PES UNIVERSITY

CELEBRATING 50 YEARS

DIGITAL SYSTEM DESIGN

TOPIC – WASHING MACHINE CONTROLLER

SEMESTER - 5

TEAM MEMBERS

P N UTTUNGA - PES1UG21EC175

PRAJWAL P KULKARNI - PES1UG21EC186

PRANAV J P - PES1UG21EC189

GUIDE : - Prof. Anuros Thomas K

Project Report:

Washing Machine Controller

Introduction

The **Washing Machine Controller** project is a digital system design undertaken during the fifth semester as part of the Digital System Design course. The primary objective of this project is to develop a washing machine controller using a Finite State Machine (FSM) block and a Timer block. The controller supports multiple washing modes, each with distinct time durations for different states in the washing cycle.

Project Overview

Project Structure

The project is structured into two key components:

1. **Finite State Machine (FSM) Block:** Responsible for managing state transitions and receiving signals from both the user and the timer.
2. **Timer Block:** Generates the required time periods for each washing cycle. It includes an up-counter and combinational logic to produce accurate timing signals based on count values. The timer values are determined by the clock frequency used in the system.

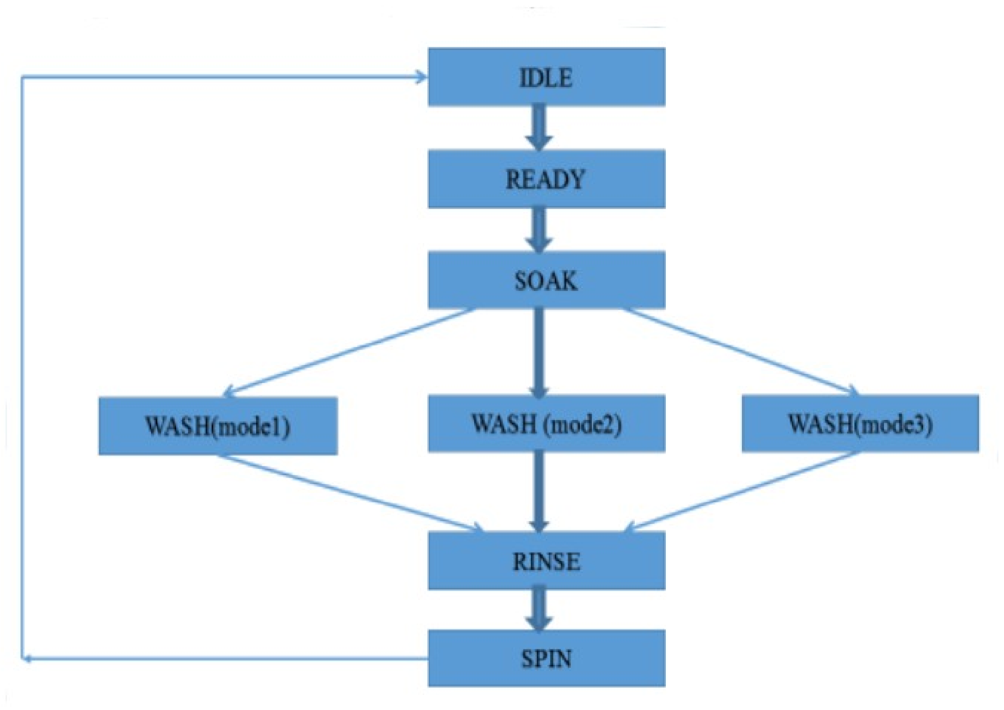
Functionalities

The washing machine controller offers the following functionalities:

1. **States:** The washing machine has the following states: idle, soak, wash, rinse, and spin.
2. **Modes of Operation:** Three modes are available: mode1, mode2, and mode3.
3. **Timing:** Different time durations are allocated to each mode of operation.

System Flow Chart

The washing machine controller operates based on the interaction between the FSM and Timer blocks. The FSM handles state transitions, while the Timer block ensures precise timing for each state in the washing cycle. The user interacts with the system by inserting a coin and selecting a washing mode.



Finite State Machine (FSM) Block

The FSM block manages the following states:

- **IDLE:** Initial state.
- **READY:** Transitioned to after a coin is inserted.
- **WASHING STATES:** Soak, wash, rinse, spin, based on the selected washing mode.

