<https://courses.cs.washington.edu/courses/cse370/10sp/pdfs/lectures/regfile.txt>

// This is a Verilog description for an 4 x 16 register file

`timescale 1ns / 1ns

module regfile4x16a

(input clk,

input write,

input [2:0] wrAddr,

input [15:0] wrData,

input [2:0] rdAddrA,

output [15:0] rdDataA,

input [2:0] rdAddrB,

output [15:0] rdDataB);

reg [15:0] reg0, reg1, reg2, reg3;

assign rdDataA = rdAddrA == 0 ? reg0 :

rdAddrA == 1 ? reg1 :

rdAddrA == 2 ? reg2 :

rdAddrA == 3 ? reg3 : 0;

assign rdDataB = rdAddrB == 0 ? reg0 :

rdAddrB == 1 ? reg1 :

rdAddrB == 2 ? reg2 :

rdAddrB == 3 ? reg3 : 0;

always @(posedge clk) begin

if (write)

case (wrAddr)

0: begin

reg0 <= wrData;

end

1: begin

reg1 <= wrData;

end

2: begin

reg2 <= wrData;

end

3: begin

reg3 <= wrData;

end

endcase // case (wrAddr)

end // always @ (posedge clk)

endmodule