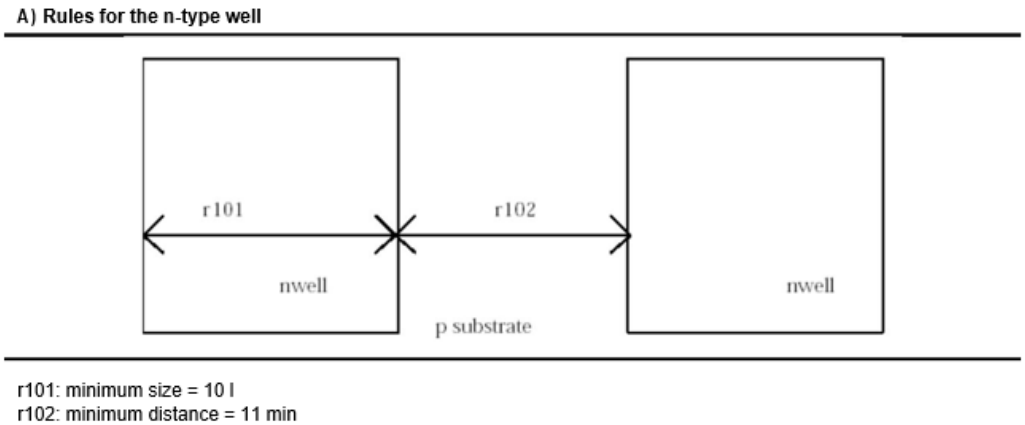
2nd Laboratory Exercise

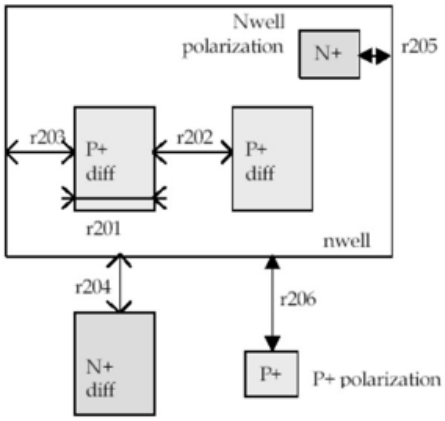
Design rules and simulation

1 Design rules

The design of a VLSI system always begins with the choice of manufacturing technology. Each technology, depending on the manufacturing company and the minimum element dimension (2λ), is characterized by a set of manufacturing constraints called design rules. In practice, drawing rules are minimum distance values between different rectangles inthe drawing. Non-compliance with the design rules results in problematic circuit behavior of the respective element. Inthe MICROWIND program, the manufacturing technology is selected by choosing from the main menu “File” and then *“Select Foundry”.* For each technology there is a file with the extension .rul, in the subdirectory under
the main program installation directory, which contains the corresponding design rules, parasitic capacitances and SPICEmodel parameters for the p-type and n-type transistors that are manufactured with the specific technology (the last two will concern us in the next laboratory exercise). The files are text files and you can observe or change them (very carefully!!) using any relevant tool (text editor). Each drawing rule is encoded in the .rul file with an alphanumeric field starting with the letter 'r', the sign '=' and a numeric value. The alphanumeric field is symbolic and corresponds to a specific dimension of the drawing, which must be at least as large as the numerical value of the rule. Alternatively, all design rules are displayed in a separate screen window, in a table layout (with rows the materials and columns the rules),by selecting from the main menu *“Help”* and then *“Design Rules”.* Along with the design rules,
the MICROWIND program also stores the parasitic capacitances of each material. In detail, the design rules of CMOS035technology (0.4μ technology, ÿ=0.2μ, file “cmos035.rul”) are schematically presented below:

