CSE3142 Integrated Circuit Design

Low-Level Full-Custom Layout Design with IC Station

Practical Experiment 2 **Duration**: two weeks

2.1 About this tutorial

In this study we will familiarize ourselves with fundamentals of graphic editing using Mentor Graphics' IC Station. Details of the IC Station operations are available from:
/sw/mentor/IC_Flow/2004.1/shared/pdfdocs/icstn_user.pdf
In particular, you should study from this manual:

- from Chapter 1: Figure 1-1. IC Station Overview
- from Chapter 2: Figure 2-2. IC Station Full Custom IC Design Flow.
- Chapter 12 for details of Full Custom Editing Operations. Specifically read about graphics operations pp. 12–56 to 12–87.
- Chapter 23: Tutorial. You can find here lots of useful ready to apply information regarding editing. Our practical is partly based on this tutorial.

A specific objective of this tutorial is to use an inverter from the Standard Cell Library as an example and design a **minimum-size** inverter of your own. The new design should pass the **Design Rule Check** (DRC).

2.2 Layers, Shapes, Paths, Contacts

A layout of an integrated circuit consists of **shapes** of different **materials** like diffusion, polysilicon, metal 1 etc., each material existing in its own **layer**. Layers are interconnected by **contacts**. Every material/layer is represented by different colours and colour patterns.

We start with investigation of constituent layers in an existing library cell, and then we study methods of creating and editing shapes and path in selected layers.

2.2.1 Investigation of an Inverter from the Standard Cell Library.

- From a directory **$ICDES/prac2** invoke: **adk_ic &**

  It will open an IC Station window with the **Session** palette on its right. Note that in the upper-left corner of the window there is a message: Process: ami05(-R). It is an indication that the AMI 0.5μm technology is the default and is loaded.

- Open a **library cell** selecting from the **Session** palette **Cell** > **Open** and in the dialog box navigate to **$ADK/technology/ic/ami05/inv01**. This will open a new window with a library cell **inv01** . Maximise the inverter window and from the pull-down menu select **View** > **All** .

- The inverter cell has a hierarchical structure and to see all details we have to "peek" through the hierarchy. To do this select: **Select** > **Select** > **All** and **Context** > **Hierarchy** > **Peek:** and in the prompt bar box enter 99 as the number of levels.

  Use the F2 key to **Unselect All** . Note that new details have appeared in the inverter layout.

- Select from the pull-down menu: **Other** > **Layers** > **Show Layer Palette** and in the dialog box enter 41–51 as the range of selected layers. Alternatively type in in the window: **sho la p 41–51** .

  This opens the layer palette as in Figure 1. Study the layer palette because we will be routinely identifying the circuit layers by their colour codes.

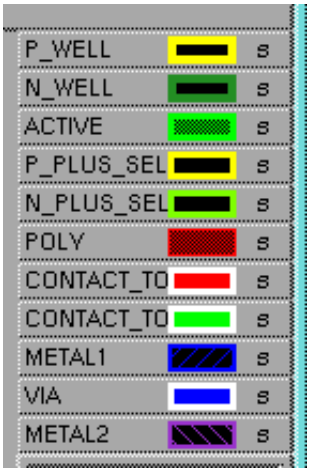

Figure 1: The layer palette.

- We will restrict the visible layers to those 11 layers shown in the layer palette. To do this select **View** > **Visible Layers ...** and in the dialog box type in **41–51** .

- **An Ample Script**

  Up to now all our activities can be recreated by the following Ample script:

```
//  $dofile("p2a.do")  dof p2a.do
  $open_cell("$ADK/technology/ic/ami05/inv01", @read_only);
  $maximize_window();
  $view_all();
  $select_all(void);
  $peek(99);
  $unselect_all(@nofilter);
  $show_layer_palette(@replace, ["41-51"]);
  $set_visible_layers(@replace, ["41-51"]);
  $view_all();
```

- **Layer Palette**

  In the layer palette as in Figure 1 on its right hand side you can identify the **svf** indicators:

  s indicates that objects on the layer are **selectable**,

  v indicates that objects on the layer are **visible**,

  f indicates that objects on the layer have **fill** pattern.

  The **svf** indicators facilitate selection and operation on graphics objects in a desired layer. Test it.

