Hardware Description Language

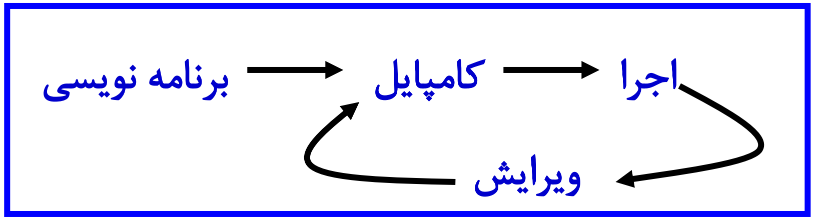
HDL

Hardware Description Language

- **HDL**
  - ➢ Describes circuits and systems in text.
    - – As a software program.
  - ➢ Can be processed by computers.
  - ➢ VHDL, Verilog, AHDL, SystemC.
- **Applications:**
  - ➢ Simulation
  - ➢ Synthesis

Simulation and Synthesis

- Simulation
  - Input waveform and circuit description --> Output waveform
  - Predicts circuit behavior before fabrication (debug before physical implementation).



برنامه نویسی

کامپایل

اجرا

ویرایش

Verilog HDL

- Verilog or VHDL?
  - Similar concepts, different syntax
- Similar to C/C++ Syntax
  - Case sensitive,
  - Comments: //
  - ; at the end of each statement

Example

- **module:**
  - Hardware component
- **Ports:**
  - As function parameters
    - But must specify input or output
- **wire:**
  - Intermediate signals

```
//HDL Example 3-1
//--------------------------
module smpl_circuit(A,B,C,x,y);
   input A,B,C;
   output x,y;
   wire e;
   and g1(e,A,B);
   not g2(y, C);
   or  g3(x,e,y);
endmodule
```

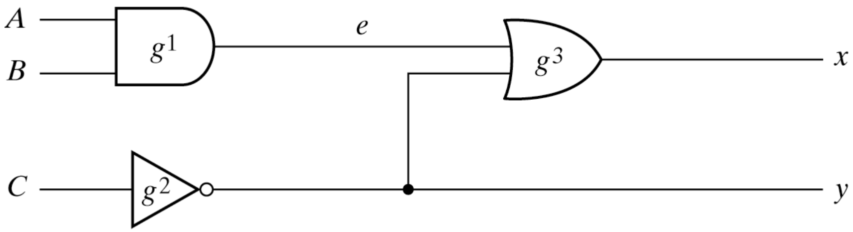
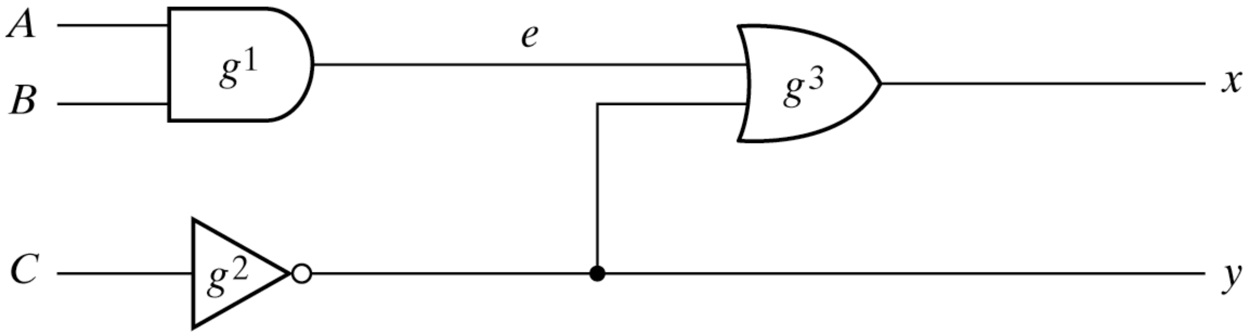


Fig. 3-37 Circuit to Demonstrate HDL

- **Concurrent Execution**
  - **Order not important.**

Example



```
//HDL Example 3-1
//-----------------------------
module smpl_circuit(A,B,C,x,y);
   input A,B,C;
   output x,y;
   wire e;
   and g1(e,A,B);
   not g2(y, C);
   or  g3(x,e,y);
endmodule
```

- Concurrent Execution
  - Order not important.

Delays

- **Timescale:**
  - `` `timescale 1ns/100ps ``
    - **`1ns:`** time unit
    - **`100ps:`** precision (unit used for rounding)
    - Default: 1ns/100ps (0.1 ns)
- **Gate Delays:**
  - Used by simulator to generate correct waveforms

Delays

```
//HDL Example 3-2
//-----------------------------------
//Description of circuit with delay
module circuit_with_delay (A,B,C,x,y);
   input A,B,C;
   output x,y;
   wire e;
   and #(30) g1(e,A,B);
   or  #(20) g3(x,e,y);
   not #(10) g2(y,C);
endmodule
```

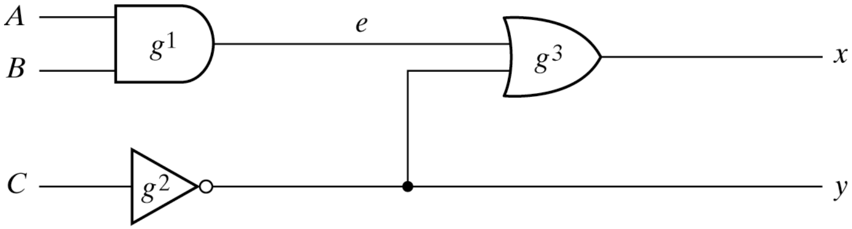


Fig. 3-37 Circuit to Demonstrate HDL

Testbench

- **Testbench:**
  - Program written to test the circuit
  - Sets the input values,
    - Changes them over time.
  - Instantiates the module(s).
  - Inputs by **`reg`** (generally in sequential constructs)
  - Outputs by **`wire`**

