HDL

1)
a) Draw the HDL design flow.
b) State the verilog HDL supported levels of abstraction for desiging digital circuit.
c) write HDL code to implement a 4-to-2 line priority encoder.

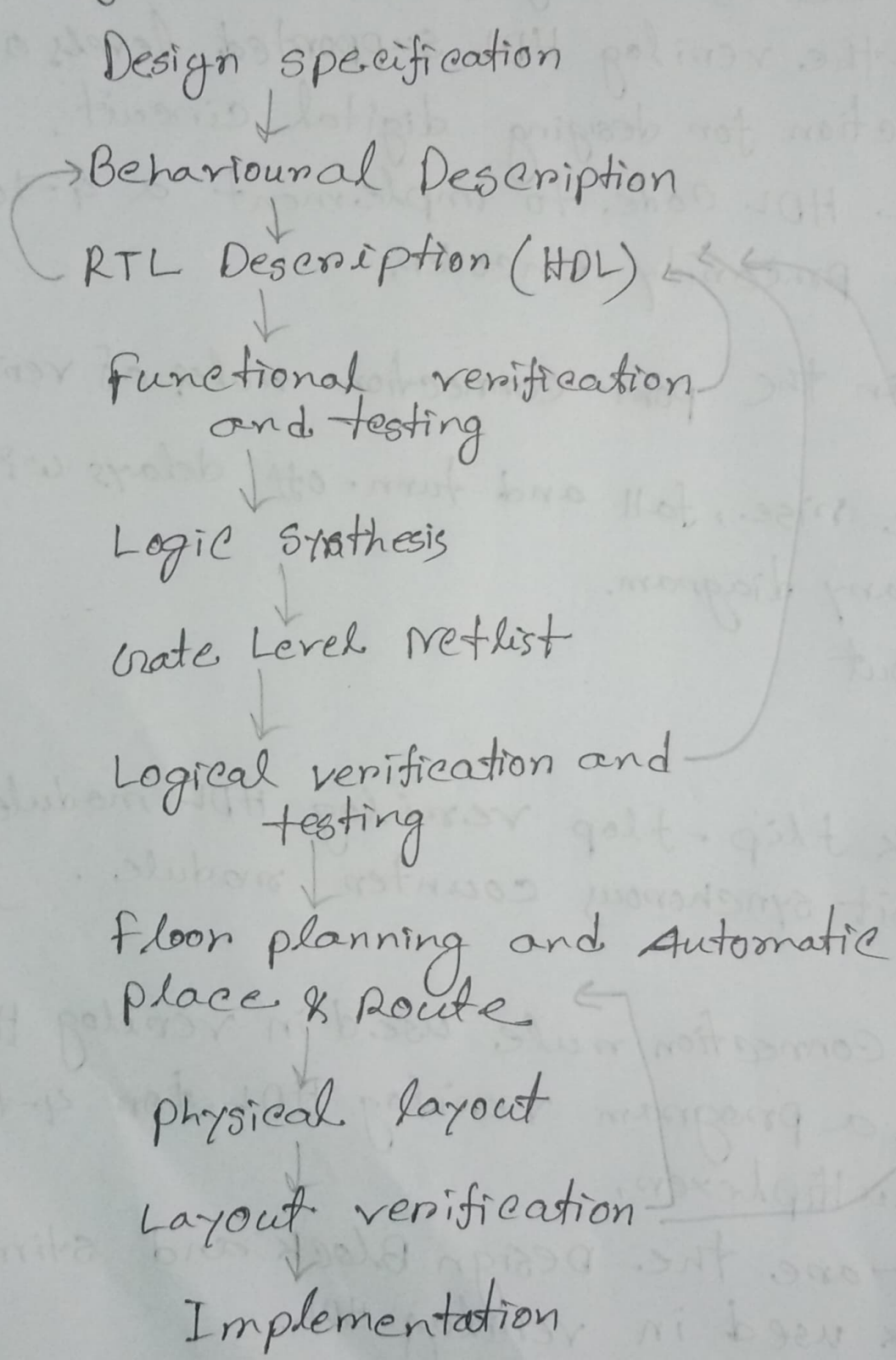
2. a) Explain the port connection rules of verilog HDL.
b) Define rise, fall and turn-off delays with necessary diagram.
c) output

3. a) JK flip-flop verilog HDL module } code
b) 4 bit synchonous counter module. } for

* port connection rule used in verilog HDL.
*) write a program verilog HDL for 4-to-1 multiplexer.
*) what are the design Block and stimulus Block used in verilog HDL.

C# Code and diagram

HDL Design flow:



Port Connection rules:

i) Inputs: Internally must always be of type net, externally the inputs can be connected to a variable of type reg or net.

ii) Output: Internally can be of type net or reg, externally the outputs must be connected to a variable of type net.