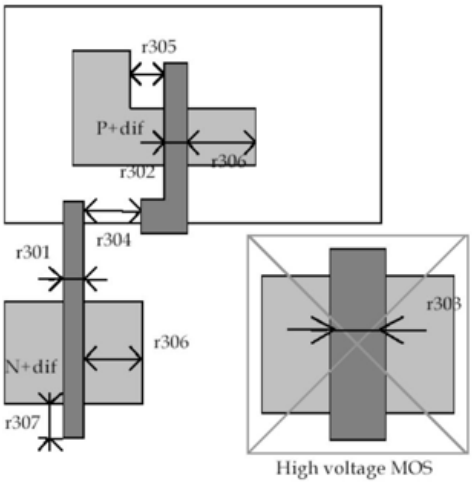


B) **Rules for spillovers**



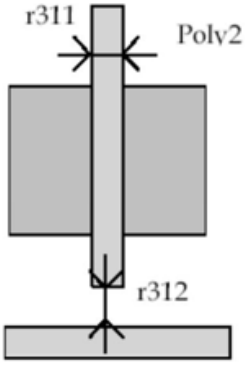
r201: minimum width (p-type and n-type) = 4 ÿ
r202: minimum distance (p-type and n-type) = 4 ÿ
r203: minimum distance between p-type diffuser and n-type well = 6 ÿ r204 :
minimum distance between n-type diffusion and n-type well = 6 ÿ r205:
minimum distance between n-type well and bias contact Vdd (n-type) = 3 ÿ r206: minimum
distance between n-type well and bias contact Vss (p-type) – not defined

C) **Rules for polysilicon**



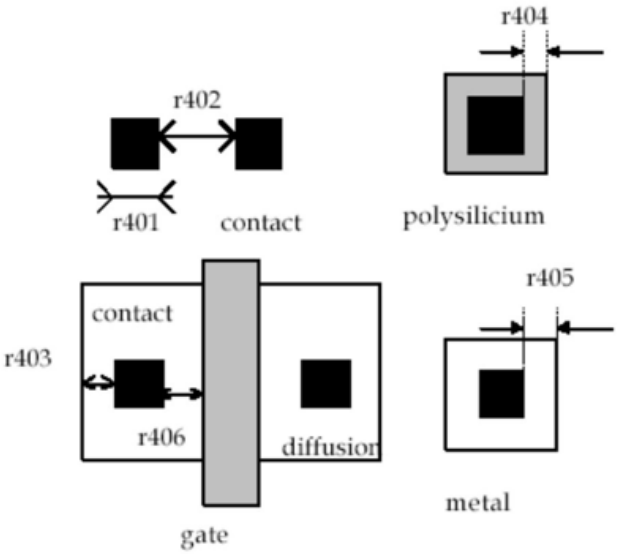
r301: minimum width = 2 ÿ
r302: minimum gate length (L) = 2
ÿ r303: minimum gate length (L) for high voltage transistors – not defined
r304: minimum distance = 3
ÿ r305: minimum distance between polysilicon and uncorrelated diffusion = 1
ÿ r306: minimum distance between polysilicon and diffusion =
4 ÿ r307: minimum gate width (W) outside the diffusion = 2 ÿ

D) **Rules for polysilicon-2**



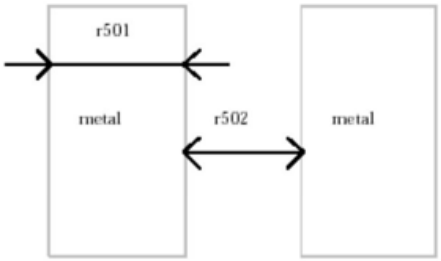
r311: minimum width – not defined
r312: minimum gate length (L) – not defined

