Verilog Assignment

- Continuous Assignment (Data flow Design)
- Procedural Assignment (Behavioral Design)
  - Blocking Assignment ‘=‘
  - Nonblocking Assignment ‘<=‘

Verilog Assignments

- Continuous Assignment (Dataflow Design)
  - LHS must be wire
- Procedural Assignment (Behavioral Design)
  - within always block
  - LHS must be reg

```
module some_circuit (
        // inputs and outputs
    );
        LHS          RHS
    assign x = (a & b) | c;

    always @(*)
    begin
        LHS          RHS
        y = (a & b) | c;

    end
endmodule
```

Continuous

Procedural

baiyh@sustech.edu.cn

Procedural Assignments

- Blocking assignment (=):
  - Executes sequentially, causing the next assignment to wait until the current assignment is completed.
- Nonblocking assignment (<=):
  - Executes concurrently, allowing multiple assignments to happen simultaneously without waiting for each other.

```
module some_circuit (
    // inputs and outputs
);

always @(*)
begin
    y = (a & b) | c;
end
endmodule
```

Blocking Assignment

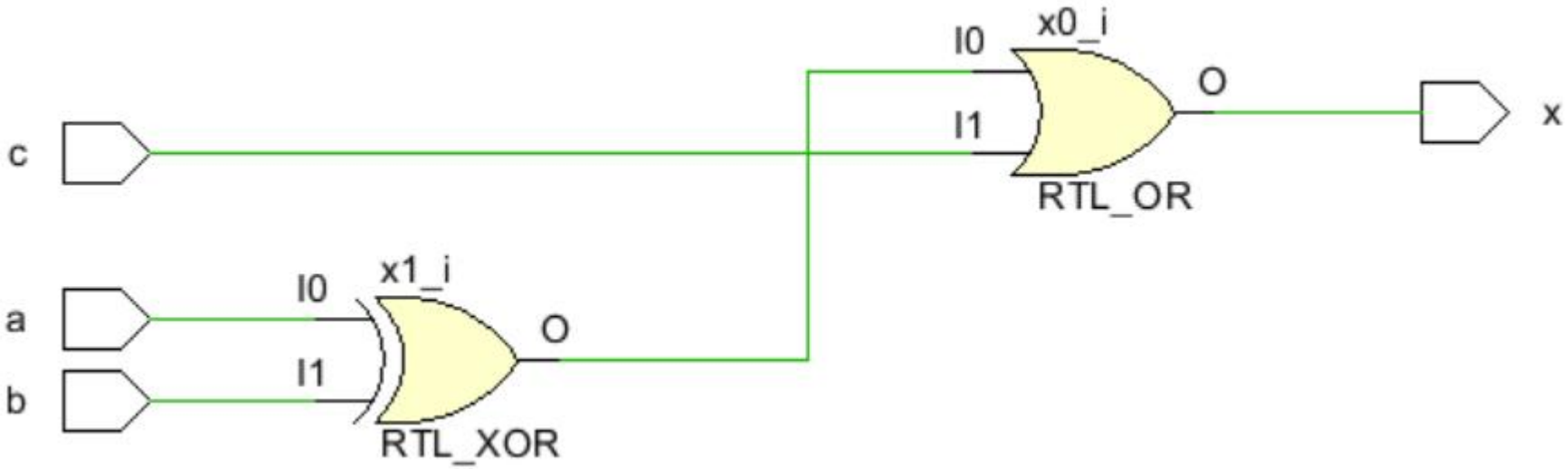
```
module some_circuit (
    // inputs and outputs
);

always @(*)
begin
    y <= (a & b) | c;
end
endmodule
```

Nonblocking Assignment

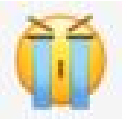
Combinational logic + blocking

```
// Blocking assignments
always @(a, b, c)
begin
    x = a;
    x = x ^ b;
    x = x | c;
end
```

The order matters!

