Э.001.01. Конечный автомат для выполнения операции деления двух 4-битовых чисел. Текст программы. Приложение.

Листинг 1. Файл "fsm_div.v"

```
module fsm_div
(
    input reset,
    input clk,
    input valid_in,
    input [3:0] d_in,
    output reg [3:0] d_out,
    output reg valid_out,
    output reg error_out
);

// Константы ошибок
localparam NO_ERROR = 0, DIV_BY_ZERO = 1;

// Регистры операндов
reg signed [3:0] a_reg, b_reg;

// Состояния конечного автомата
localparam S0 = 0, S1 = 1, S2 = 2, S3 = 3;
reg [1:0] state;
initial state = S0;

always@(posedge clk)
begin
    if (reset)
        state <= S0;
    else
        case(state)
            // Сброс регистров
            S0: begin
                    a_reg <= 0;
                    b_reg <= 0;
                    d_out <= 0;
                    error_out <= 0;
                    valid_out <= 0;

                    state <= 1;
                end

            // Ввод первого операнда (делимого)
            S1: if (valid_in)
                    begin
                        a_reg <= d_in;
                        state <= S2;
                    end
```

```
            // Ввод второго операнда (делителя)
            S2: if (valid_in)
                    begin
                        b_reg <= d_in;
                        state <= S3;
                    end

            // Выполнение операции деления
            S3: begin
                    if (b_reg == 0)
                        begin
                            error_out <= DIV_BY_ZERO;
                            valid_out <= 1;
                        end
                    else if (a_reg == 0)
                        begin
                            d_out <= 0;
                            valid_out <= 1;
                        end
                    else
                        begin
                            d_out <= a_reg / b_reg;
                            valid_out <= 1;
                        end

                    state <= S0;
                end
        endcase
end

endmodule
```

Листинг 2. Файл "create.tcl"

```
cd [file dirname [info script]]
create_project div_project div_project -part xc7a100tcsg324-1
import_files -norecurse div_project/fsm_div.v
update_compile_order -fileset sources_1
file mkdir div_project/div_project.srcs/constrs_1
add_files -fileset constrs_1 -norecurse ccs.xdc
import_files -fileset constrs_1 ccs.xdc
launch_runs impl_1 -to_step write_bitstream -jobs 16
```

Листинг 3. Файл "ccs.xdc"

```
set_property -dict { PACKAGE_PIN E3    IOSTANDARD LVCMOS33 }
[get_ports { clk }]
create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5}
[get_ports { clk }]
```

```
set_property -dict { PACKAGE_PIN C12   IOSTANDARD LVCMOS33 }
[get_ports { reset }]
set_property -dict { PACKAGE_PIN N17   IOSTANDARD LVCMOS33 }
[get_ports { valid_in }]

set_property -dict { PACKAGE_PIN J15   IOSTANDARD LVCMOS33 }
[get_ports { d_in[0] }]
set_property -dict { PACKAGE_PIN L16   IOSTANDARD LVCMOS33 }
[get_ports { d_in[1] }]
set_property -dict { PACKAGE_PIN M13   IOSTANDARD LVCMOS33 }
[get_ports { d_in[2] }]
set_property -dict { PACKAGE_PIN R15   IOSTANDARD LVCMOS33 }
[get_ports { d_in[3] }]

set_property -dict { PACKAGE_PIN R17   IOSTANDARD LVCMOS33 }
[get_ports { d_out[0] }]
set_property -dict { PACKAGE_PIN T18   IOSTANDARD LVCMOS33 }
[get_ports { d_out[1] }]
set_property -dict { PACKAGE_PIN U18   IOSTANDARD LVCMOS33 }
[get_ports { d_out[2] }]
set_property -dict { PACKAGE_PIN R13   IOSTANDARD LVCMOS33 }
[get_ports { d_out[3] }]

set_property -dict { PACKAGE_PIN H17   IOSTANDARD LVCMOS33 }
[get_ports { valid_out }]
set_property -dict { PACKAGE_PIN K15   IOSTANDARD LVCMOS33 }
[get_ports { error_out }]
```