eProduct Designer

A Simple Design and Simulation Tutorial

Written by Bahram Dahi

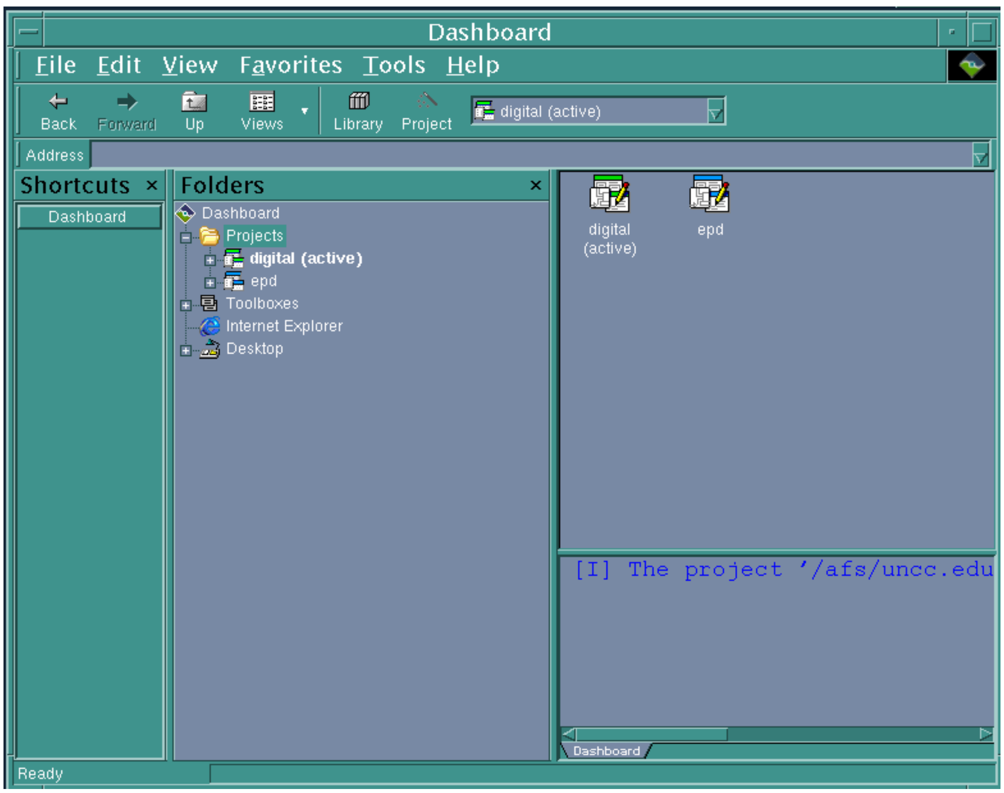
Fall 2003

Updated Spring 2007

Dashboard

Project management tool

1. In the main window, click on the *File* menu and select *New*→*Project*.
2. In the Create Project dialog box, set a directory for your class project. Click OK. Use the name "ITCS3183" (or the current class you are taking) in the root directory of your **H drive. If you save the project folder on the C drive, it will be erased after you log out.** As well, do not use punctuation, special characters or whitespaces in any naming for this program. You will only need one project file. It will contain all of your schematics that you are going to create in the semester.
3. In the main window, select the project you have just created (ITCS3183 in our example). Right click → *Set as the Active Project*. Now the project should turn green and "(active)" should appear to the right of the name.

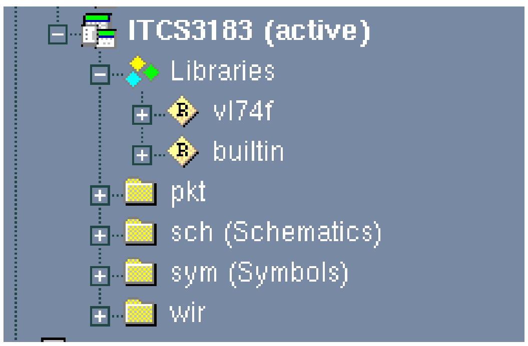


4. Expand the project by clicking on the small "+" sign on the left side of the project title. You can see several default directories in the project. Right click on the Libraries and select *Add Library*.