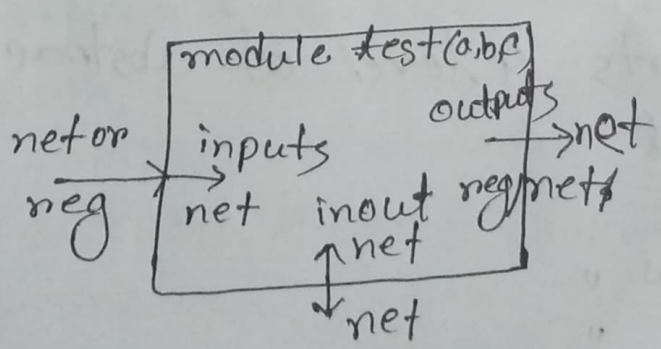
iii) Inouts: internally or externally must always be type net, can only be connected to a variable net type.

iv) Unconnected ports: Unconnected ports are allowed by using a ",".

v) The net data types are used to connected structure.

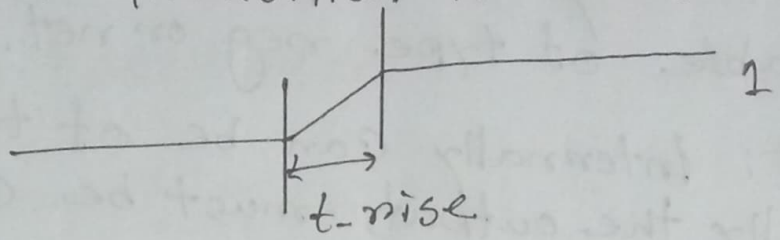
vi) A net data type is required if a signal can be driven a structural connection.

vii) Width matching: It is legal to connect internal and external ports of different sizes.

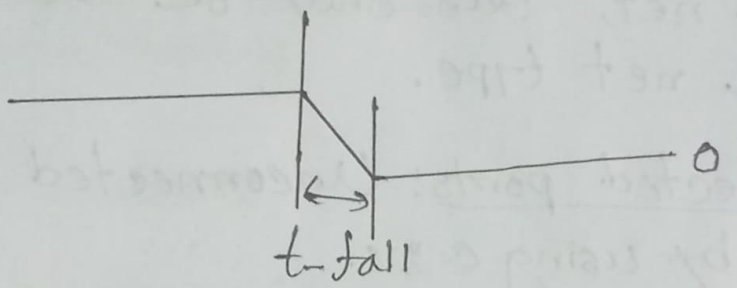


b) Define Rise Delay, Fall and Turn-off delay.

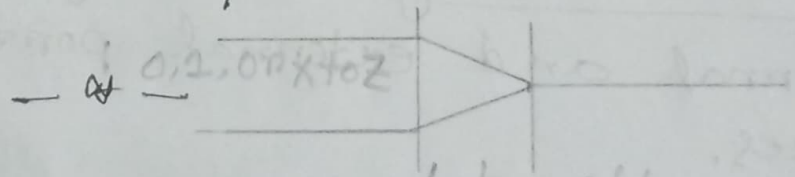
Rise delay: The rise delay is associated with a gate output transition to a 1 from another value.



Fall delay: The fall delay is associated with a gate output transition to a 0 from another value.



Turn-off delay: The turn off delay is associated with a gate output transition to the high impedance value (z) from another value. If the value changes to x, the minimum of the three delays is considered.



2.b)

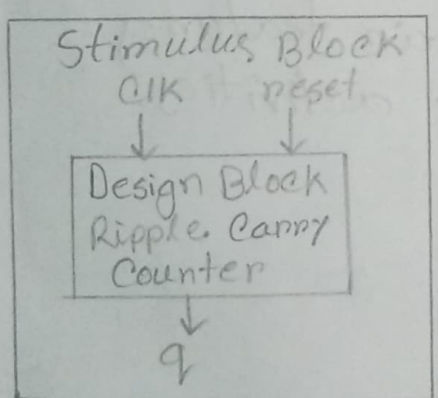
Verilog supports 4 level of abstraction namely

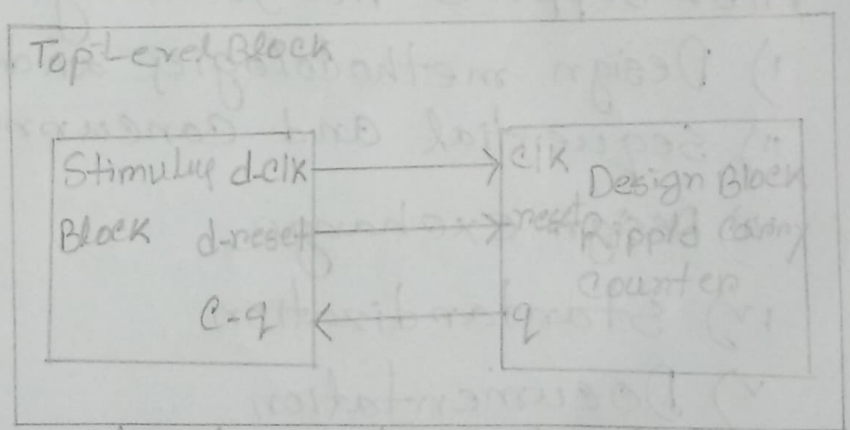
i) Switch level
ii) Gate "
iii) Data flow "
iv) Behavioral "

At the switch level, the implementation is done with the help of switches. At the gate level, we talk about gate interconnections. In data flow level design is done based on equations, which can be said that it is based on data flow, how the data is flowing in the circuit, based on that the equations have written. The behavioral level is based on the behavior of the circuit.