E) **Rules for contacts**



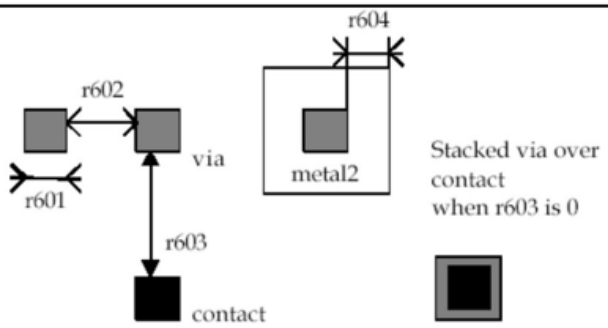
r401: minimum width = 2
ÿ r402: minimum distance =
3 ÿ r403: minimum diffusion width over contact = 2
ÿ r404: minimum width of polysilicon over contact = 2 ÿ
r405: minimum width of metal over contact = 2 ÿ
r406: min distance between contact and gate – not defined

E) Rules for metal



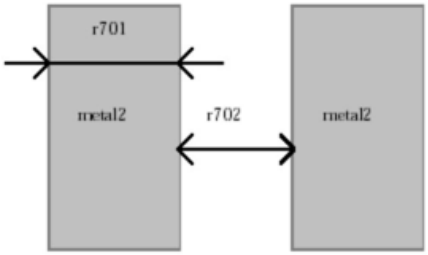
r501: minimum width = 3 ÿ
r502: minimum distance = 3 ÿ

F) Rules for passage



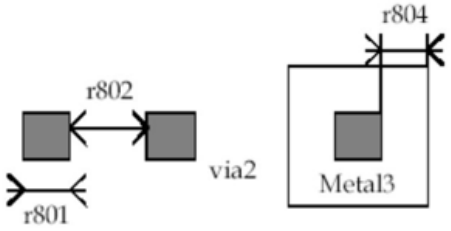
r601: minimum width = 2 ÿ
r602: minimum distance = 3
ÿ r603: minimum distance between pass and contact = 0 ÿ (pass over contact) r604:
minimum width of metal over pass = 2 ÿ

G) Metal Rules-2



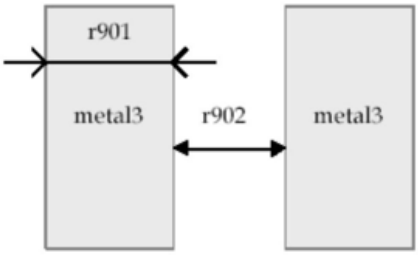
r701: minimum width = 4 ÿ
r702: minimum distance = 4 ÿ

H) Rules for passage-2



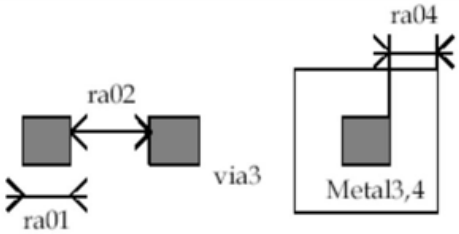
r801: minimum width = 2 ÿ
r802: minimum distance = 3
ÿ r804: minimum width of metal-2 over pass-2 = 2 ÿ

I) Rules for metal-3



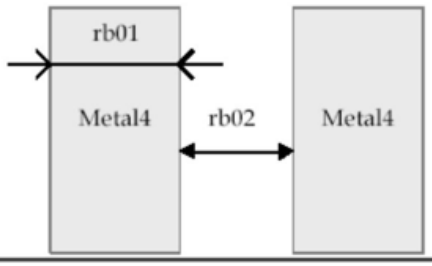
r901: minimum width = 4 ÿ
r902: minimum distance = 4 ÿ

I) Rules for passage-3



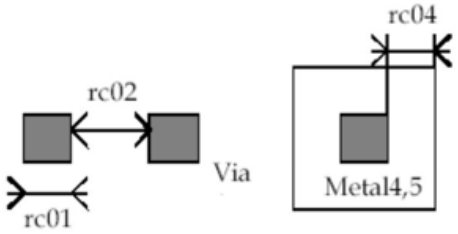
ra01: minimum width = 2 ÿ
ra02: minimum distance = 3
ÿ ra04: minimum width of metal-3 over pass-3 = 2 ÿ