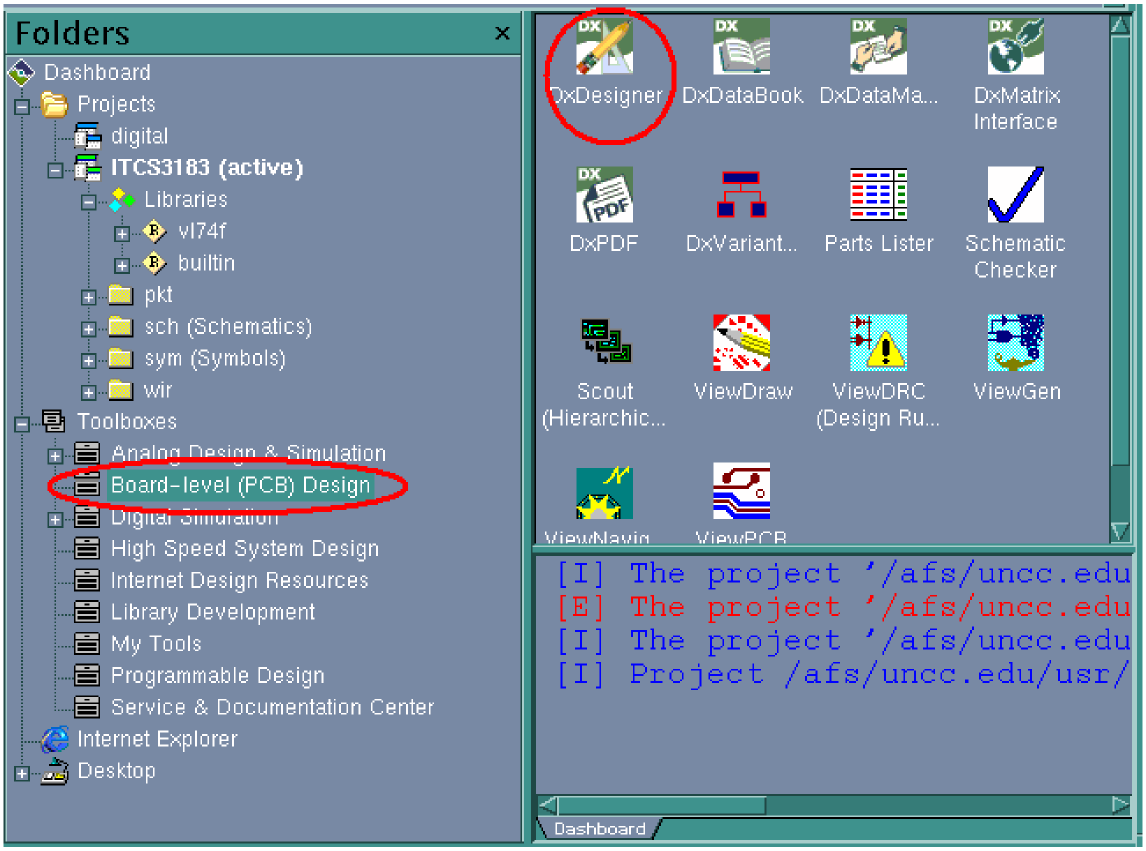
5. In the Properties dialog box, use the following path to get the installed libraries on the system: *C:\ProgramFiles\MentorGraphics\Library*. Use the *Browse* button (…) to browse through the directories to find the libraries needed for the project. The two libraries used most are built-in (*/builtin*) and Motorola 74f (*/74f*).

   Note: Both libraries are needed. The builtin library is used when creating circuitry in an ideal world with no delays. This can be used to make sure the logic of combinational circuits is correct. Motorola 74f add delays to the gates. This is used to simulate more realistic scenarios (allowing one to check for timing issues) and is typically what is used in a final product. As well, if one is using sequential logic one must use only Motorola 74f since the design of such logic rely on delays in the design.

   Important: Double-click on the folders to go inside a library folder and then click on *Select*. If you do not see the yellow "R" sign after selecting the library, or the "R" sign is crossed out, do it again.

6. Your project is ready now. In the Folders view, expand Toolboxes and choose Board-level (PCB) Design.
7. Double-click on DxDesigner to go to the schematic design environment.



