```
1
2   /*
3      Modular 2^depth x width register file with 1 read port and 1 write port.
4      Read port accesses r_data according to address r_addr.
5      The single write port writes w_data to address w_addr.
6   */
7
8   /*
9      DESIGN DECISION:
10     An internal register array was chosen rather than instatiating 16 register.v entities,
11     since register arrays are implemented using on-board FPGA memory, rather than costly FFs.
12  */
13
14  module reg_file #(
15     // for modularity
16     parameter depth = 4,
17     parameter width = 32
18     )(
19     input clk, clr, wr_en,
20     input [depth-1:0] r_addr, w_addr,
21     input [width-1:0] w_data,
22     output [width-1:0] r_data
23     );
24     integer i;
25
26     // Internal memory array consisting of 2^depth, width-lengthed registers.
27     // This definition enforces a right-to-left increasing bit significance, and an
   up-to-down addressing scheme.
28     reg [width-1:0] reg_array [0:2**depth-1];
29
30     // Default all contents to 0.
31     initial begin
32        for (i=0; i<2**depth; i = i+1) begin
33              reg_array[i] = 32'b0;
34        end
35     end
36
37     // r_data is immediately available as r_addr is presented to the RF (asynchronous read)
38     assign r_data = reg_array[r_addr];
39
40     // Data writes are synchronous to clk with wr_en high.
41     always @(posedge clk) begin
42        if (clr) begin
43           for (i=0; i<2**depth; i = i+1) begin
44                 reg_array[i] = 32'b0;
45           end
46        end
47
48        else if (wr_en) begin
49           reg_array[w_addr] = w_data;
50        end
51
52     end
53
54  endmodule
55
```