- Practice selection of a single shape. Information about selected shapes is printed in the information bar at the bottom of the IC window. For instance, start with the F2 key to **Uselect All** and select the upper part of the N_Well. The info bar should read:

  Shape ... layer 42 (N_WELL) ... area is 495 ... perimeter is 96

  Note that in the upper bar of the window there is information about:

  – Cursor position in the 0.5λ units.
  – The current layer from the layer palette.
  – Number of selected objects.

- Examine the inverter layout carefully trying to identify as many objects (shapes, paths contacts) as possible.

- Finally reduce the inverter window to occupy approximately the left quarter of the IC Station window making room for your own test cell.

2.2.2 Creating a new cell

- Now you can create your own cell in which you will create and edit layout objects. Use the

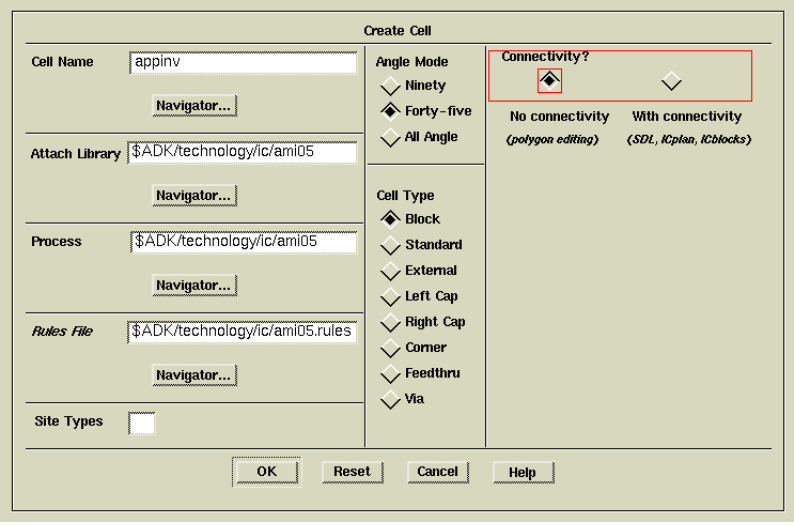

Figure 2: The Create Cell dialog box

Cell > **Create** option on the **IC Session** palette menu or use the **File** > **Cell** > **Create** menu to do this.

For the cell name enter **Up2** , where **U** represents your initials.

You can specify the Library, Process, and Rule files as in the dialog menu in Figure 2. The AMI 0.5μm technology is already the default and loaded, but you can re-load it to practice.

You should have a window open titled: IC 0: Up2> Up2

- Arrange the **Up2** windows side by side with the inverter window to occupy the remaining three quarters of the IC Station window.
- From the **IC Palettes** select the **Easy Edit** palette. You can activate the scroll bar (using the right mouse button) to see all items of the palette.

2.2.3 Creating and editing shapes

A shape in general can be polygon, but for simplicity we restrict our attention to rectangles.

- Let us create the metal 1 rectangle of the size 16×4. From the layer palette select **metal 1** and from the edit palette select **Shape** and draw the rectangle at starting at [30, 0].

  This is equivalent to two functions:

```
$set_ic_layer("METAL1");
$add_shape([[30, 0], [46, 4]], "METAL1", @internal, @nokeep);
```

  There is a number of 'Hotkeys' available when the shape tool is active. They are displayed in the info bar at the bottom of the screen. You might like to investigate how they work.

- The shape can be edit in many complex ways described in IC Station documentation. I find particularly useful a **relative stretch** function. The important thing to remember is that we **stretch vertices** enclosed by a rectangle.

  For example, assuming that two right vertices of the metal 1 shape are at [46, 0] and [46, 4], in order to stretch the rectangle horizontally by 5 units we:

  - Select the shape.
  - In the edit menu we click on **Rel Stretch**.
  - In the command bar type in 5 and 0 as the relevant offset values,
  - Select the **Polygon** option,
  - Enclose two vertices that you would like to move with a rectangle which performs the stretch operation

  This is equivalent to the following function:

```
$stretch_relative(5, 0, [[45,-1], [47,5]]);
```

- Practice more stretch and shrink (negative offset) operations.

2.2.4 Creating and editing paths

A path is a polyline in a selected layer of specified width.