DxDesigner

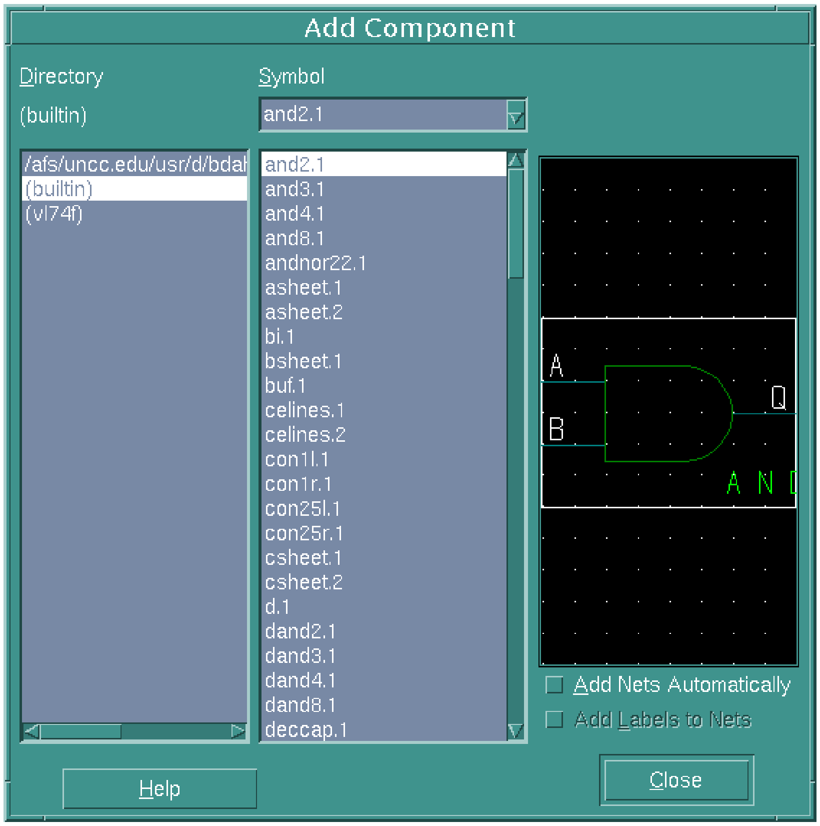
Schematic design tool

Here is a step-by-step method to build a custom XOR component:

8. In the main window, go to *File*→*New…*
9. Select *Schematic* and choose a name for it. In our example the name of the project is "c-*xor21*". Click OK. A blank schematic will come up.

   Tips: There will be two toolbars that appear on the schematic. The toolbar on the left are mostly commands from the *Add* menu and the toolbar on the right corresponds to the *View* menu. There are also shortcuts for each of the commands that we are going to talk about. You can use either of the methods.

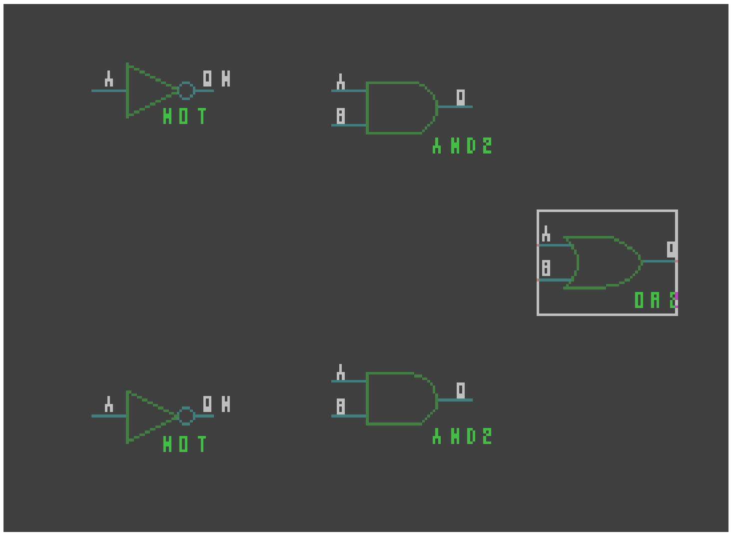
10. Select the C*omponent* button from the left toolbar (or *Add*→ *Component* or press the lowercase "*i*").
11. In the Add Component dialog box, choose the library you are going to use. In our example we use the *builtin* library.
12. You will get the names of the components in the middle section.



Note: The number at the end of the names means the number of inputs. For example *and2.1* means the component is an AND component with 2 inputs. ".1" is simply the extension for the file. Do not use this convention in your own custom components as it uses a "." in the name.

