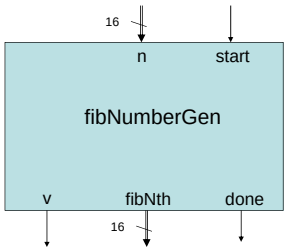
Verilog Behavioral Modeling Example

Problem Statement

- Create and test a module that computes Fibonacci numbers
- Definition of Fibonacci numbers:
  fib(0) = 0
  fib(1) = 1
  fib(i) = fib(i-1) + fib(i-2) for i >= 2

The fibNumberGen module



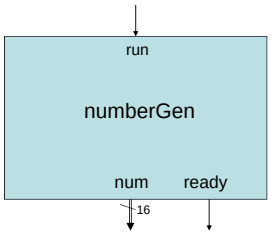
When a 0 to 1 transition occurs on **start**, the module computes **fibNth** = fib(n). When the result is stable on **fibNth**, **done** is held high for 10 nanoseconds. If the computation results in an overflow (i.e. if fib(n) is too large to represent in 16 bits) **v** (overflow flag) is set remains set until the next 1 to 0 transition of the done flag.
The entire computation, from the leading edge of the start signal to the leading edge of the done signal takes 100 nsec.

The Verilog Code for fibNumberGen

```
//Module to generate nth Fibonacci
//Number

module fibNumberGen(n, start, done, v,
fibNth);

 input [15:0] n;
 input start;
 output done, v;
 output [15:0] fibNth;
 reg done, v;
 reg [15:0] fibNth;

 reg [15:0] count, prevNum, temp;

 initial
  begin
    done = 0;
    v = 0;
 end
```

```
always @(posedge start)
    begin
      #95 if (n == 0) fibNth = 0;
       else if (n == 1) fibNth = 1;
              else begin
                count = n;
                prevNum = 0;
            for (fibNth = 1; count > 1;
                count = count -1)
            begin
              temp = fibNth;
              fibNth = fibNth + prevNum;
                  if (fibNth < prevNum) v = 1;
            prevNum = temp;
            end
          end
      #5 done = 1;
      #10 done = 0;  v = 0;
  end
endmodule
```

A Module to “Exercise” fibNumberGen



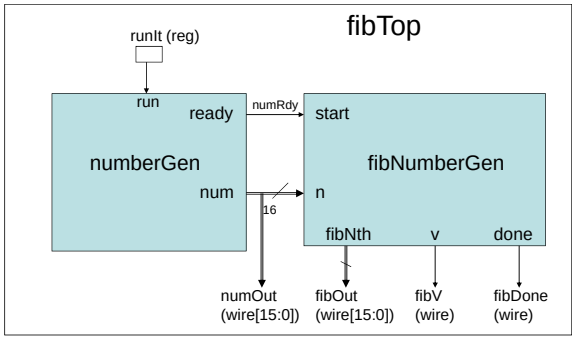
As long as **run** is high, this module generates successive values 0, 1, 2, 3… on output **num**. A new output value is generated every 150 nsec. and the **ready** flag held high for 10 nsec when a new value is stable on the **num** output.

Verilog Code for Module numberGen

```
// module to exercise Fib.Num Generator

module numberGen(num, ready, run);

  output [15:0] num;
  output ready;
  input run;
  reg [15:0] num;
  reg ready;


  initial
    begin
      num = -1 ;
      ready = 0;
    end
```

```
always
    begin
    wait(run ==1);
    #145 num = num + 1;
    #5  ready = 1;
    #10 ready = 0;
   end
endmodule
```

Putting the Modules Together



Verilog Code for Module fibTop

```
module fibTop();

reg runIt;
wire [15:0] numOut, fibOut;
wire numRdy, fibDone, fibV;

numberGen M1(numOut, numRdy, runIt);
fibNumberGen M2(numOut, numRdy,
                fibDone, fibV, fibOut);


initial
  begin
    #1;
    runIt = 1;
  end
```

```
always @(posedge fibDone)
  begin
    if (fibV ==1)
      begin
        $display($time,,
"Overflow at Fib(%d).  Simuation terminated",
numRdy);
        $finish;
      end
    else
      $display($time,,"Fib(%d) is: %d",
                numOut, fibOut);
    wait(fibDone == 0);
  end
endmodule
```

Compiling and Running the Verilog Model

```
l-ecn004% vlog fibNumberGen.v
Model Technology ModelSim SE vlog 5.5b Compiler 2001.05 May 23 2001
-- Compiling module fibNumberGen

Top level modules:
        fibNumberGen
l-ecn004% vlog numberGen.v
Model Technology ModelSim SE vlog 5.5b Compiler 2001.05 May 23 2001
-- Compiling module numberGen

Top level modules:
        numberGen
l-ecn004% vlog fibTop.v
Model Technology ModelSim SE vlog 5.5b Compiler 2001.05 May 23 2001
-- Compiling module fibTop

Top level modules:
        fibTop
l-ecn004% vsim -c fibTop
Reading /usr/local/apps/modeltech53d/bin/../hp700/../tcl/vsim/pref.tcl

# 5.5b

# vsim -c fibTop
# //  ModelSim SE 5.5b May  4 2001 HP-UX B.11.23
# //
...
# Loading work.fibTop
# Loading work.numberGen
# Loading work.fibNumberGen
VSIM 1>
```

Simulation Run of Module fibTop

```
VSIM 1> run 10000
#          251 Fib(   0) is:   0
#          411 Fib(   1) is:   1
#          571 Fib(   2) is:   1
#          731 Fib(   3) is:   2
#          891 Fib(   4) is:   3
#         1051 Fib(   5) is:   5
#         1211 Fib(   6) is:   8
#         1371 Fib(   7) is:  13
#         1531 Fib(   8) is:  21
#         1691 Fib(   9) is:  34
#         1851 Fib(  10) is:  55
#         2011 Fib(  11) is:  89
#         2171 Fib(  12) is: 144
#         2331 Fib(  13) is: 233
#         2491 Fib(  14) is: 377
#         2651 Fib(  15) is: 610
#         2811 Fib(  16) is: 987
#         2971 Fib(  17) is: 1597
#         3131 Fib(  18) is: 2584
#         3291 Fib(  19) is: 4181
#         3451 Fib(  20) is: 6765
#         3611 Fib(  21) is: 10946
#         3771 Fib(  22) is: 17711
#         3931 Fib(  23) is: 28657
#         4091 Fib(  24) is: 46368
#         4251 Overflow at Fib(  25).
Simuation terminated
# ** Note: $finish    : fibTop.v(23)
#    Time: 4251 ns  Iteration: 1  Instance:
/fibTop
```