// ==========================================================================

// CRC Generation Unit - Linear Feedback Shift Register implementation

// (c) Kay Gorontzi, GHSi.de, distributed under the terms of LGPL

// ==========================================================================

module CRC\_Unit(BITVAL, BITSTRB, CLEAR, CRC);

input BITVAL; // Next input bit

input BITSTRB; // Current bit valid (Clock)

input CLEAR; // Init CRC value

output [6:0] CRC; // Current output CRC value

reg [6:0] CRC; // We need output registers

wire inv;

assign inv = BITVAL ^ CRC[6]; // XOR required?

always @(posedge BITSTRB or posedge CLEAR) begin

if (CLEAR) begin

CRC = 0; // Init before calculation

end

else begin

CRC[6] = CRC[5];

CRC[5] = CRC[4] ^ inv;

CRC[4] = CRC[3];

CRC[3] = CRC[2] ^ inv;

CRC[2] = CRC[1];

CRC[1] = CRC[0] ^ inv;

CRC[0] = inv;

end

end

endmodule