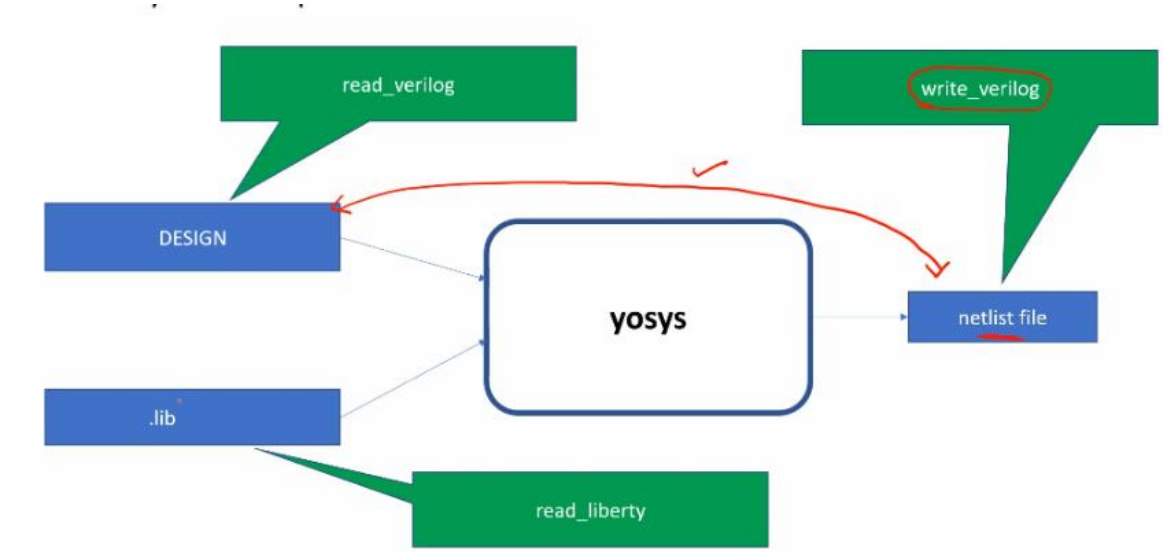
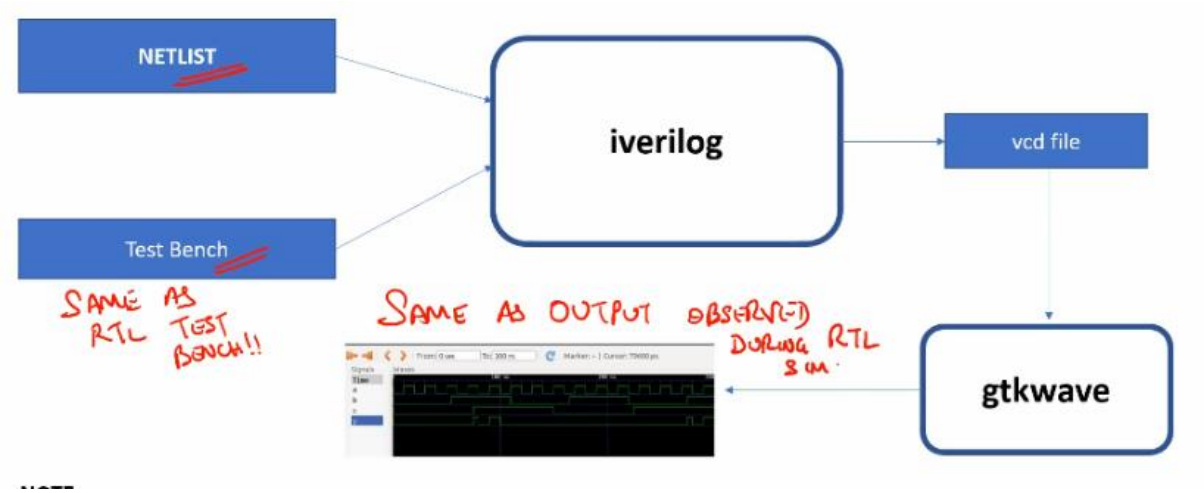
Yosys Tool

- **RTL (Register Transfer Level)** is a behavioral representation of a digital circuit, describing functionality in an HDL (such as Verilog).
- **Synthesis** is the automated process that translates RTL code into a netlist — a gate-level description using standard cells provided by a technology library (.lib).
- The **netlist** is essentially the hardware blueprint, describing the logic gates and their interconnections that together implement the specified behavior.

