update

git clone https://github.com/YosysHQ/yosys.git

cd yosys

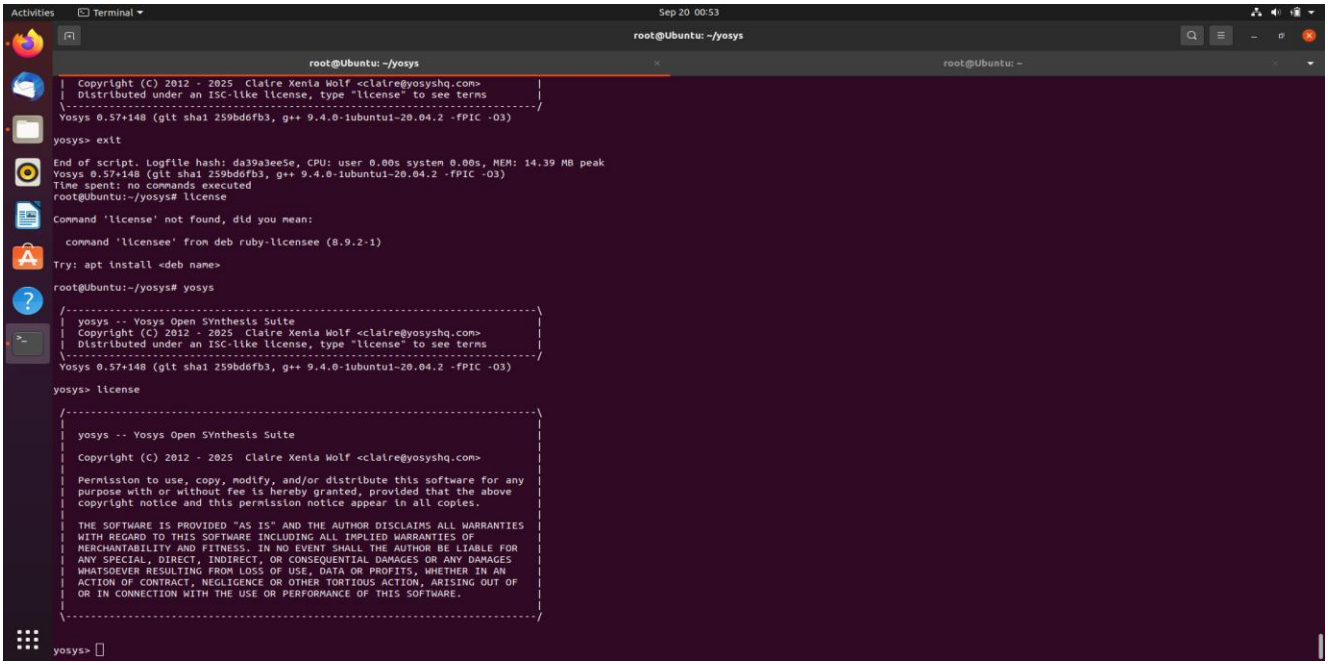
sudo apt install make (If make is not installed, please install it)

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

make config-gcc

make

sudo make install



Iverilog: # sudo apt-get update

sudo apt-get install iverilog

GTKWave: # sudo apt-get update

sudo apt install gtkwave