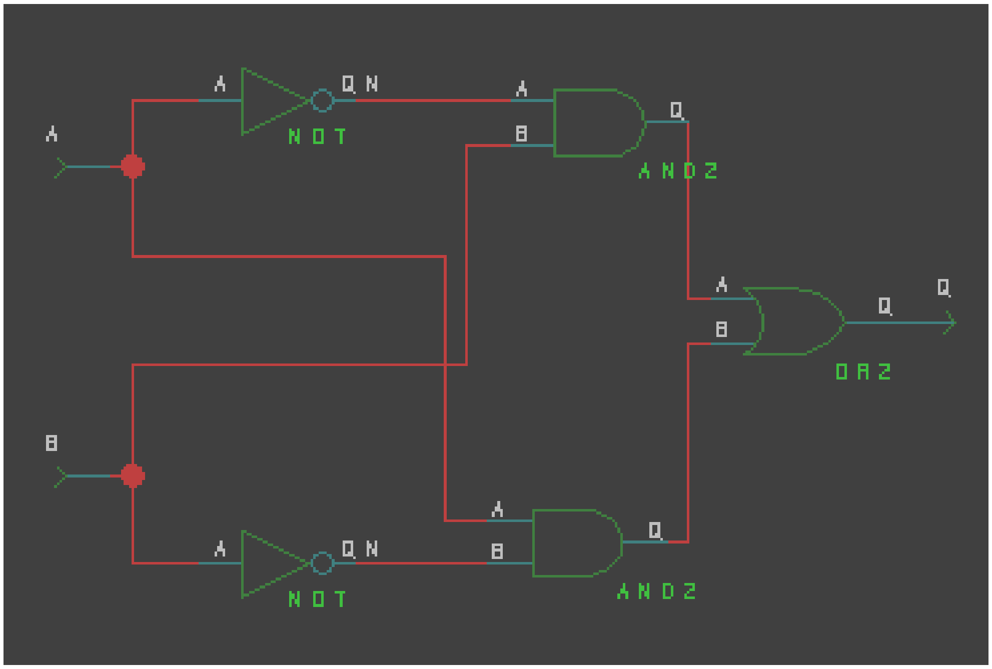
Tips: You might have noticed that our component is called "*c-xor21*". The prefix "c" shows that it's a custom built component, "2" means that this component has 2 inputs and 1 indicates it had 1 output. If you would like, follow the same naming convention for your custom components. It may make them easier to find them in the future. **Do not however use the convention sometimes seen in the library components that involved using a "." in the name. Never use punctuation, special characters or whitespaces in the names of anything involving this program.**

13. Choose "*and2.1*" and drag it into your schematic page (hold the left mouse button while selecting the component, move the component to the schematic and release the button).
14. Also, drag one "*or2.1*" and one "*not.1*" using the procedure above. Now you have at least one copy of all the components you will need to make an XOR.
15. Make a copy of your "*and2.1*" and "*not.1*" components by dragging them to a blank space of the schematic while holding the *Ctrl* key. Now you have two "*not.1*", two "*and2.1*" and one "*or2.1*". Arrange them in the following order:



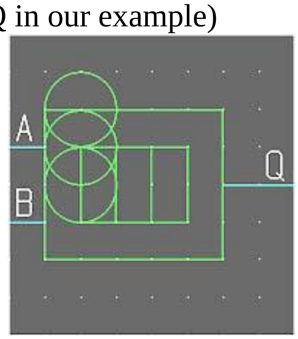
16. Now is the time to add the input and output components of the schematic. Our *c-xor21* component has two inputs A and B and one output Q. Follow the steps above to add two *in.1* and one *out.1* components to the schematic.
17. Now label your inputs and outputs by double-clicking on each input and output component. In our *c-xor21* component, we name them A, B and Q.

18. After you labeled your input and output components, you need to connect the components you have just dragged to your schematic using *nets*. Point the mouse pointer to pin A of the top AND component and press “n”. Move the mouse pointer to the Q pin of the top NOT component and click. You can see a red line drawn connecting the two pins.
19. Connect the rest of the pins using the map below.



Tips: Try to put the components at a reasonable distance from each other. Two pins that are very close together are hard to distinguish and connect separately. They also have a tendency to get connected automatically. On the other hand, components that are too far apart make the schematic look bad.