- From the layer palette select **poly** and from the edit palette select **path** . In the ADD PA prompt bar select Options... and in the dialog box confirm that **POLY of width 2** is selected. Check the Keep Option Settings button.
- Start drawing with single clicks at bends at: `[30,10], [40,10], [40,15]`
  and the double click at the end at `[60,15]`. You should see a poly path of width 2 doubly bent up and right. This is equivalent to the following function:

```
$add_path([[30,10], [40,10], [40,15], [60,15]], "POLY",
              @internal, 2, @center, @normal, @keep, @nopad);
```

- The basic editing operation with paths is a modification their centerlines. Select the poly path and from the **pop-up menu** select **Edit** > **Modify Ctrline:** Just perform a single click at the centerline you are going to move and a double click at the new position. You can shrink, stretch, and bend at the end or in the middle of the path. Practice this operation.
- There are **two macros** provided by **adk_ic** (see the ADK documentation) to facilitate creation of paths:

  p Place a poly path of width 2 (minimum width); Prompts you to place a poly path.

  m Place a metal1 path of width 3 (minimum width); Prompts you to place a metal1 path.

At this stage your Up2 window should be similar to the bottom part of Figure 3.

2.2.5 Contacts and vias

Contacts are structures connecting electrically two layers, normally, metal 1 layer with either polysilicon, or diffusion, or metal 2. The metal1–metal2 contact is called a via.

A contact consists of three masks: two for the materials that are being joined, and the third one for the cut (through the insulating silicon dioxide) which will be filled in with metal.

In order to simplify creation of contacts there are following AMPLE macros written for the purpose:

pc Place a poly contact; places a poly contact centered at the current cursor position.

pp Place a port contact (via); creates a port and places it where you click.

nwc Place an n-well contact; creates and places an n-well contact on the standard cell power rails.

pwc Place a p-well contact; creates and places p-well contact on the standard cell power rails.

Diffusion contacts are created automatically with the transistor generation macro and will be examined in the next section.