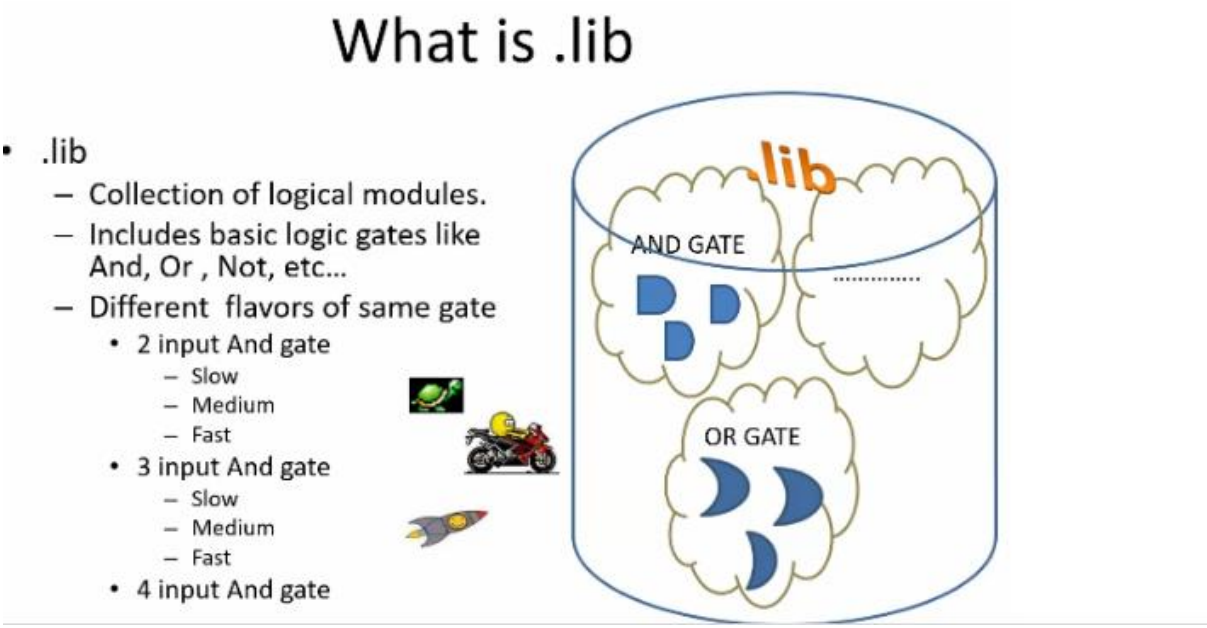


Synthesis



Role of the .lib (Standard Cell Library)

- The **.lib file** contains definitions for logic gates (standard cells) available in a given technology, including various types and flavors (e.g., 2-input AND, 3-input AND, slow/medium/fast versions).
- These cells have differing properties to allow the synthesizer to make trade-offs between speed, power, and area.
- The library must be sufficiently comprehensive to implement any Boolean logic, as NAND and NOR gates are functionally complete.



Timing Constraints: Setup and Hold

- Timing analysis in digital circuits revolves around two critical constraints: setup and hold.
- **Setup time**: Minimum interval before the clock edge during which input data must be stable.
- **Hold time**: Minimum interval after the clock edge during which input data must remain stable.

For a pipeline with flip-flops (A and B) separated by combinational logic:

