

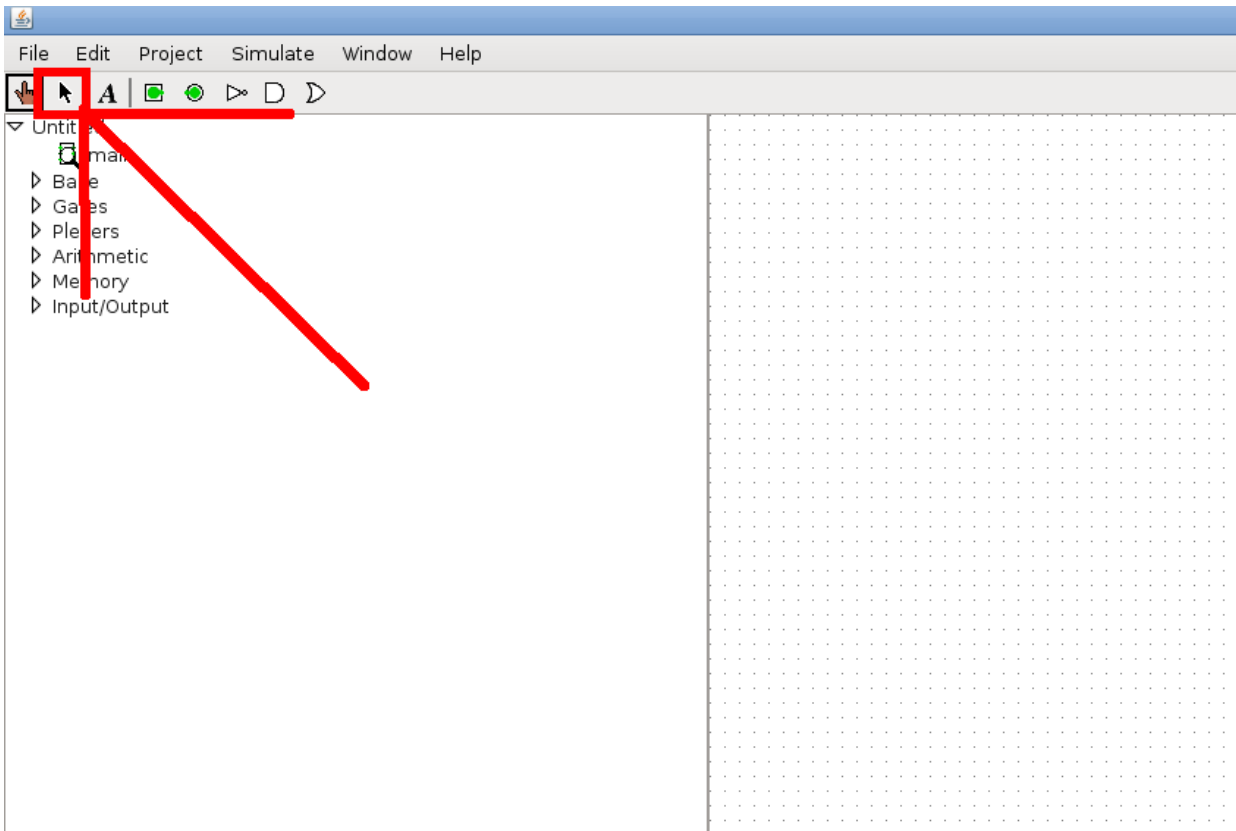
Logisim Tutorial

Last Updated: 9/6/09

1 Getting Started

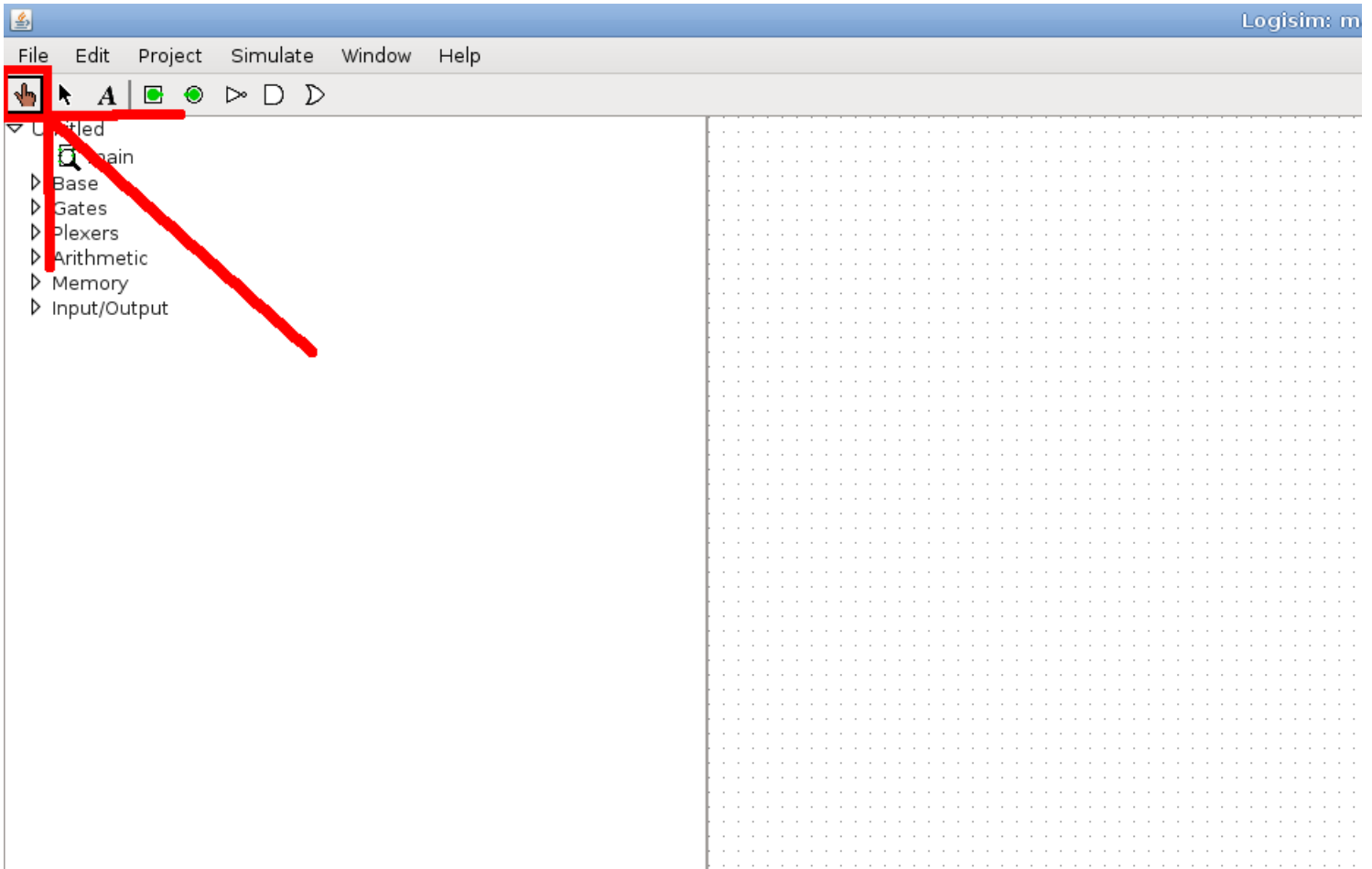
Welcome to Logisim! This short tutorial is intended to describe all the features of Logisim that may not be intuitive, so it is recommended that you have Logisim open while reading this tutorial. To run get started, type "logisim_template" into a terminal. This will copy template.circ into your current directory, which has a subcircuit that you'll need for risc. To start logisim, just type "logisim" into a terminal. Open template.circ and save it as whatever you want your project to be named. When you are finished, type "cs031_handin risc_logisim" into a terminal.