K) Rules for metal-4



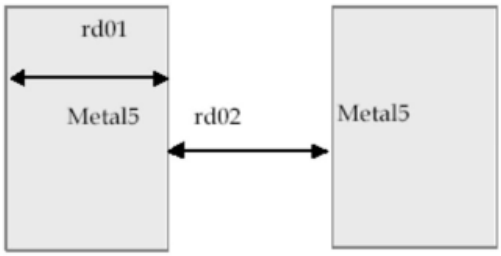
rb01: minimum width = 4 ÿ
rb02: minimum distance = 4 ÿ

IB) Rules for passage-4



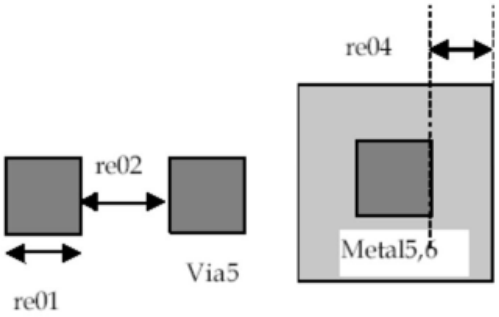
rc01: minimum width = 2 ÿ
rc02: minimum distance = 3
ÿ rc04: minimum width of metal-4 over pass-4 = 2 ÿ

I) Rules for metal-5



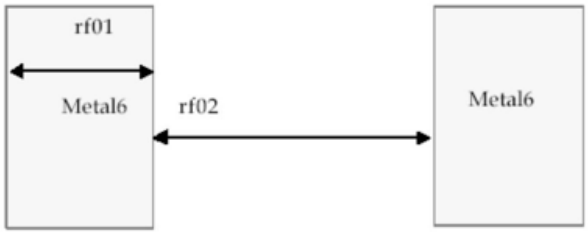
rd01: minimum width = 4 ÿ
rd02: minimum distance = 10 ÿ

N) Rules for passing-5



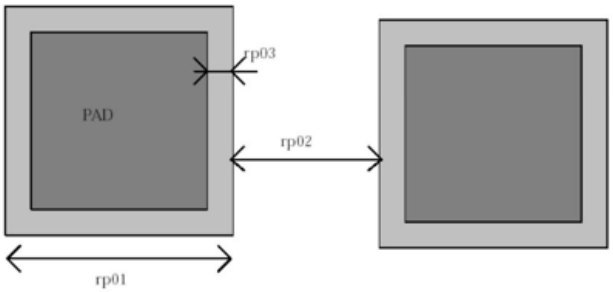
re01: minimum width – not defined
re02: minimum distance – not defined
re04: minimum metal width-5 over pass-5 – not defined

IE) Rules for metal-6



rf01: minimum width – not defined
rf02: minimum distance – not defined

F) Rules for E/E terminals



rp01: minimum width = 550 ÿm
rp02: minimum distance = 550 ÿm
rp03: minimum distance between terminal and connection via = 25 ÿm rp04:
minimum distance between terminal and connection metal = 25 ÿm rp05:
minimum distance between terminal and uncorrelated conductive area = 150 mm

Checking whether the design rules are followed or not is done by the corresponding button onthe toolbar or by selecting from the main menu *“Analysis”* and then *“Design Rule Checker”.* In case a rule is violated, the relevant point of the drawing is shown enlarged alongwith the description and symbolic name of the rule, as shown in figure 2.1. At the same time, between the rectangles that violate the rule, a ruler is shown that certifies the mistake.

In the designs you will make in the lab try to check and follow the design rules so that youobserve correct results when testing and simulating the corresponding circuits.