20. The schematic is now ready. Use *File* → *Save + Check* in the menus to save your schematic and check for any errors. If there are any errors, a window will pop up to notify you.
21. Next, create a symbol for the custom component. On the left panel, right-click on the *c-xor21* and select *Open Symbol*
22. Make a symbol using lines, arcs, rectangles and texts. Be creative!
23. To add *pins* to your component, click on the *pin* icon on the design toolbar. Pins are your components inputs and outputs. The number of input and output pins in your symbol should be the same as the number of input and output components in your schematic, respectively.
24. As a standard, we try to put all the inputs on the left side of the symbol and all the outputs on the right side. Add two inputs and one output for the symbol. Label them by double-clicking on each pin. Make sure your labels match with the schematic labels (A, B, and Q in our example)



25. Save the symbol. You will see a page showing the information about the symbol you have just made. Simply close it.

    Note: Schematics, Symbols and the information pages like above are all of type "window". Don't get confused when you see your schematic is gone and a white page is there instead! You can easily close unnecessary windows, or switch between all the open windows using the *Window* menu.

26. Now create the digital netlist files for the schematic. Go to *Tools→Create Digital Netlist (common)* and click on the *Create Netlist* button. It automatically calls another program in *eProduct Designer* to create one set of files necessary for the simulation.
27. Look to see if your schematic has any errors or warnings. You are done with the design part!

VWaves

Simulation Tool

You can find this program in the *Digital Simulation* toolbox in the *Dashboard*. However, for simple projects you don't need to call the program separately. Here is a simpler way to call *VWaves* and simulate your schematic:

28. When your schematic is open in the DxDesigner program, go to *Tools→ Create Digital Netlist (common)*. This time, notice that there is a check box called *Simulate After Netlisting*. It calls several programs after the Netlisting program to check and make binary files. In the end, it calls VWaves to simulate your schematic based on a *Command File* that you provide in the textbox below the checkbox.
29. We need to create a *Command File* before we can continue with the simulation. A command file is like a batch file that tells the simulator what the value of each input pin is at each clock cycle. To begin, open Notepad to a blank text file.

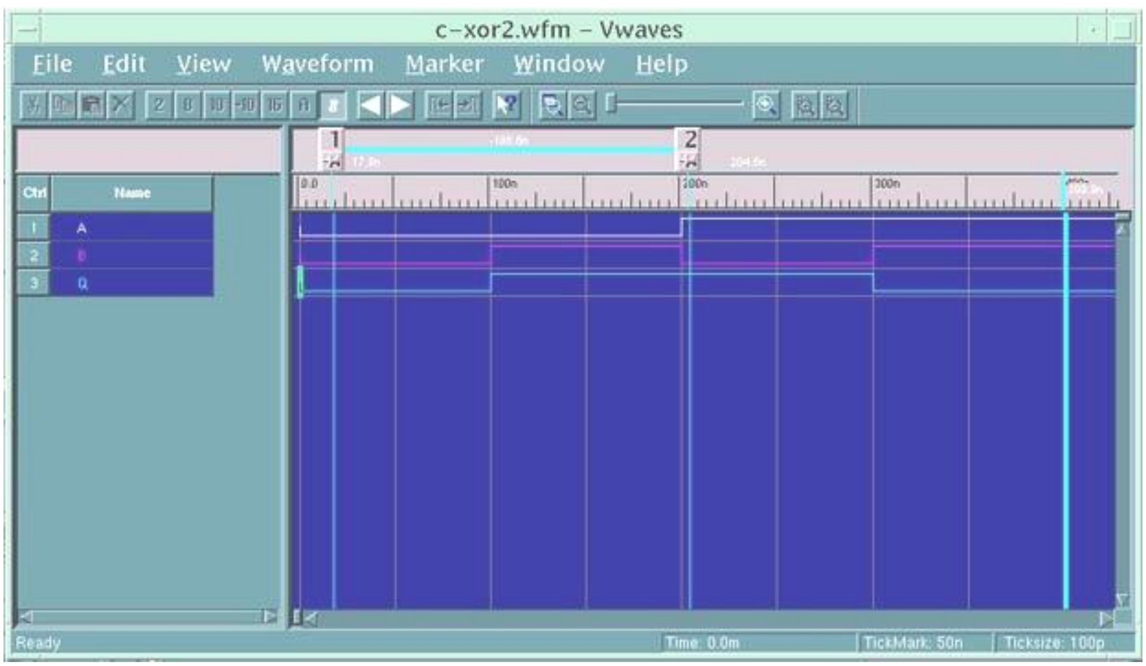
30. Write the following lines in the new text file:

```
restart
wave c-xor21.wfm A B Q
pattern A 0 0 1 1
pattern B 0 1 0 1
run
```

The first line simply tells the simulator to start the simulation from the beginning. The second line identifies the compiled schematic and all the inputs and outputs that you want to see in your simulation. The third and forth lines in our example are the patterns for the inputs. They tell the simulator to use A = 0, B = 0 in the first cycle, A = 0, B = 1 for the second cycle and so on. Finally, the last line tells the simulator to run the command file. An alternative command could be *cycle*, which repeats the simulation and is more useful for systems with feedbacks.