2 Edit Tool



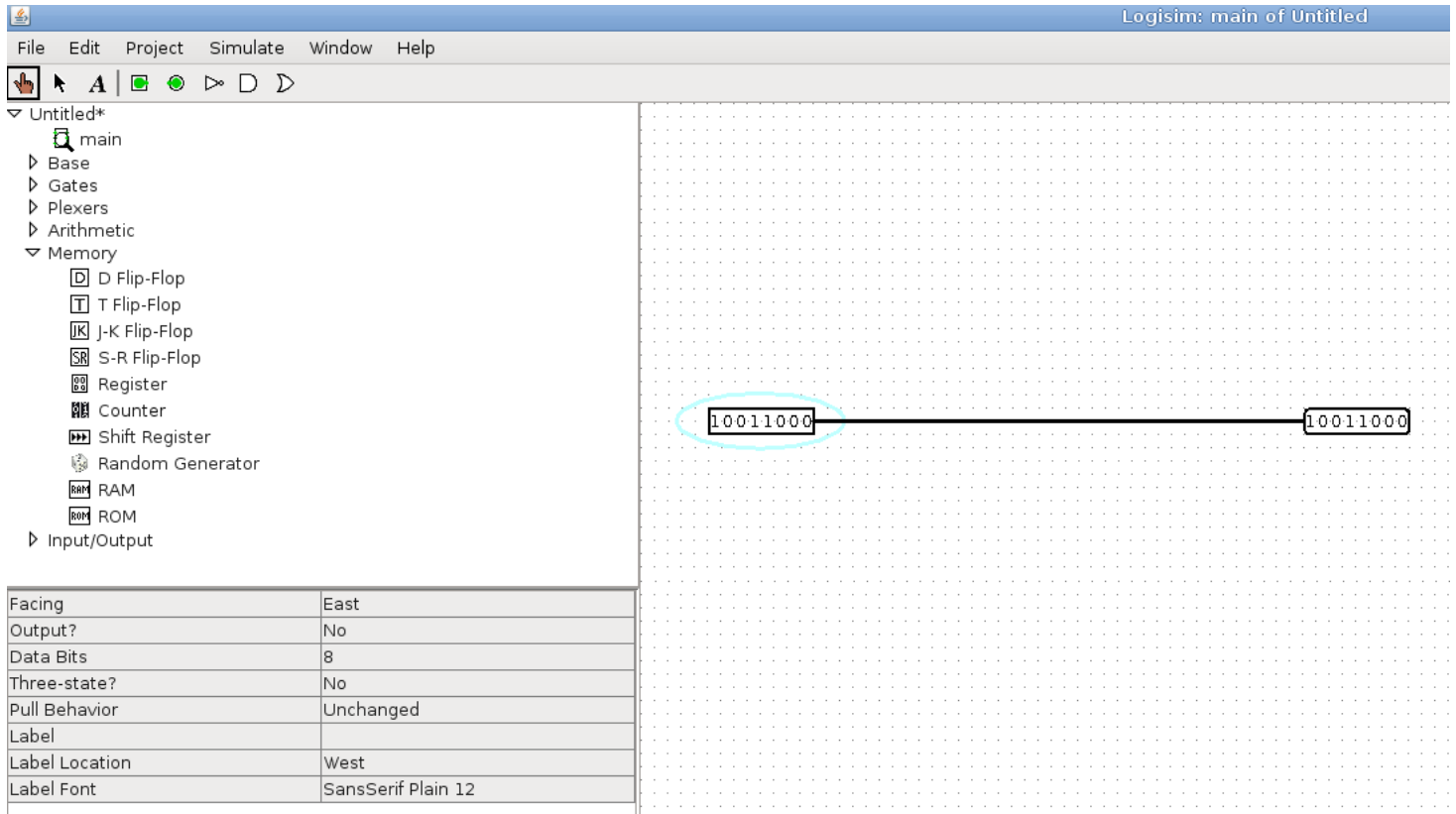
The Edit Tool is what you must use if you are altering your circuit in any way. This includes placing components, dragging components, or creating wires (by clicking and dragging). You can also use this to change the attributes of a component, as described later.

3 Poke Tool



The Poke Tool is what you use to test your circuit. Using the poke tool, you can click on wires to see their current value, and toggle switches.