Testbench Example

```
//HDL Example 3-3
//----------------------
//Stimulus for simple circuit
module stimcrct;
reg A,B,C;
wire x,y;
circuit_with_delay cwd(A,B,C,x,y);
initial
   begin
        A = 1'b0; B = 1'b0; C = 1'b0;
     #100
        A = 1'b1; B = 1'b1; C = 1'b1;
     #100  $finish;
   end
endmodule


//Description of circuit with delay
module circuit_with_delay (A,B,C,x,y);
   input A,B,C;
   output x,y;
   wire e;
   and #(30) g1(e,A,B);
   or  #(20) g3(x,e,y);
   not #(10) g2(y,C);
endmodule
```

Testbench:

- Doesn’t have port.
- Instantiation.
- <size>’<base><value>
  - Base: b, o, d, h.
- ABC = “000” → “111”
- 100ns delay.
- Finishes at 200ns.

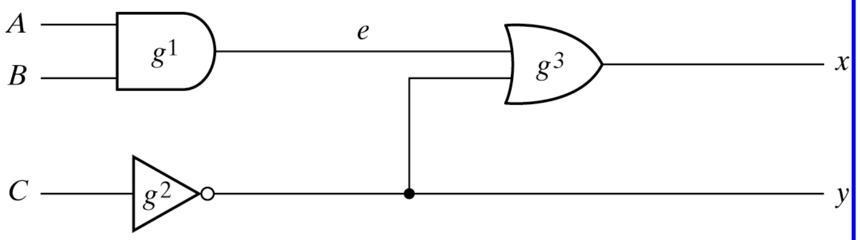


Fig. 3-37 Circuit to Demonstrate HDL

Testbench Example

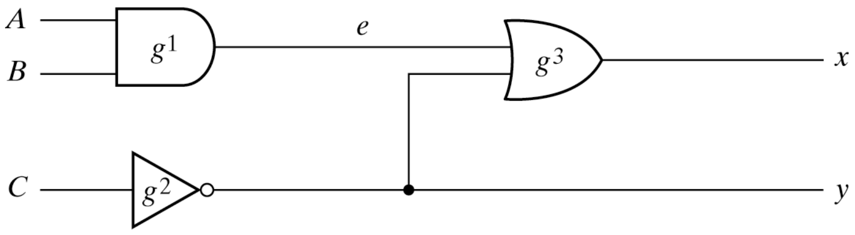


Fig. 3-37 Circuit to Demonstrate HDL

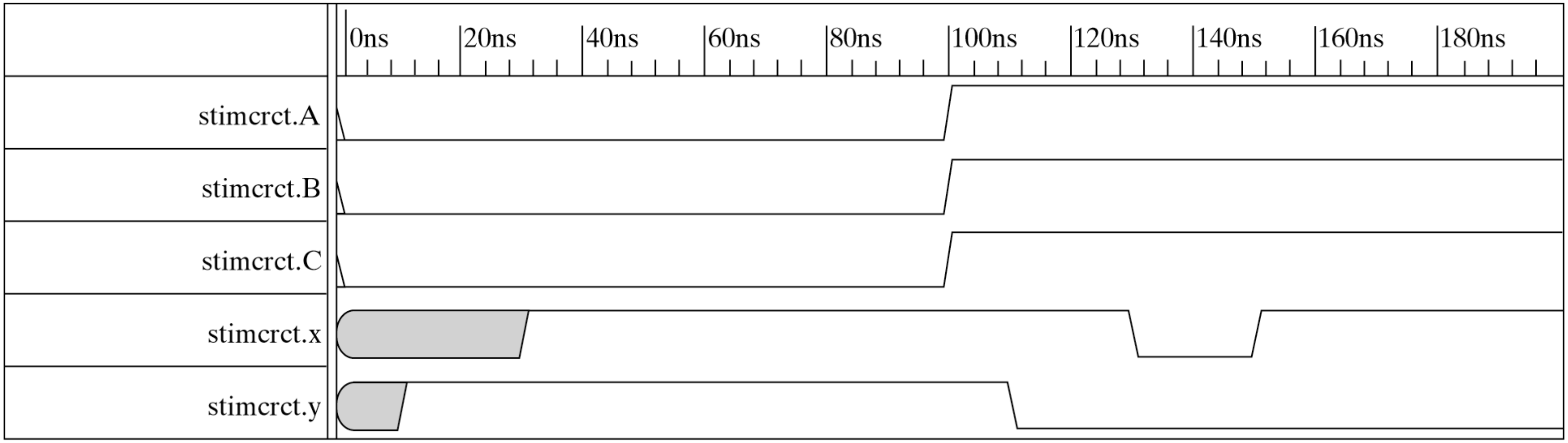


Fig. 3-38 Simulation Output of HDL Example 3-3

Primitives

- **Built-in Primitives:**
  - ➢ and, or, not, nand, nor, xor, xnor, buf.
- **User-Defined Primitives (UDP):**
  - ➢ User can define by truth table.

Simulation

- **ModelSim:**
  - An industrial and widely used simulator
  - $25K