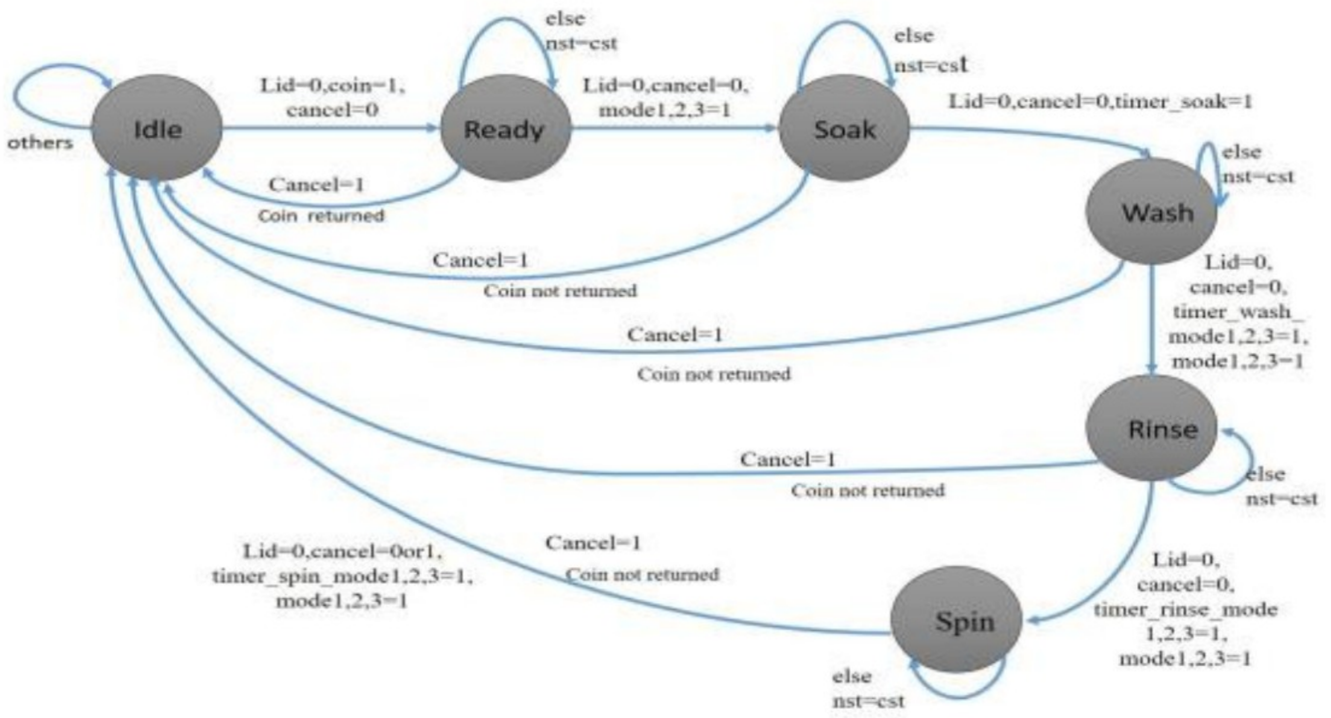


Figure 2: State Diagram of Washing Machine Control System

Controller Operation

The operation of the washing machine controller follows a structured sequence:

1. **IDLE State:** The FSM is initially in the IDLE state.
2. **READY State:** After a coin is inserted, the FSM transitions to the READY state.
3. **Mode Selection:** Users in the READY state can choose one of the three washing modes: mode1, mode2, or mode3, determining the timing for each state.
4. **Cancellation Handling:**
    - If no mode is selected, the controller remains in the READY state.
    - If the washing process is canceled at this point, the coin is returned, and the FSM returns to the IDLE state.
5. **Washing Process:** Once the washing process starts, canceling it results in a loss of the coin, and the washing machine proceeds with the current cycle.