4 Input and Output

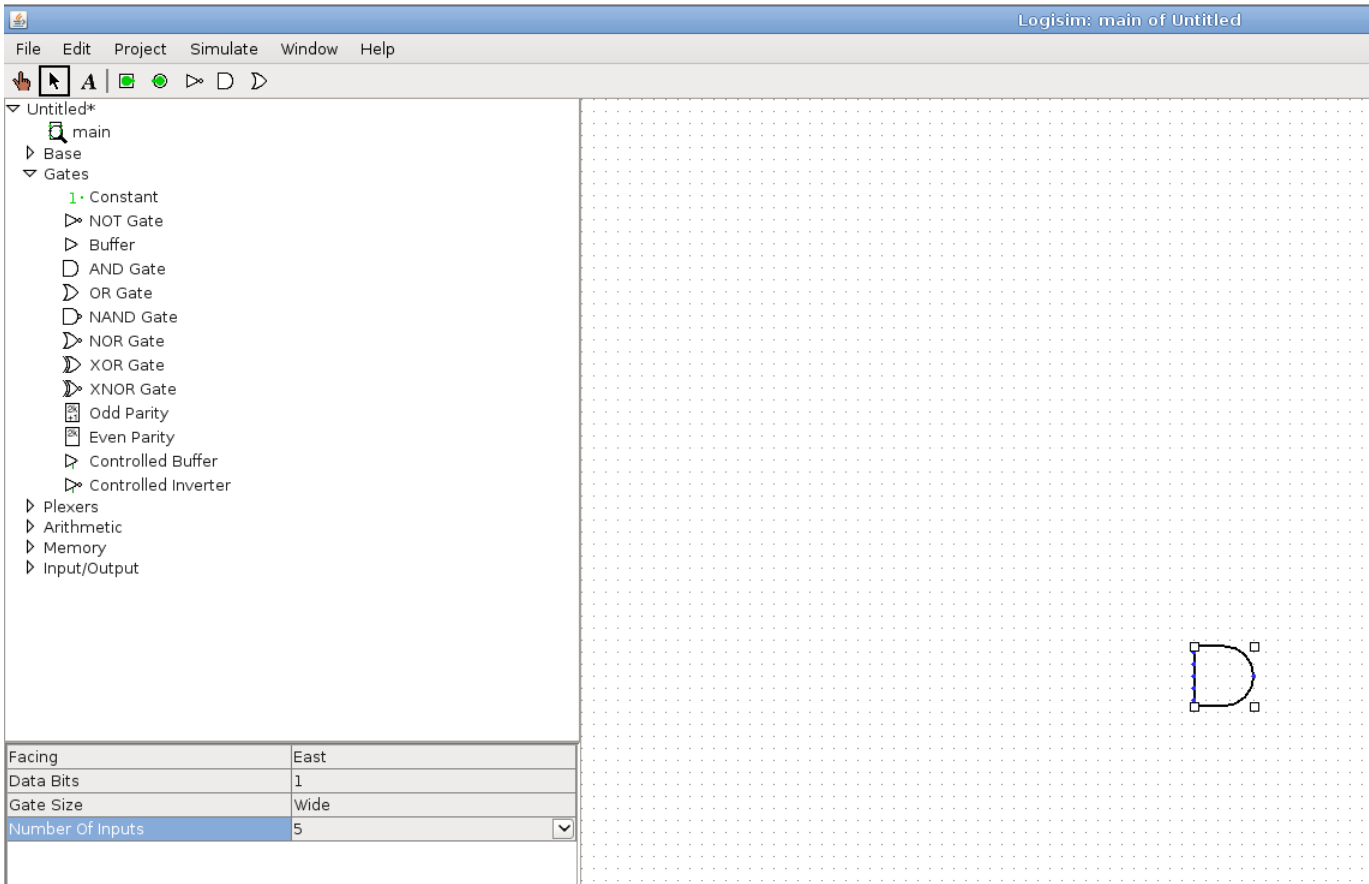


The screenshot shows the Logisim software interface. The top menu bar includes File, Edit, Project, Simulate, Window, and Help. The toolbar contains various tools for editing and simulation. The left sidebar shows a project tree with components like D Flip-Flop, T Flip-Flop, J-K Flip-Flop, S-R Flip-Flop, Register, Counter, Shift Register, Random Generator, RAM, and ROM. The main workspace shows a wire with two 8-bit pins. The left pin is circled in cyan and labeled '10011000'. The right pin is labeled '10011000'. A table in the bottom left corner shows the attributes of the selected pin.