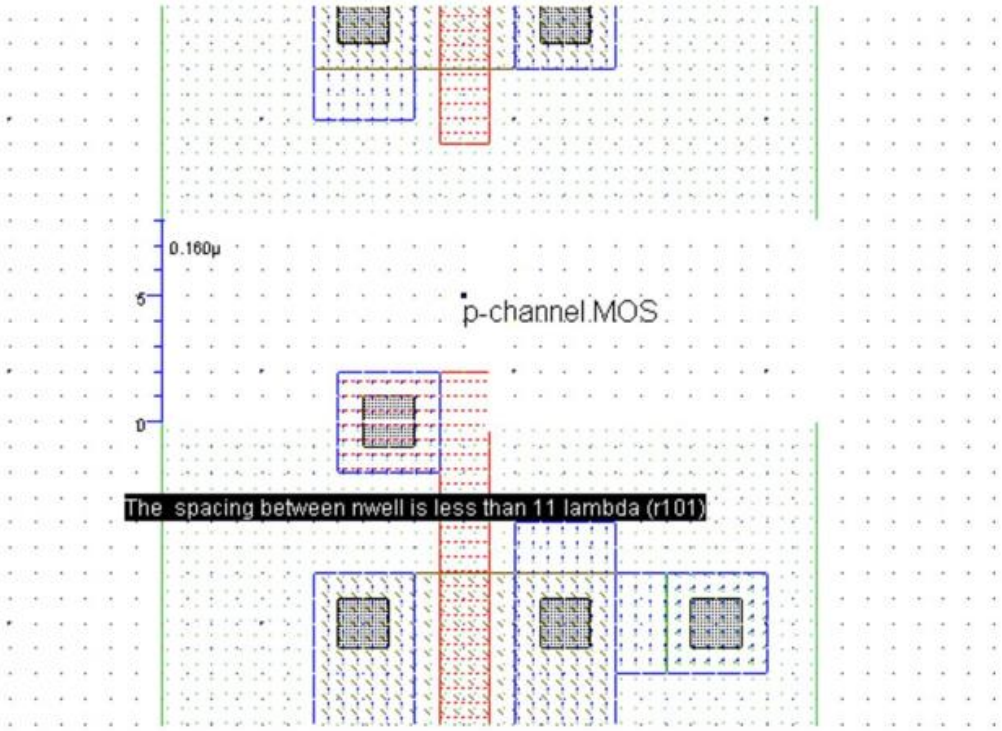


Figure 2.1: Finding errors in design rules

2 Simulation details

Simulation is the process by which the designer checks the circuit he has built for functional errors. The MICROWIND program includes an internal simulation mechanism, which is activated by pressingthe corresponding button, but also the possibility to use an external SPICE engine, choosing from themenu “Convert Into”, “SPICE netlist”.

The internal simulation mechanism is simple and easy to use and is therefore an advantage of the MICROWIND program. All that is required is to apply voltages to the design nodes, both for supply *(Vdd* and *Vss* or *GND,* substrate bias), and for input, and the placement of output terminals. Positioningis done with buttons located in the material palette and allow the graphical determination of all the characteristic quantities of a signal. For example, notice the definition window input square pulse of figure 2.2. The parameters that can be adjusted in it are the level of the negativeand positive front of the pulse, obtained by the manufacturing technology, and the times of negative front *tl,* rise *tr,* positive front *th* and fall *tf.* Note that while the times *tr* and *tf* are desirable to have smallvalues, they are not accepted by the program to be zero. Finally, of the window buttons that all have an obvious function, notice the bottom right button that says *"Visible in simu".* If this button is unchecked, in which case it will read *"Not in simu",* the corresponding waveform will take part in the output waveform calculations but will not be shown in the corresponding graphs.

