`timescale 1ns / 1ps

module main( rst,clk,clk\_1,

data,

wr\_1,

rd\_1,

full\_1,

empty\_1

);

input rst;

input clk;

output clk\_1;

input rd\_1;

input wr\_1;

inout [3:0] data;

output full\_1;

output empty\_1;

reg [3:0] data\_in;

wire [3:0] data\_out;

// Slow Clock Module instantiated...//

slow\_clock clocl\_inst1 (.clk(clk),.reset(rst),.SlowClk(clk\_1));

// FIFO Module instantiated..//

fifo write\_fifo (

.rst(rst), // input rst

.wr\_clk(clk\_1), // input wr\_clk

.rd\_clk(clk\_1), // input rd\_clk

.din(data), // input [3 : 0] din

.wr\_en(wr\_1), // input wr\_en

.rd\_en(rd\_1), // input rd\_en

.dout(data\_out), // output [3 : 0] dout

.full(full\_1), // output full

.empty(empty\_1) // output empty

);

assign data = (rd\_1) ? data\_out : 4'bz;

always @ (posedge clk\_1)

data\_in = data;

endmodule