Э.001.01. «Конечный автомат для обработки двух 8-битных входных сигналов». Текст программы. Приложение.

Листинг 1. Файл " fsm_8bit_shift.v"

```
module fsm_8bit_shift (
    input clk,
    input reset,
    input data_in1_valid,
    input data_in2_valid,
    input [7:0] data_in1,
    input [7:0] data_in2,
    output reg [15:0] data_out,
    output reg output_valid
);

    reg [1:0] state, next_state;

    localparam IDLE = 2'b00, SHIFT = 2'b01, COMBINE = 2'b10;

    reg [7:0] data_temp1, data_temp2;

    always @(posedge clk or posedge reset) begin
        if (reset) begin
            state <= IDLE;
            data_out <= 16'b0;
            output_valid <= 0;
        end else begin
            state <= next_state;
            case (state)
                IDLE: begin
                    output_valid <= 0;
                    if (data_in1_valid && data_in2_valid) begin
                        data_temp1 <= data_in1;
                        data_temp2 <= data_in2;
                    end
                end
                SHIFT: begin
                    data_out <= data_temp1 << 2;
                end
                COMBINE: begin
                    data_out <= {data_temp1, data_temp2};
                    output_valid <= 1;
                end
            endcase
        end
    end

    always @(*) begin
        case (state)
            IDLE: begin
```

Продолжение Листинг 1

```
                if (data_in1_valid && data_in2_valid)
                    next_state = SHIFT;
                else
                    next_state = IDLE;
            end
            SHIFT: next_state = COMBINE;
            COMBINE: next_state = IDLE;
            default: next_state = IDLE;
        endcase
    end
endmodule
```

Листинг 2. Файл "create.tcl"

```
cd [file dirname [info script]]
create_project fsm_8bit_shift_project fsm_8bit_shift_project -part
xc7a100tcsg324-1
import_files -norecurse fsm_8bit_shift.v
update_compile_order -fileset sources_1
file mkdir fsm_8bit_shift_project/
fsm_8bit_shift_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 P17 IOSTANDARD LVCMOS33 } [get_ports
{ data_in1_valid }];
set_property -dict { PACKAGE_PIN M17 IOSTANDARD LVCMOS33 } [get_ports
{ data_in2_valid }];

set_property -dict { PACKAGE_PIN R11 IOSTANDARD LVCMOS33 } [get_ports
{ output_valid }];

set_property -dict { PACKAGE_PIN J15 IOSTANDARD LVCMOS33 } [get_ports
{ data_in1[0] }];
set_property -dict { PACKAGE_PIN L16 IOSTANDARD LVCMOS33 } [get_ports
{ data_in1[1] }];
set_property -dict { PACKAGE_PIN M13 IOSTANDARD LVCMOS33 } [get_ports
```

```
{ data_in1[2] }];
set_property -dict { PACKAGE_PIN R15 IOSTANDARD LVCMOS33 } [get_ports
{ data_in1[3] }];
set_property -dict { PACKAGE_PIN R17 IOSTANDARD LVCMOS33 } [get_ports
{ data_in1[4] }];
set_property -dict { PACKAGE_PIN T18 IOSTANDARD LVCMOS33 } [get_ports
{ data_in1[5] }];
set_property -dict { PACKAGE_PIN U18 IOSTANDARD LVCMOS33 } [get_ports
{ data_in1[6] }];
set_property -dict { PACKAGE_PIN R13 IOSTANDARD LVCMOS33 } [get_ports
{ data_in1[7] }];

set_property -dict { PACKAGE_PIN T8 IOSTANDARD LVCMOS18 } [get_ports {
data_in2[0] }];
set_property -dict { PACKAGE_PIN U8 IOSTANDARD LVCMOS18 } [get_ports {
data_in2[1] }];
set_property -dict { PACKAGE_PIN R16 IOSTANDARD LVCMOS33 } [get_ports
{ data_in2[2] }];
set_property -dict { PACKAGE_PIN T13 IOSTANDARD LVCMOS33 } [get_ports
{ data_in2[3] }];
set_property -dict { PACKAGE_PIN H6 IOSTANDARD LVCMOS33 } [get_ports {
data_in2[4] }];
set_property -dict { PACKAGE_PIN U12 IOSTANDARD LVCMOS33 } [get_ports
{ data_in2[5] }];
set_property -dict { PACKAGE_PIN U11 IOSTANDARD LVCMOS33 } [get_ports
{ data_in2[6] }];
set_property -dict { PACKAGE_PIN V10 IOSTANDARD LVCMOS33 } [get_ports
{ data_in2[7] }];

set_property -dict { PACKAGE_PIN H17 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[0] }];
set_property -dict { PACKAGE_PIN K15 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[1] }];
set_property -dict { PACKAGE_PIN J13 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[2] }];
set_property -dict { PACKAGE_PIN N14 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[3] }];
set_property -dict { PACKAGE_PIN R18 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[4] }];
set_property -dict { PACKAGE_PIN V17 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[5] }];
set_property -dict { PACKAGE_PIN U17 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[6] }];
set_property -dict { PACKAGE_PIN U16 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[7] }];
set_property -dict { PACKAGE_PIN V16 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[8] }];
set_property -dict { PACKAGE_PIN T15 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[9] }];
set_property -dict { PACKAGE_PIN U14 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[10] }];
set_property -dict { PACKAGE_PIN T16 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[11] }];
set_property -dict { PACKAGE_PIN V15 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[12] }];
set_property -dict { PACKAGE_PIN V14 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[13] }];
```

```
set_property -dict { PACKAGE_PIN V12 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[14] }];
set_property -dict { PACKAGE_PIN V11 IOSTANDARD LVCMOS33 } [get_ports
{ data_out[15] }];
```