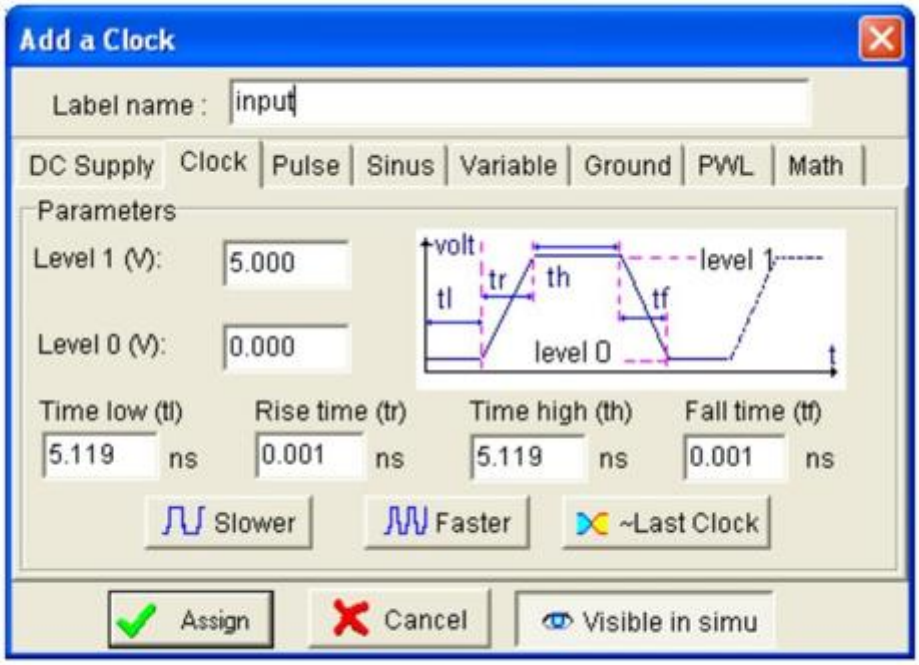


Figure 2.2: Definition of a square input pulse

The internal simulation engine is activated by the corresponding button on the toolbar or by selecting from the *"Simulate" menu, "Run Simulation".* The time diagrams of the change of the input and the output of figure 2.3 (inverter case) are shown on the screen. To the right of the diagram are various settings. On the top right there are two scrolling lists of signals from which the *in* signal has been initially selected in the upper one and the out signal in the lower one . These lists are inserted betweenthe phrase *"Display delay* (selected with check box) *between* … (top list with initial selection *in) and* … (bottom list with initial selection out)". These options determine the time intervals indicated in the bottom waveform. Specifically, they determine the time interval from the moment when the first selectedsignal *(in)* drops to half of its maximum value until the moment when the second selected signal *(out)* rises to half of its maximum value (that is, it is the propagation delay of the circuit) . Also, the user can drag the mouse with the left button pressed in a horizontal or vertical direction, to measure the time

or voltage difference between two points. The *"More"* button continues the simulation for as long as it fits on one screen. The *"Time scale"* parameter (eg 20 *nsec)* changesthe horizontal scale of the charts while the *"Reset"* button refreshes the screen to show the changes from the modified settings. Finally, the *"Step"* parameter (e.g. 2,000 *psec)* determinesthe step of the simulation, i.e. the time distance of the points for which the waveform calculations are done. Below the charts are headings that lead to more simulationpages. On the *"Voltages and currents"* page (figure 2.4), the screen is divided into two sections and the currents of the nodes are shown in the upper time diagram and their voltages in the lower one. On the *"Voltage vs. Voltage"* (Figure 2.5), the transfer characteristic of the circuit isdisplayed, i.e. its output voltage as a function of the input voltage. More details about the simulation can be found in the MICROWIND user manual.

