16th Day

Single Port RAM

Design

```
module sp_ram #(parameter DW=8,AW=10) (input clk,cs,wen,[DW-1:0]d_in,[AW-
1:0]addr,output reg [DW-1:0]d_out);


reg [DW-1:0] mem [0:(1<<20)-1];


always @(posedge clk)
begin
if(cs & wen)
  mem[addr]<=d_in;
else if(cs & (~wen))
   d_out<=mem[addr];
end



endmodule
```

Testbench

```
module ram_sp_tb();

parameter DW=8,AW=10;

reg clk,cs,wen;
reg [DW-1:0]  din;
reg [AW-1:0]  addr;
wire [DW-1:0] dout;
reg [AW-1:0] addr_mem [0:9];

sp_ram #(DW,AW) dut (clk,cs,wen,din,addr,dout);

always #5 clk=~clk;

initial
begin
clk=0;
cs=1;
wen=1;

for(integer i=0;i<9;i=i+1)
begin
addr_mem[i] = $random;
  addr = addr_mem[i];
  din=$random;
#10;
end

wen=0;
```

```
for(integer i=0;i<9;i=i+1)
begin
addr=addr_mem[i];
#10;
end

cs=0;
#50;

$finish;
end
```

Simulation