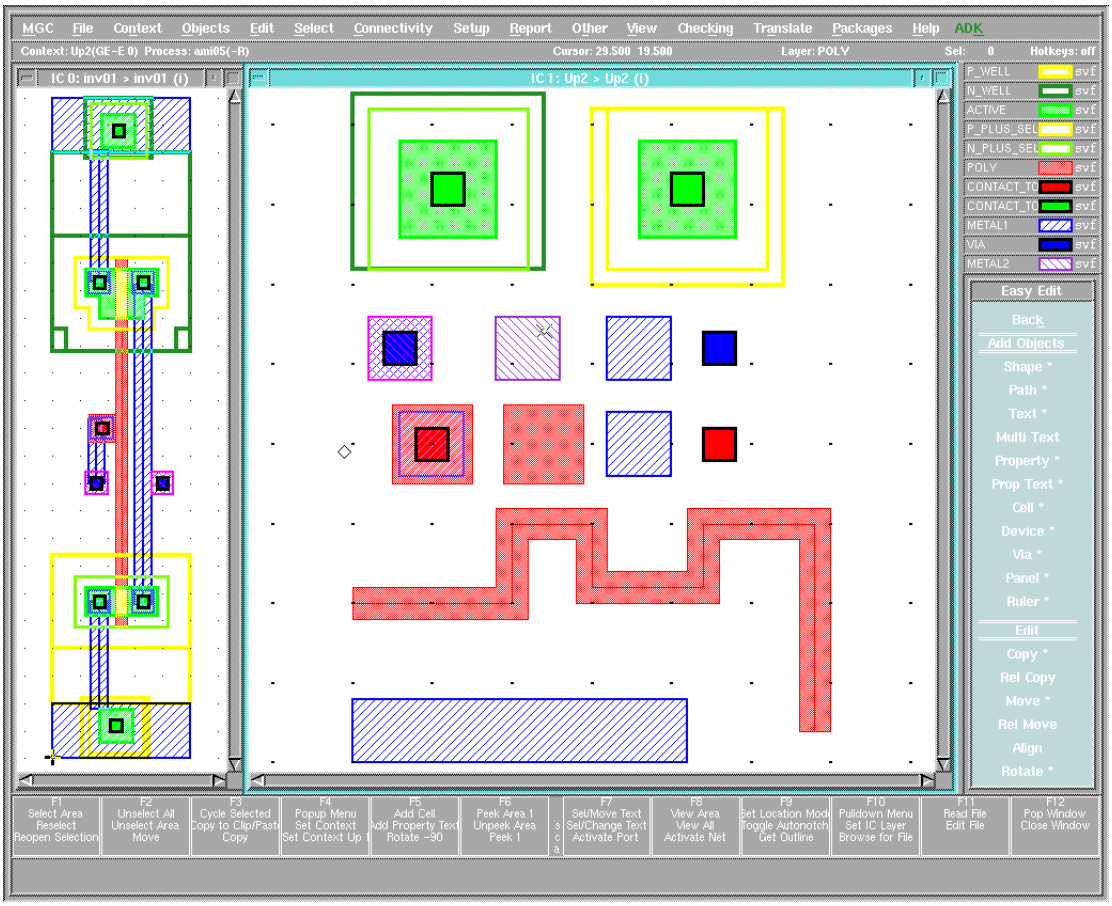

Figure 3: A snapshot of the Up2 window.

- Create two **polysilicon contacts** with the **pc** macro, one at [35, 20] and the other at [42, 20]. Select and move horizontally apart three constituents of the second contact. Find out their sizes.
- Examine the pc macro which can be found in: `$ADK/userware/ic/user_ic.ample`
- Create a **metal1-metal2 contact** (via) with the **pp** macro at [33, 26]. This macro in addition to creating geometric shapes, also associated the shapes with an electrical port. Type in the `peek 99` command, or execute **Context** > **Hierarchy** > **Peek** to see details of the contact.

  Note that the pp macro instantiates only the respective library component, namely:

  ```
  $add_cell("$ADK/technology/ic/ami05_via/via", [33, 26]);
  ```

- Create your own metal1–metal2 via using the **Shape** function at [41, 26] and then move its components horizontally apart.
- The **n-well/p-well contacts** connect an n-well/p-well (or substrates) to VDD or GND, respectively. These are in principle metal1 to diffusion contacts, but in the area of the other diffusion. We will examine it latter on in the context of an inverter.

Use the **nwc and pwc** macros to create the n-well/p-well contacts at [38,36] and [54, 36], respectively. Peek through their hierarchy and find out dimensions of all its components.

Note that the nwc, pwc macros just instantiate the respective library components, namely:

```
$add_cell("$ADK/technology/ic/ami05_via/nwell_contact", [36, 36]);
$add_cell("$ADK/technology/ic/ami05_via/pwell_contact", [51, 36]);
```

At this stage your Up2 window should be similar to that in Figure 3.

2.2.6 MOS transistors

MOS transistors and associated diffusion contacts can be built using the 'Shape' function, but it is a tedious task. Therefore there is a macro to speed up transistor generation.

- From the **Easy Edit** palette select **Device** and in the dialog box select **Specify Name** . Type in **mos** as the device name and in the **MOS Parameters** dialog box specify:

Width	5
Length	2
Sequence	cggc (contact-gate-gate-contact)

and Mos Transistor Types nmos

This will generates a serial connection of two nMOS transistors with diffusion contacts on both sides. Peek through the hierarchy and find out dimensions of all its components.

- Practice generation of other combinations of MOS transistors.

2.2.7 Print or capture the screen

At this stage arrange your Up2 window to be similar to that as in Figure 4.