Note: For small schematics, all the possible combinations for inputs are usually tested (like in the example above), however in larger schematics it is impractical to test all the combinations. Instead, we find a reasonable number of combinations that can accurately test our schematic (usually about 16 combinations is sufficient for up to 10 input pins).

31. Save the file as *c-xor21.cmd* in your project folder.
32. Go back to *Tools*→*Create Digital Netlist (common)*. Check the *Simulate After Netlisting* checkbox and click on *Browse* to find the *c-xor21.cmd* command file that you just created.
33. Click on *Create Netlist* and wait for the simulation results (multiple windows will open, the last being the simulation results).
34. Your schematic for our example should look like this:



The main window shows the cycle period as well as the value of each input and output pin during the simulation. As you can see, in the first cycle A and B are both zero (the simulation line is low) and so is Q the output for our *XOR*. However in the second cycle, A = 0 and B = 1 (the line is raised) and Q = 1. If the schematic is correct, with the given inputs, this simulation should contain the same results as the *XOR* truth table.

35. You can now close the simulation program and the other four windows that opened with it during the simulation process.

Note: Closing the windows after simulation too quickly may cause the application to freeze up. As well, simulating with circuits open in DxDesigner can cause freezing. In case of freezing, do not restart the computer. This will cause the application to not be able to unlock files that are in use and may destroy your project folder (and all your work in the class). If this happened go to a TA in the lab for help.

Note: If files do become locked while using the application. First try restarting the program. If this doesn't work, turn off the program and in the project folder look at properties of the file related to the locked circuits. The file can be unlocked from the properties window for the file (found be right clicking on the file and selecting properties). Manipulating files manually without the use of the MentorGraphics is not recommended. Any manipulation to the internal files of the project without the use of the program is done at your own risk.

Other Useful Information

How to print your schematic:

- Use Zoom/Pan to zoom your schematic to full window size. This is to make sure that you get the largest print possible on one page.
- Add your name to the schematic by going to *Add*→*Text*, and click on the schematic where you want the text placed.
- Go to *File*→*Print Setup* and select *Landscape*. Click *OK*.
- Go to *File*→*Print* and select *window* instead of *page*.
- Choose the desired printer.
- Use *Preview* to verify what you are printing.
- Click on *Print*.

How to print your simulation/command file:

- Simply use *File*→*Print* to print the simulation. The command file can be easily printed using the *Text Editor* print option. Try to be environmentally friendly and keep the number of printed pages to the minimum!

Using the custom components in other schematics:

- It is possible to use the custom symbols with their respective components in other projects just as we used the built-in components before. Simply use *Add*→ *Component* and choose your component from the project folder rather than a builtin component. Make sure you make a symbol for every custom component you create.
- If changes are made to any components used in a schematic, after the internal component is saved, you will notice a pink border around all the symbols in the schematic that are affected by this change. In order to update your schematic to use the newest version of the component, right-click on the component(s) with the pink border in your schematic, choose *Component Update (selected)*→*Apply All Symbol Updates*.

restart
wave c-xor21.wfm A B Q
pattern A 0 0 1 1
pattern B 0 1 0 1
run

Expected Results:

Q 0 1 1 0

Component	Inputs	Outputs	Name of component	
			builtin	74f
input pin	---	---	in.1	use builtin
output pin	---	---	out.1	use builtin
and	2	1	and2.1	08.1
and	3	1	and3.1	11.1
and	4	1	and4.1	21.1
nand	2	1	nand2.1	37.1
nand	3	1	nand3.1	10.1
nand	4	1	nand4.1	20.1
nor	2	1	nor2.1	02.1
nor	3	1	nor3.1	---
nor	4	1	nor4.1	---
not	1	1	not.1	04.1
or	2	1	or2.1	32.1
or	3	1	or3.1	---
or	4	1	or4.1	---