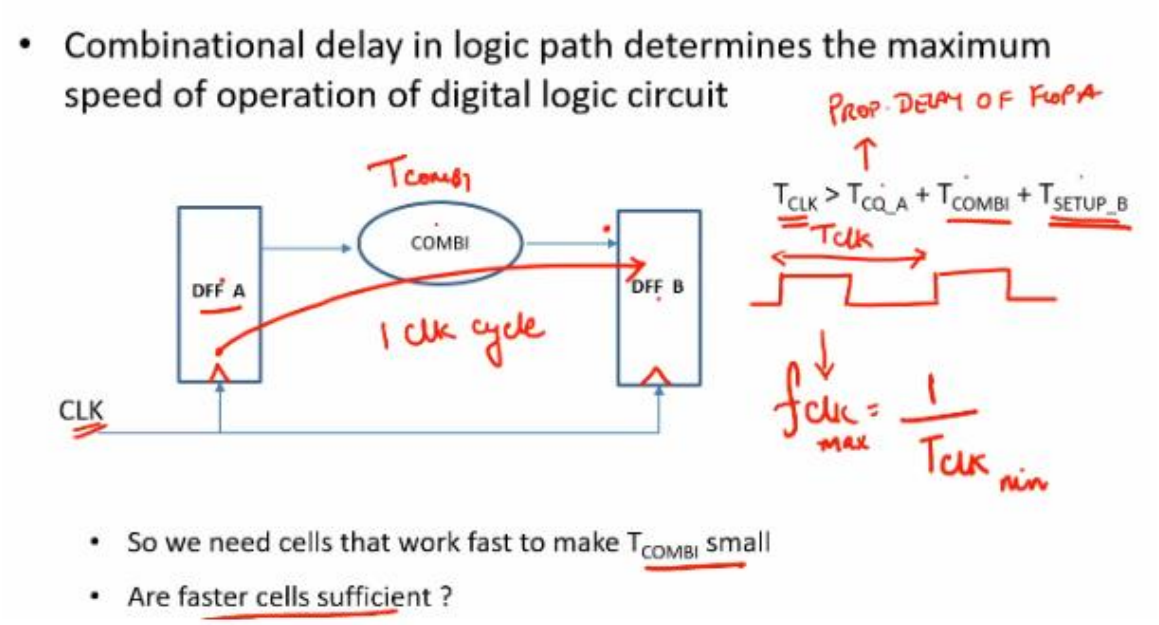
- The **clock period (tclock)** must satisfy:

  tclock≥tCQA+tcomb+tsetup

  - tCQA: Propagation delay from flip-flop A's clock to its output (clock-to-Q).
  - tcomb: Propagation delay of the combinational block.
  - tsetup: Setup time of flip-flop B.

- The **maximum clock frequency (fclock)** is:

  fclock=1tclock



Fast and Slow Standard Cells: Performance vs. Safety

- Faster cells decrease combinational delay but consume more area and power due to wider transistors.
- Slower cells increase combinational delay but may be necessary to prevent hold time violations, where data changes too quickly after a clock edge.
- A balance between fast and slow cells is needed:
  - Fast cells help meet performance (setup constraint).
  - Slow cells can be essential to ensure correct data capture (hold constraint).
- The synthesizer is directed to choose suitable cells using *constraints* provided by the designer.

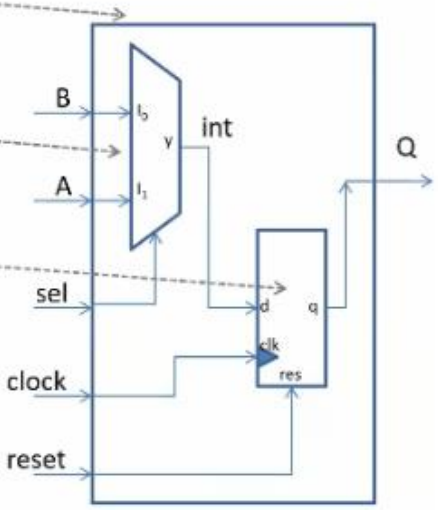
Gate Delay and Physical Characteristics

- Gate delay is primarily determined by the load capacitance and the current drive strength of the cell (related to transistor width).
- Wider transistors drive loads faster (yielding lower delay), but result in larger area and higher power dissipation.
- Narrower transistors save area and power, but introduce greater delay.
- Designer's challenge is to guide the synthesizer toward the optimal tradeoff for a given application, using constraints on timing, area, and power.

Netlist Functional Equivalence

- The synthesized netlist should be functionally equivalent to the original RTL, with identical primary inputs and outputs.
- The same testbench used for simulating the RTL can be reused for simulating the netlist to verify equivalence.
- A correct synthesis flow ensures that functional behavior is preserved from RTL to gate-level representation.

```
module (A,B,sel,clock, reset,Q)
input A,B,sel,clock,reset;
output Q;
wire  int;
assign int = sel ? A :B;
always @(posedge clock or posedge reset)
      begin
        if (reset)
        begin
            Q <= 1'b0;
        end
        else if (clk)
        begin
            Q <= int;
        end
      end
endmodule
```



The circuit on the right is created from RTL using the gates available in the .Lib and given out as Netlist.