[Lemmings4](https://hdlbits.01xz.net/wiki/Lemmings4)

|  |
| --- |
| module top\_module(  input clk,  input areset, // Freshly brainwashed Lemmings walk left.  input bump\_left,  input bump\_right,  input ground,  input dig,  output walk\_left,  output walk\_right,  output aaah,  output digging );    parameter w\_left=4'b0001, w\_right=4'b0010, dig\_left=4'b0011, dig\_right=4'b0100, fall\_left=4'b0101, fall\_right=4'b0110, splatter=4'b0111,keep\_fall\_left=4'b0000,keep\_fall\_right=4'b1000;  reg [3:0] state, next\_state;  reg [7:0] count;  always@(posedge clk or posedge areset)  begin  if (areset)  begin  state<=w\_left;  count<=0;   end    else  begin  state<=next\_state;  if ((count>8'd19)&&(ground))  state<=splatter;  else if (next\_state==keep\_fall\_right|next\_state==keep\_fall\_left)  count<=count+1;  else begin  count<=0;   end  end  end  always@(\*)  begin  case (state)  w\_left: begin  if(!ground)  next\_state<=fall\_left;  else if (dig)  next\_state<=dig\_left;  else if (bump\_left)  next\_state<=w\_right;  else  next\_state<=state;  end  w\_right: begin  if(!ground)  next\_state<=fall\_right;  else if (dig)  next\_state<=dig\_right;  else if (bump\_right)  next\_state<=w\_left;  else  next\_state<=state;  end  dig\_left: begin  if(!ground)  next\_state<=fall\_left;  else next\_state<=state;  end  dig\_right: begin   if(!ground)  next\_state<=fall\_right;  else next\_state<=state;  end  fall\_left: begin  if(ground)  next\_state<=w\_left;  else   begin   next\_state<=keep\_fall\_left;   end  end  keep\_fall\_left:  if(ground)  next\_state<=w\_left;  else   begin   next\_state<=keep\_fall\_left;   end  fall\_right: begin  if(ground)  next\_state<=w\_right;  else begin  next\_state<=keep\_fall\_right;  end  end  keep\_fall\_right:  if(ground)  next\_state<=w\_right;  else begin  next\_state<=keep\_fall\_right;  end  splatter: begin  next\_state<=splatter;   end  default: next\_state<=state;  endcase  end      always@(\*)  begin  if(state==fall\_left|state==fall\_right|state==keep\_fall\_right|state==keep\_fall\_left) begin digging<=0; walk\_left<=0; walk\_right<=0; aaah<=1; end  else if (state==dig\_left|state==dig\_right) begin digging<=1; walk\_left<=0; walk\_right<=0; aaah<=0; end  else if (state==w\_left) begin digging<=0; walk\_left<=1; walk\_right<=0; aaah<=0; end  else if (state==w\_right) begin digging<=0; walk\_left<=0; walk\_right<=1; aaah<=0; end  else if (state==splatter) begin digging<=0; walk\_left<=0; walk\_right<=0; aaah<=0; end  else begin digging<=0; walk\_left<=0; walk\_right<=0; aaah<=0; end  end   endmodule |