Facing	East
Output?	No
Data Bits	8
Three-state?	No
Pull Behavior	Unchanged
Label	
Label Location	West
Label Font	SansSerif Plain 12

The basic input and output consists of pins and the clock. The input and output pins can be found on the top toolbar. Here is an 8 bit input pin and an 8 bit output pin. Note the attributes in the bottom left corner. You can change the attributes by clicking on an object with the edit tool, and then clicking on the appropriate attributes. If you click a tool in the menu and then edit its attributes, that changes the default attributes of that tool. The two important attributes shown here are "Facing" which sets the direction, and "Data Bits" which sets how many bits are on the input and output lines. The components on two different sides of the same wire should expect the same number of bits. For example, if on one end of the wire something is outputting 8 bits of data, while the other end only expects one, the wire will turn Orange to warn you of this.

5 Gates



Here we have a basic gate. This can be found in the "Gates" menu on the top left. Here, we can set how many inputs the gate takes, and the data bits. Note that the number of data bits here specifies the number of data bits going into each input, and coming out of the output.

6 Splitter

The screenshot shows the Logisim software interface. On the left is a component palette with a tree view showing 'Base' components, including the Splitter. Below the palette is a properties table for the selected Splitter component. On the right is a circuit workspace with a grid background, showing a circuit diagram of an 8-bit data bus being split into three outputs: a 3-bit data bus, a 4-bit data bus, and a 1-bit data bus.

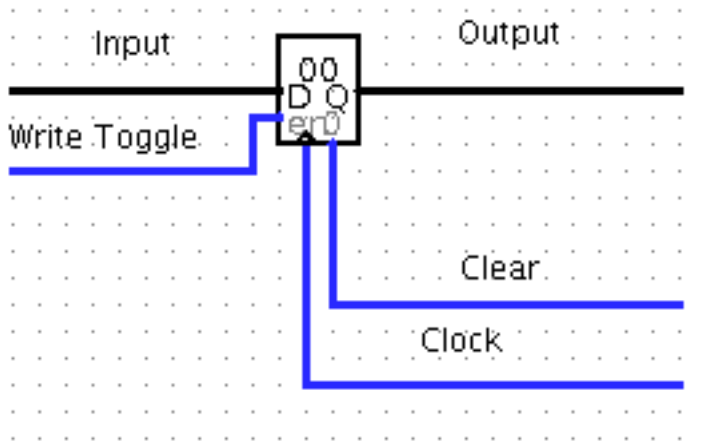
Facing	East
Fan Out	3
Bit Width In	8
Bit 0	0 (Top)
Bit 1	0 (Top)
Bit 2	0 (Top)
Bit 3	1
Bit 4	1
Bit 5	1
Bit 6	1
Bit 7	2 (Bottom)

The circuit diagram shows an 8-bit data bus on the left, labeled '8 bits of data'. It splits into three outputs on the right: a top output labeled '3 bits of data', a middle output labeled '4 bits of data', and a bottom output labeled '1 bit of data'. The splitter component is represented by a small square with a blue arrow pointing to the right.