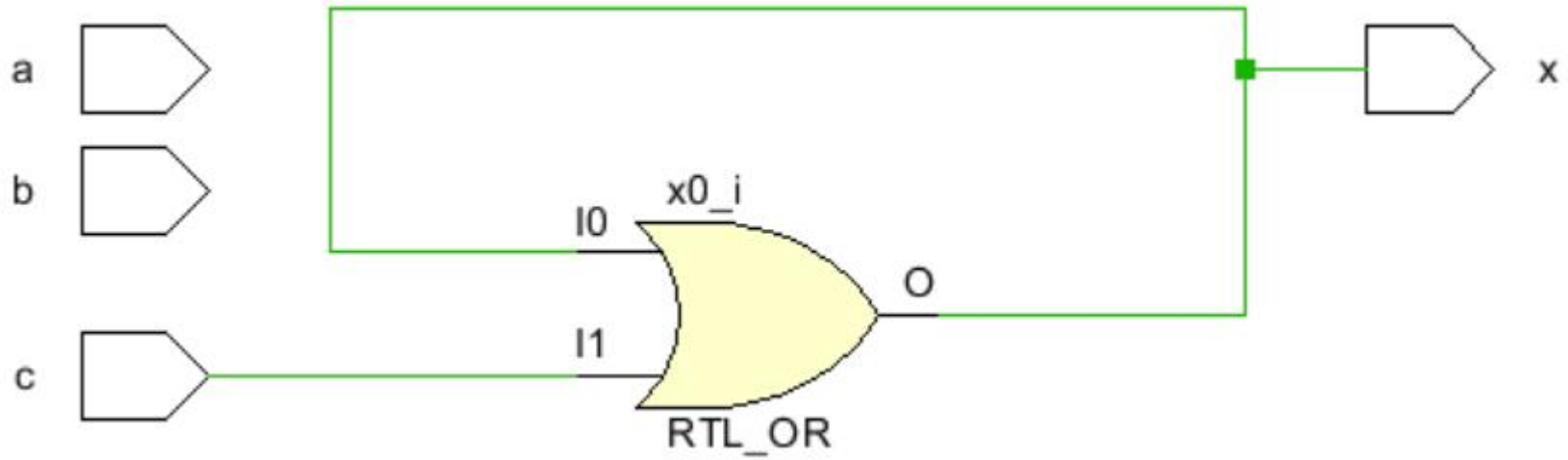


Combinational logic + nonblocking



```
// Nonblocking assignments
 always @(a, b, c)
 begin
      x <= a;
      x <= x ^ b;
      x <= x | c;
 end
```