Design Code:-

```
module washing_machine_controller (
    input clk, // System clock
    input rst, // System reset
    input start, // Start signal to begin the washing process
    input cancel, // Cancel signal to stop the washing process
    input lid_open, // Signal indicating whether the lid is open
    input mode1, // Mode 1 selection signal
    input mode2, // Mode 2 selection signal
    input mode3, // Mode 3 selection signal
    output reg water_inlet, // Water intake control signal
    output reg idle_op, // Idle state signal
    output reg ready_op, // Ready state signal
    output reg soak_op, // Soaking operation signal
    output reg wash_op, // Washing operation signal
    output reg rinse_op, // Rinsing operation signal
    output reg spin_op, // Spinning operation signal
    output reg coin_rtrn // Coin return signal
);

// Declare the states as parameters
parameter IDLE = 3'b000;
parameter READY = 3'b001;
parameter SOAK = 3'b010;
parameter WASH = 3'b011;
parameter RINSE = 3'b100;
parameter SPIN = 3'b101;

// Declare the state register
reg [2:0] state, next_state;

// Default values
always @ (posedge clk or posedge rst) begin
    if (rst) begin
        state <= IDLE;
    end else begin
        state <= next_state;
    end
end

// Next state logic
always @ (*) begin
    case (state)
        IDLE: begin
            if (start) begin
                next_state = READY;
            end else begin
                next_state = IDLE;
            end
        end
        READY: begin
            if (cancel) begin
                coin_rtrn = 1'b1;
                next_state = IDLE;
            end else if (mode1) begin
                // Set timing for mode 1
                next_state = SOAK;
            end else if (mode2) begin
                // Set timing for mode 2
                next_state = WASH;
            end else if (mode3) begin
                // Set timing for mode 3
                next_state = RINSE;
            end else begin
                next_state = READY;
            end
        end
        SOAK: begin
            // Add logic for SOAK state
            if (lid_open) begin
                // Pause the washing process
                next_state = IDLE;
            end else begin
                // Continue with the current state
                next_state = WASH; // Transition to the next state
            end
        end
        WASH: begin
            // Add logic for WASH state
            // Transition to RINSE state
            next_state = RINSE;
        end
        RINSE: begin
            // Add logic for RINSE state
            // Transition to SPIN state
            next_state = SPIN;
        end
        SPIN: begin
            // Add logic for SPIN state
            // Transition back to IDLE state
            next_state = IDLE;
        end
        default: next_state = IDLE;
    endcase
end

// Output logic
```

```
always @ (state) begin
    case (state)
        IDLE: begin
            idle_op = 1'b1;
            ready_op = 1'b0;
            soak_op = 1'b0;
            wash_op = 1'b0;
            rinse_op = 1'b0;
            spin_op = 1'b0;
            water_inlet = 1'b0;
            coin_rtrn = 1'b0;
        end
        READY: begin
            idle_op = 1'b0;
            ready_op = 1'b1;
            soak_op = 1'b0;
            wash_op = 1'b0;
            rinse_op = 1'b0;
            spin_op = 1'b0;
            water_inlet = 1'b0;
            coin_rtrn = 1'b0;
        end
        SOAK: begin
            idle_op = 1'b0;
            ready_op = 1'b0;
            soak_op = 1'b1;
            wash_op = 1'b0;
            rinse_op = 1'b0;
            spin_op = 1'b0;
            water_inlet = 1'b1; // Activate water inlet during soak state
            coin_rtrn = 1'b0;
        end
        WASH: begin
            idle_op = 1'b0;
            ready_op = 1'b0;
            soak_op = 1'b0;
            wash_op = 1'b1;
            rinse_op = 1'b0;
            spin_op = 1'b0;
            water_inlet = 1'b0;
            coin_rtrn = 1'b0;
        end
        RINSE: begin
            idle_op = 1'b0;
            ready_op = 1'b0;
            soak_op = 1'b0;
            wash_op = 1'b0;
            rinse_op = 1'b1;
            spin_op = 1'b0;
            water_inlet = 1'b1; // Activate water inlet during rinse state
            coin_rtrn = 1'b0;
        end
        SPIN: begin
            idle_op = 1'b0;
            ready_op = 1'b0;
            soak_op = 1'b0;
            wash_op = 1'b0;
            rinse_op = 1'b0;
            spin_op = 1'b1;
            water_inlet = 1'b0;
            coin_rtrn = 1'b0;
        end
        default: begin
            idle_op = 1'b0;
            ready_op = 1'b0;
            soak_op = 1'b0;
            wash_op = 1'b0;
            rinse_op = 1'b0;
            spin_op = 1'b0;
            water_inlet = 1'b0;
            coin_rtrn = 1'b0;
        end
    endcase
end
endmodule
```