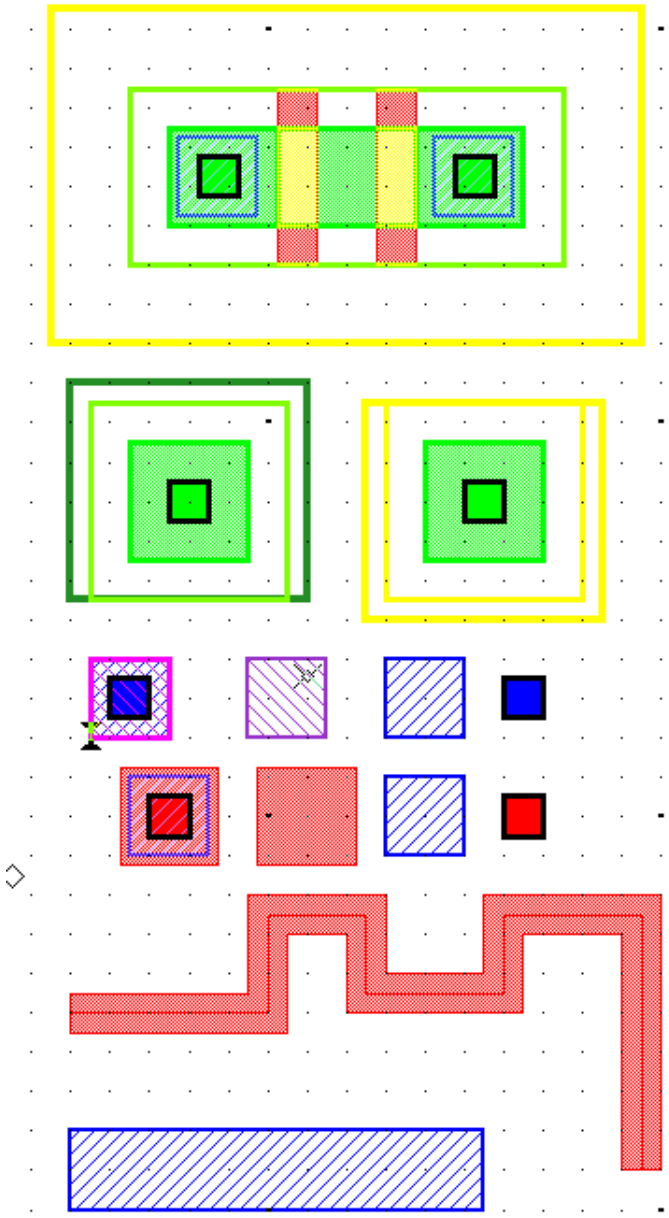

Figure 4: Current state of the Up2 window.

Print in a postscript format or capture the screen in a pixel form. If you capture the screen inverse the black background to white before printing.

Finally clean up the Up2 window by making it active and selecting **Select** > **Select** > **All** and deleting everything.
Note also that after each **Save Cell** operation you have to reserve the cell for the edit operation executing **File** > **Cell** > **Reserve** > **Current Context** .

2.3 Creating a minimum-size inverter

In this part of our practical activities we will create an inverter similar to the one from the standard cell library that should still be present in the left hand window.
We will aim at the **smallest possible** inverter which still satisfies the **Design Rules**.
A possible version of a small size inverter is shown in Figure 5

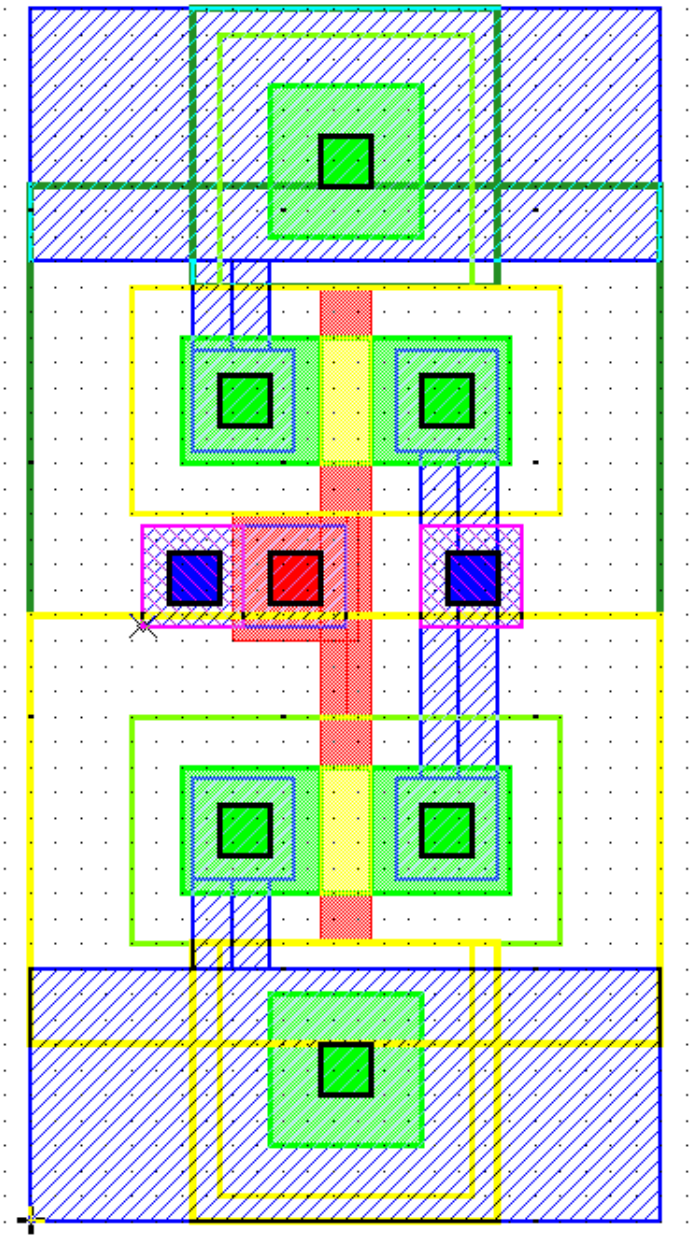

Figure 5: An example of a small size inverter satisfying the design rules

One of the ways to proceed with creation of the inverter is as follows:

