```
module regarray(
    input clk,
    input rst,
    input reg_read1,
    input reg_read2,
    input [2:0] addr1,//address
    input [2:0] addr2,
    input reg_write1,//write control
signal
    input reg_write2,
    input [15:0] data_in1,//three
data inputs
    input [15:0] data_in2,
    input [15:0] data_in3,
    output reg [15:0]
reg_out1,//output data from port 1
or 2
    output reg [15:0] reg_out2,
    output [15:0] port
    );
    reg [15:0] register [7:0];
    assign port = register[7];
    parameter num=8;//number of
registers
    integer i;
    always @ (posedge clk)
    begin
        if(!rst)
        begin
            for(i=0;i<num;i=i+1)

register[i]=16'h0000;//reset the
data in each register
        end
        else
if(reg_read1&&reg_read2)
begin

reg_out1=register[addr1];

reg_out2=register[addr2];
end
else if(reg_read1)
begin

reg_out1=register[addr1];
            reg_out2=16'hz;
end
else if(reg_read2)
begin
            reg_out1=16'hz;

reg_out2=register[addr2];
end
else
begin
            case({reg_write1,
reg_write2})
            2'b11:
            begin

register[addr1]=data_in1;

register[addr2]=data_in2;
            end
            2'b01:
            begin

register[addr1]=data_in3;
            end
            2'b10:
            begin

register[addr1]=data_in1;
            end
            endcase
        end
    end
endmodule
```