non sense circuit
should not design like that

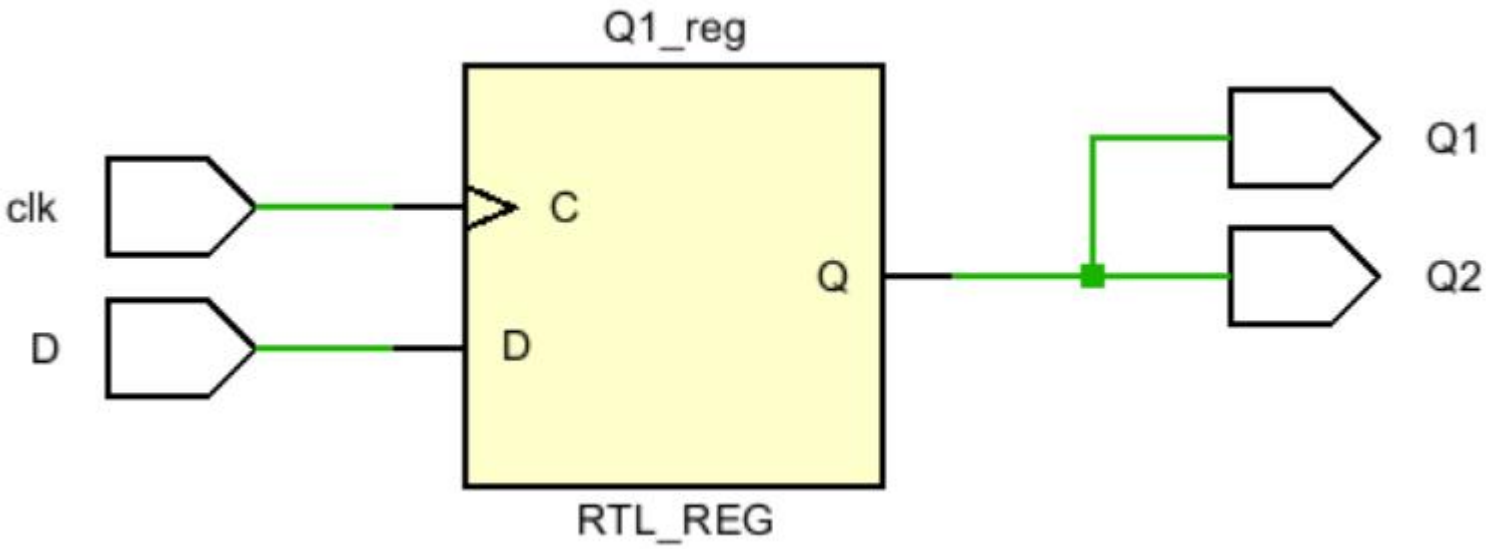


baiyh@sustech.edu.cn

Sequential logic + blocking

```
// Blocking assignment
always @(posedge clk)
begin
    Q1 = D;
    Q2 = Q1;
end
```

You must change code order
to make it working properly

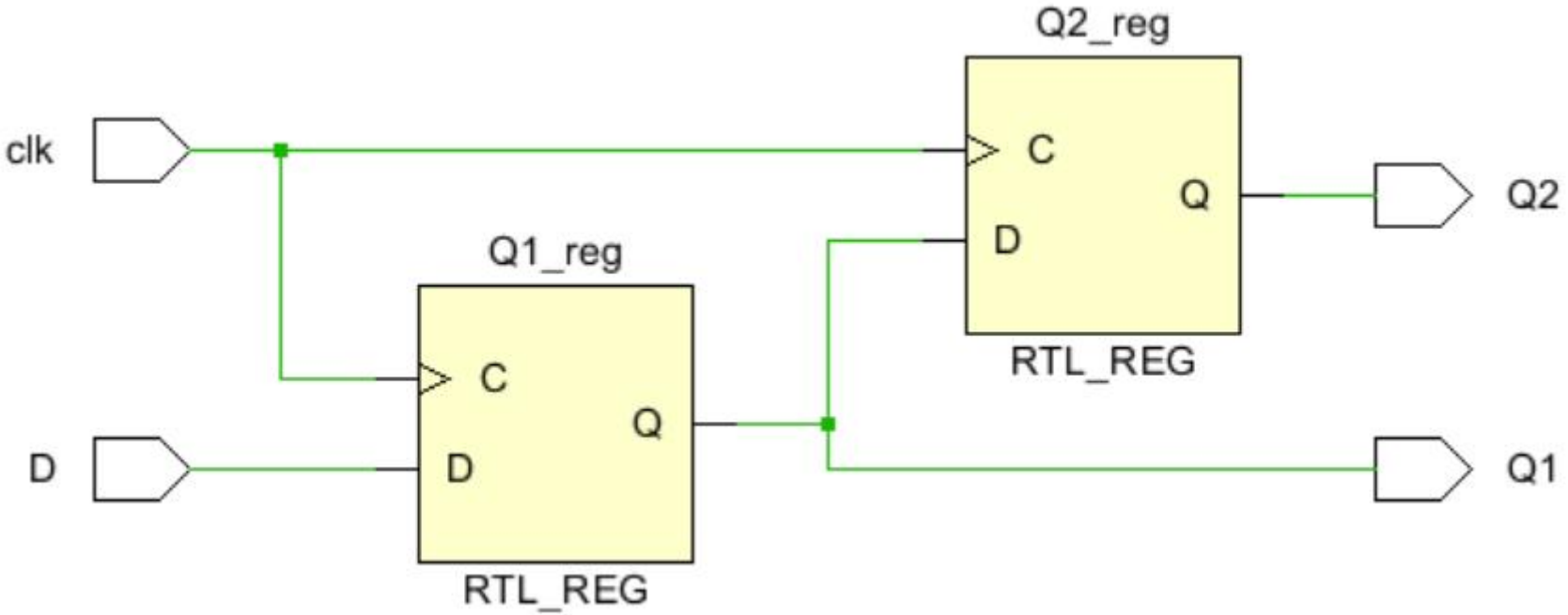


baiyh@sustech.edu.cn

Sequential logic + nonblocking

```
// Nonblocking assignment
always @(posedge clk)
begin
    Q1 <= D;
    Q2 <= Q1;
end
```

shift register with 2
cascade DFFs
the order doesn't matter



baiyh@sustech.edu.cn

Assignment suggestions

- Blocking assignments for combinational logic
  - continuous assignment
  - within always @(*) blocks.
- Nonblocking assignments for sequential logic
  - within always @(posedge clk)
  - or always @(negedge clk) blocks
- Don’t mix blocking and nonblocking together within same always block