The splitter is a very important component in Risc, because we need to split an 8 bit command into its component parts to interpret it. You can find a splitter in the "Base" menu on the top left. Experiment with the splitter to get a feel for how the attributes affect it. If you think doing things like splitting an 8-bit wire into two 3-bit wires and a 2-bit wire would make your circuit more confusing, you can just split

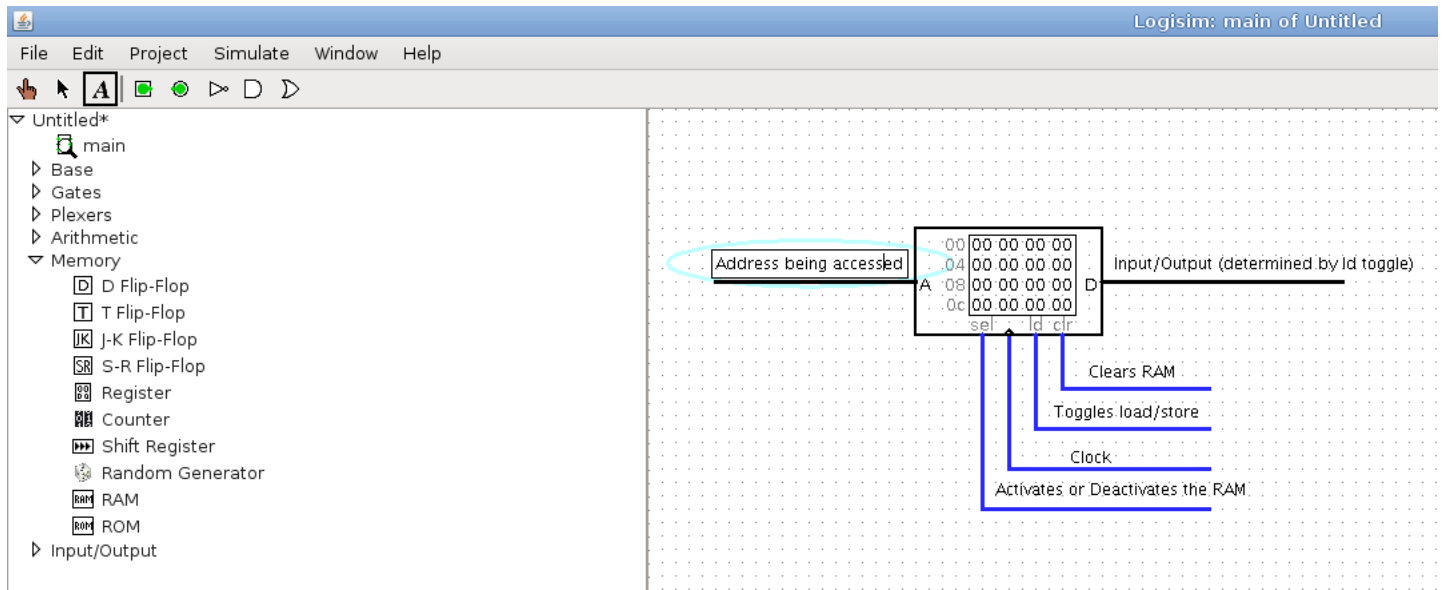
the 8-bit wire into eight 1-bit wires and go from there.

