Verilog - Data Types

There are three major data types

- reg, net, parameter
- All can be vectors (multi-bit signal or bus)
- The default is scalar

Verilog - Data Types

Reg data type

- A reg data type represents a variable in Verilog
- Type ”reg” variable stores values, but not necessarily a FF (register)
- Are only assigned in an ”always” block task or function
- If a reg data type is assigned a value in an always block that has a clock edge expression, a flip-flop is inferred.

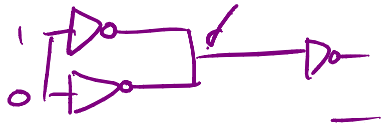
```
always_ff(posedge clk)
  q_out <= d_in;  //infers a d-type flip flop
```

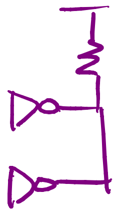
Verilog - Data Types

Net data type

- Net data types represent different kinds of electrical interconnects between hardware elements
- There are 10 different sub types of the net type
- All have no sense of memory, except trireg

Verilog - Data Types





Net Data Sub-Types

Name	Description
wire (tri)	A wire connecting one driver to other gates input. Types wire and tri are identical. Two different names are used as hint to intent. Type tri implies a tri-state buffer is driving the line.
tri0	Like tri but when the resultant value is to be Z, it resolves to resistive pulldown.
tri1	Like tri but when the resultant value is to be Z, it resolves to resistive pullup.
wand (triand)	Type indicates this wire has multiple drivers. (tri-state) The drivers are configured in a wired-and configuration.
wor (trior)	Type indicates this wire has multiple drivers. (tri-state) The drivers are configured in a wired-or configuration.
supply0	Vss, the circuit ground.
supply1	Vdd, the circuit Vdd power supply.
trireg	A tri-state capacitive wire. It holds its old value when resultant value is Z. You can specify how long the value is held.

Verilog - Data Types

Parameter data type

- Represents constants
- Declared by: parameter delay $=$ 5;
- Can be overwritten when a module is instantiated (parametrized modules)
- Cannot be changed at run time