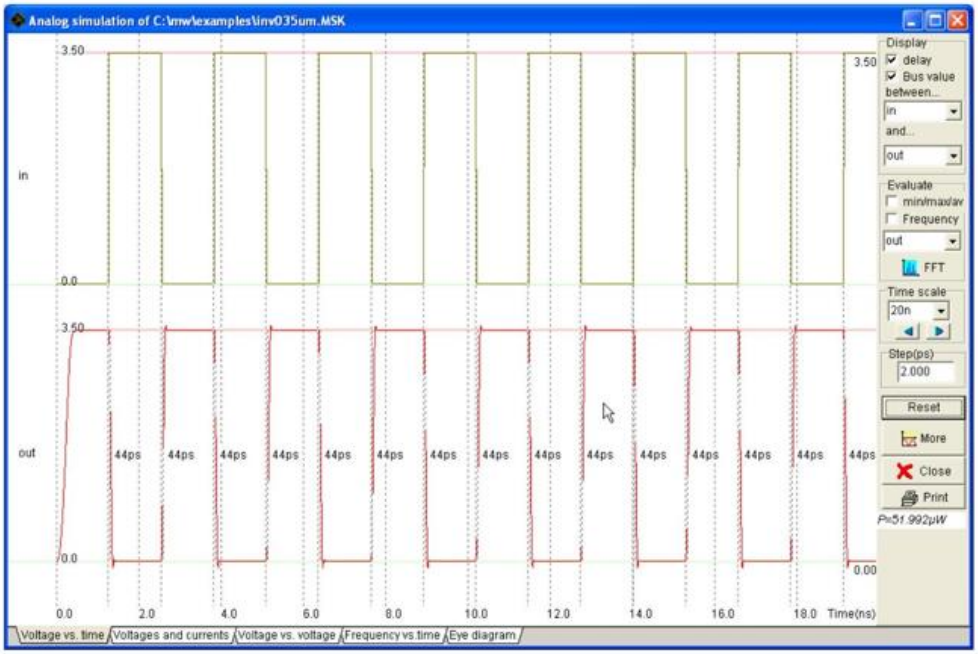


Figure 2.3: Simulation of input voltage to output voltage on a time scale

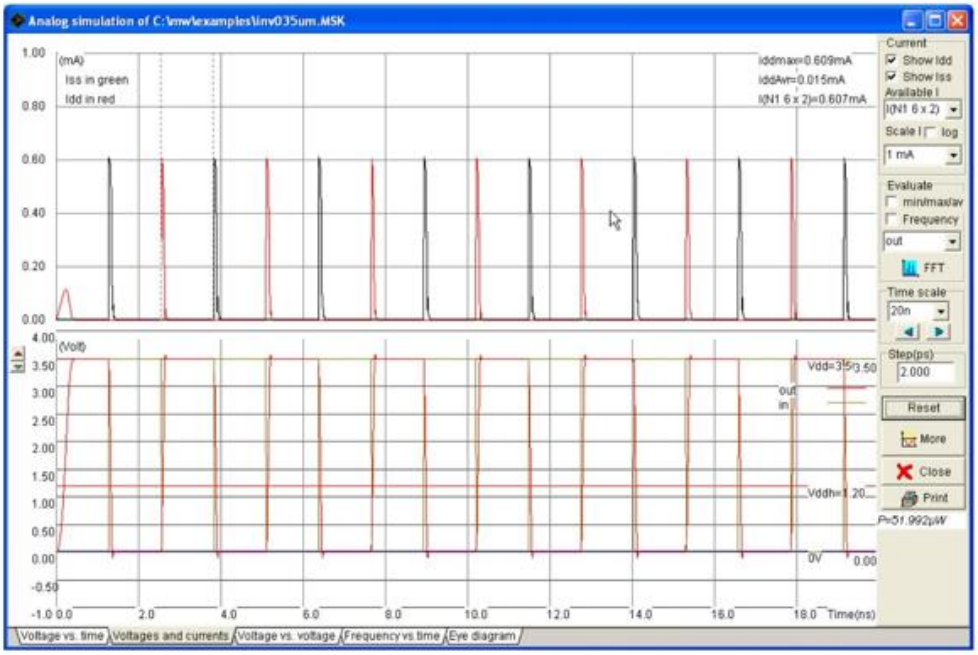


Figure 2.4: Time-scale simulation of output voltage and current

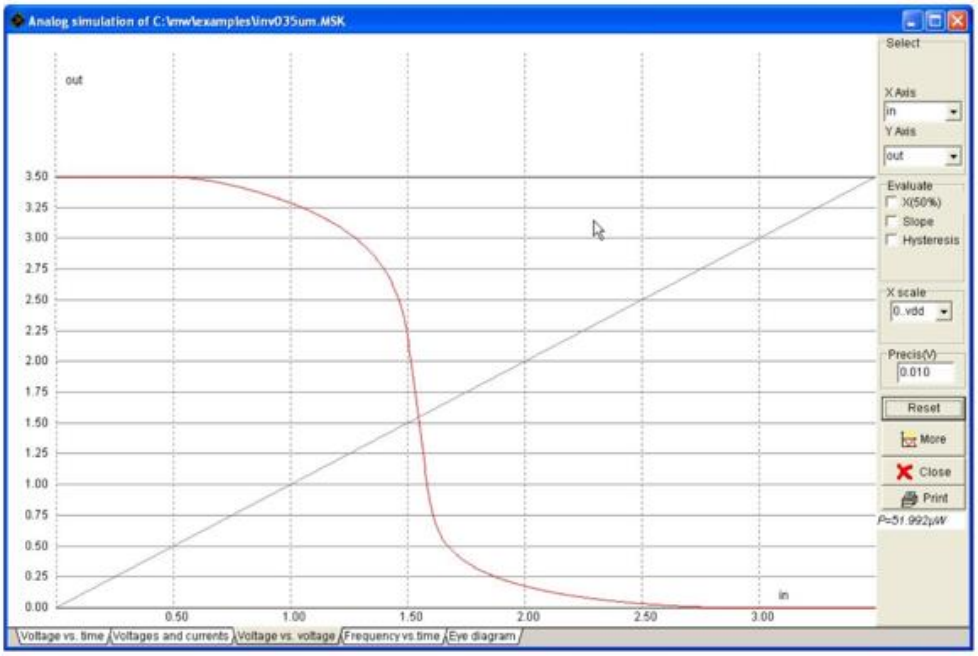


Figure 2.5: Simulation of input voltage to output voltage relationship

3 The requirements of the laboratory exercise

1. Design the layout in CMOS technology of the circuit shown in the figure below, which implements an XOR gate. Explain its operation, check its correct operation and calculate through the simulation the delay it introduces. How can an XNOR gate be similarly implemented (here only the function is explained)?

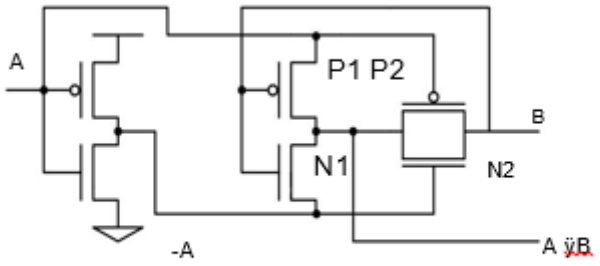


Figure 2.6: Circuit implementing the XOR gate

Also make an alternative implementation in the form of a complex gate based on the relationship *Y=AB = (AB+A'B')' =(AB+XŸ)',* where *X=A'B'=(A+B)'.* Then check for correct operation and compare with the previous form (in terms of circuit and delay). The calculation of the delay should be done through simulation (from input A to output Y) for both schemes. For the output load we assume that the line *Y* drives a capacitance of *10fF.* Select from the capacitor palette and set capacitor with this capacitance.

2. Design an inverting 2-input multiplexer

*(A,*B), with a control signal *C* andexit Y,

in two different ways:

1st way: using transmission gates (and of course with as many inverters as needed).

B' mode: *Y= (AC' + BC)',* implementation as a compound gate.

Check, through the MICROWIND program, the operation and compare the delays in order to find which isthe best design (with the smallest delay). Assume that this multiplexer drives a *10fF load.* The load can be implemented with a capacitor.

Use for all exercises (1 and 2) *CMOS90n* fabrication technology (90nm *technology,* ÿ=45nm, file cmos90n.rul), minimumchannel length on all transistors (L=2ÿ) and width Wn=4ÿ for nmos and Wp= 8l for the pmos transistors. Before simulating, ensure that the design rules are followed.