module adder #(parameter W = 8) (  
  input clk,  
  input [W-1:0] inA, inB,  
  output [W:0] out,  
  output isOdd  
);  
  
reg [W-1:0]  regA, regB;  
reg [W:0]  regOut;  
reg  regOdd;  
wire [W:0]  wireOut;  
  
assign wireOut = regA + regB;  
assign out = regOut;  
assign isOdd = regOdd;  
  
always@(posedge clk)  
begin  
  regA <= inA;  
  regB <= inB;  
  regOut <= wireOut;  
  regOdd <= out[0];  
end  
endmodule  

Describe a module called **adder** which can be used to generate a family of circuits according to parameter **W** (default value is 8.) not a good name, since it is more than an Adder.

**Input/Output Ports**

Create physical wires (Even though it says “reg”!)

Generate an addition circuit, 
Inputs are **regA** and **regB** wires. 
Output is connected to **wireOut**.

Attach wire **out** to wire **regOut**; 
Attach wire **isOdd** to wire **regOdd**;

Create four positive edged-clocked registers, attacked to clock **clk**.

The outputs of the registers should be attached to the **regA,regB,regOut,regOdd** wires.

Inputs of registers should be attached to **inA, inB, wireOut, out[0]** wires.