— ✱ —

a)





Stimulus Block instantiates Design block

A design block must be tested once it has been completed. The design block's functionality can be tested by applying stimuli and observing the results. This type of block is known as a stimulus block, and it can be written in verilog as a test bench. To completely test the design block, many test benches might be employed.

— ✱ —

(A) VHDL stands for every High-speed Integrated Circuit hardware description language.

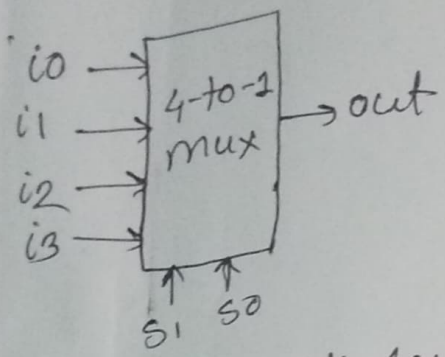
HDL stands for Hardware Description Language. It is a programming language that is used to describe, simulate, and create hardware like digital circuit(ics). HDL is mainly used to discover the faults in the design before implementing it in the hardware.

VHDL supports the following features:

i) Design methodologies and their features
ii) Sequential and Concurrent activities.
iii) Design exchange
iv) Standardization
v) Documentation
vi) Readability
vii) Large-scale design
viii) A wide range of descriptive capability

Verilog: Verilog is also HDL for describing electronic circuits and systems.

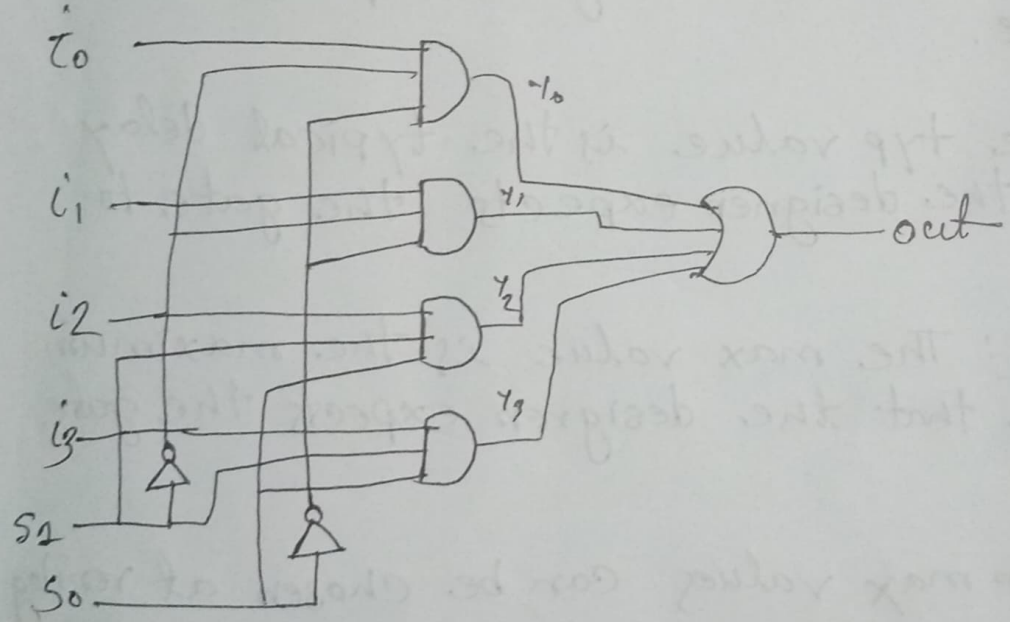
Multiplexer:



S_1	S_0	out
0	0	i_0
0	1	i_1
1	0	i_2
1	1	i_3

4-to-1 multiplexer

Logic diagram for Multiplexer




```
module mux4-to-1 (out, i0, i1, i2, i3, S1, S0)
output out;
input i0, i1, i2, i3;
input S1, S0;

wire S1n, S0n,
wire Y0, Y1, Y2, Y3;

not (S1n, S1);
not (S0n, S0);
```

```
and (Y0, i0, S1n, S0n);
and (Y1, i1, S1n, S0);
and (Y2, i2, S1, S0n);
and (Y3, i3, S1, S0);
or (out, Y0, Y1, Y2, Y3)
end module
```

—*—

④ Min value: The min value is the minimum delay value that the designer expects the gate to have.

Typ val: The typ value is the typical delay value that the designer expects the gate to have.

Max value: The max value is the maximum delay value that the designer expects the gate to have.

Min, typ, or max value can be chosen at verilog run time

—d—