7 Register



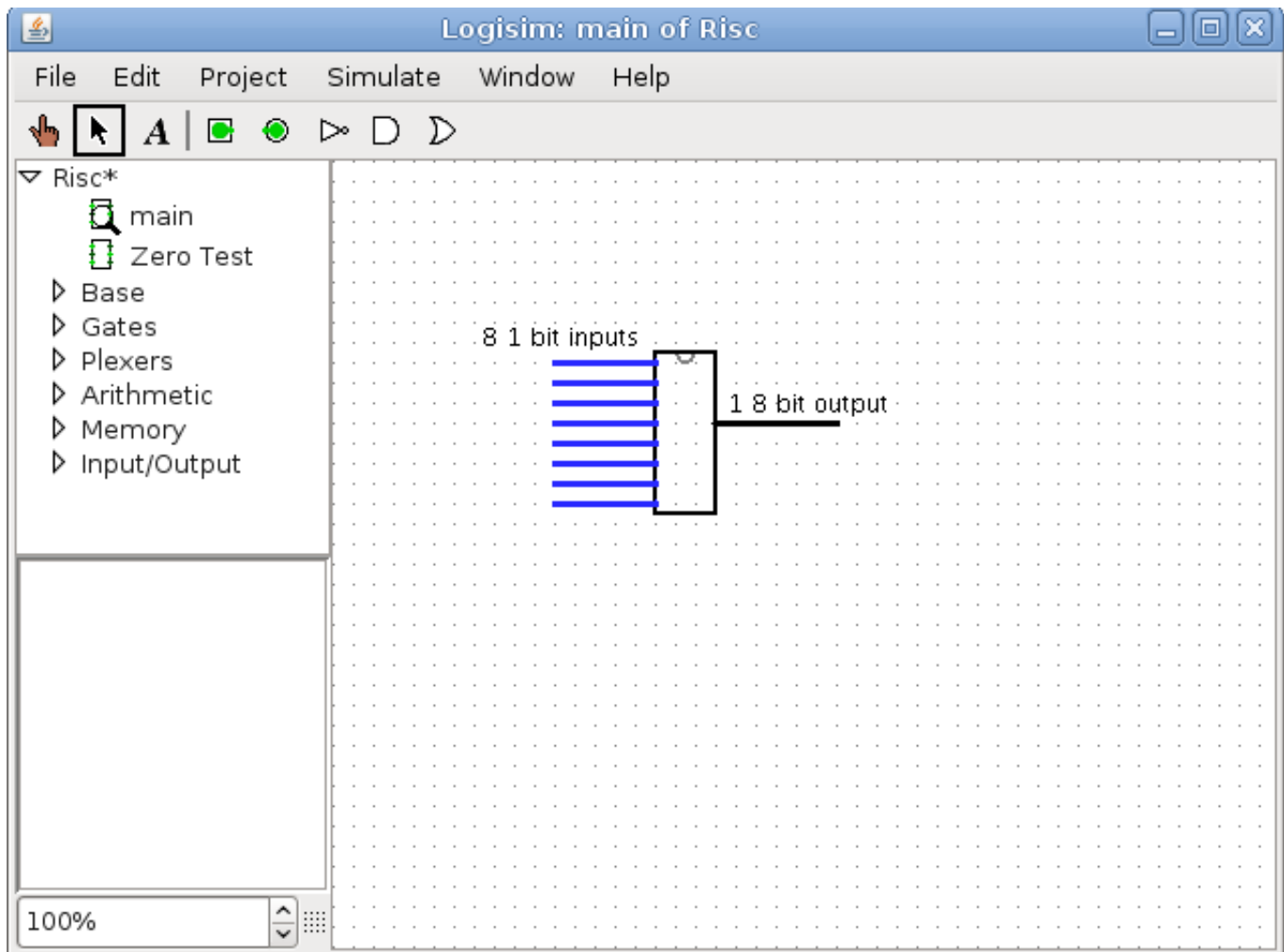
The image here is pretty self explanatory. It's here more as a reference. Experiment with the register attributes. Note that it doesn't have a "Facing" attribute. The designers haven't added this feature for registers yet. When the register has a value stored in it, the value shows up on the register itself.

8 RAM



Again, this picture is pretty self explanatory. See lecture notes about memory if this is confusing. One noteworthy feature here is that you can see the memory at all times. You can also change the memory directly by selecting the poke tool, clicking on the block of memory you want to change, and then typing a hexadecimal number. You can also right click on the memory and click "Edit Contents" or "Load Image". If you click on "Save Image" you can save your memory configuration. The RAM expects a ".img" file, so you should save as a ".img".

9 Zero Test/Subcircuits



Logisim didn't have a zero test like diglog had, so we made one as a subcircuit. Do not make additional subcircuits, because it would make your project impossible to grade. You can double click on the "Zero Test" subcircuit in the top left to see what we created for you, and then double click on "main" to return to your original circuit. What we provided is a circuit that tests 8 separate bits, and returns one if they are all zero, and zero otherwise.

10 Final Notes

After adding a clock, you can click on "Simulate" on the menu bar to start the clock and change the speed. Ticks are disabled by default, so enable them to start the clock.

IMPORTANT: For Risc, you are not allowed to use the subtractor or the comparator. You may use an inverter for subtraction, but in Logisim you don't need to add one. For example, it will turn 7 into -7, rather than into -8 as it would if you just flipped the bits.