- Generate an nMOS of size $W = 5; L = 2$ with contacts (cgc) at, say [4, 15]. Peek through to see details. Note that the p-well is $24 \times 17\lambda$
- Create a metal 1 shape of size $24 \times 10\lambda$ starting at [0,0]. It will be a GND rail.

- Add the p-well contact so that its bottom edge aligns with the bottom edge of the GND rail.
- Move the transistor assembly down checking the design rule violation.

2.3.1 Design Rule checking

Initially it is necessary to run DRC checking from the **ADK Edit** palette, it is under **Verification** > **Drc** > **Check** . Once this has been run Design Rule checking can be invoked and the ”results” examined by with following set of functions and equivalent commands:

```
$check_drc();          che drc
$set_drc_first();      set drc f
$set_drc_next();       set drc n
$delete_drc_all();     del drc all
```

After the **che drc** command it is more then likely that you are going to get a message on the bottom of the IC Station window similar to the following:

DRC completed. Total RuleChecks: 80; **Total Results: 11**; ...

It means that there are 11 design rule violations in your circuit. To see them one-by-one execute first the **set drc f** command, and then the **set drc n** command, until you see all the design rule violations.
All results are clearly shown on the layout and described in the information panel of the IC Station window.
Fix all errors and execute the command **del drc all**

- After the n section of the inverter is DRC verified, complete in a similar way the p section of the inverter.
- Use the **path** command or macros **p** and **m** to complete interconnection between the gates and to VDD, GND, and the output terminal.
- Use macros **pc** and **pp** to add required input/output terminals of the cell that should be available on the metal 2 layer.
- Perform the Design rule checking on the complete inverter until in the info bar, or in the shell window you got the message:

  Total Results: 0

- Calculate the total area of the inverter.
- At this stage you can save your inverter, prints its layout and exit the IC Station.

2.4 Creating a minimum-size layout of the 2-input NOR gate

- Create a layout of a new cell called **Unor2** which will be a two-input NOR gate. A schematic and a stick-diagram of such a gate is given in Figure 6.
- The layout should pass the Design Rule Check.
- Calculate the total area of the cell.

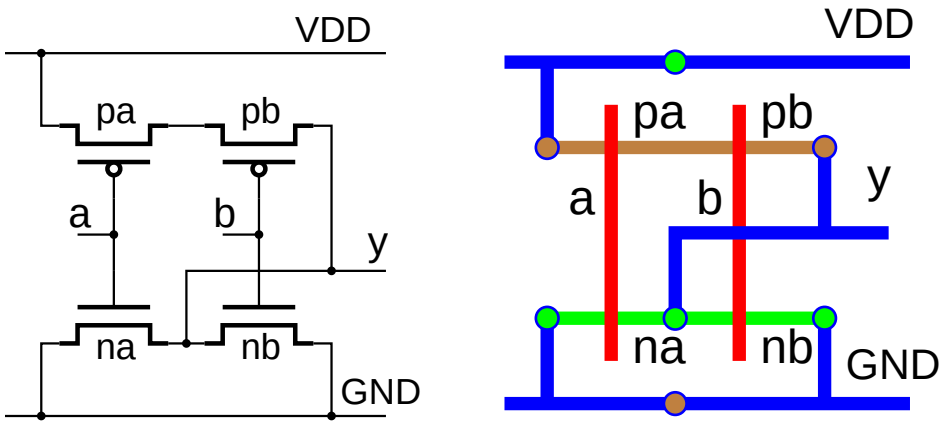

Figure 6: A schematic and a stick-diagram of a two-input NOR gate.

2.5 Demonstration and Report

- Demonstrate your circuits to your tutor.
- Write a brief report in which you include a short description of your activities and the results obtained. Include schematics, layouts, stick diagrams and relevant explanations regarding the structure of the library and your own cells.

  Edit postscript files as described in prac1 if required.