Testbench code:-

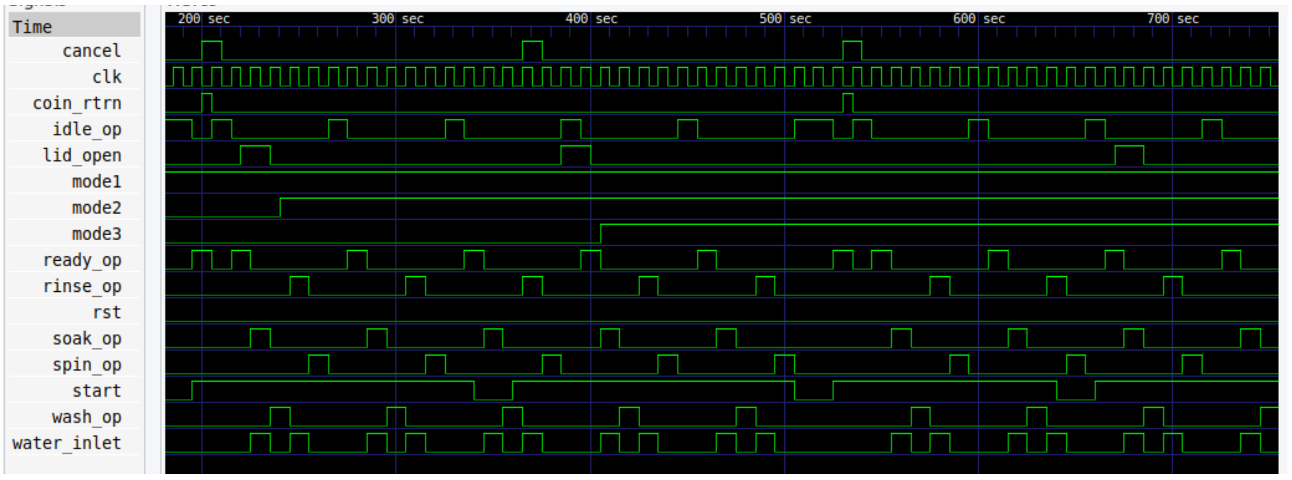
```
module washing_machine_tb;


 // Parameters
 reg clk;
 reg rst;
 reg start;
 reg cancel;
 reg lid_open;
 reg mode1;
 reg mode2;
 reg mode3;
 wire water_inlet;
 wire idle_op;
 wire ready_op;
 wire soak_op;
 wire wash_op;
 wire rinse_op;
 wire spin_op;
 wire coin_rtrn;


 // Instantiate the washing machine controller module
 washing_machine_controller dut (
   .clk(clk),
   .rst(rst),
   .start(start),
   .cancel(cancel),
   .lid_open(lid_open),
   .mode1(mode1),
   .mode2(mode2),
   .mode3(mode3),
   .water_inlet(water_inlet),
   .idle_op(idle_op),
   .ready_op(ready_op),
   .soak_op(soak_op),
   .wash_op(wash_op),
   .rinse_op(rinse_op),
   .spin_op(spin_op),
   .coin_rtrn(coin_rtrn)
 );


 // Clock generation
 always #5 clk = ~clk;


 // Initial values
 initial begin
 $dumpfile("dump.vcd");
 $dumpvars(0,washing_machine_tb);
```

```
clk = 0;

rst = 1;

start = 0;

cancel = 0;

lid_open = 0;

mode1 = 0;

mode2 = 0;

mode3 = 0;


// Reset the system

#10 rst = 0;


// Test a complete wash cycle with mode1

#20 start = 1;

#5 cancel = 1;

#10 cancel = 0;

#10 lid_open = 1;

#15 lid_open = 0;

#5 mode1 = 1;

#100 start = 0;


// Test a complete wash cycle with mode2

#20 start = 1;

#5 cancel = 1;

#10 cancel = 0;

#10 lid_open = 1;

#15 lid_open = 0;

#5 mode2 = 1;

#100 start = 0;


// Test a complete wash cycle with mode3

#20 start = 1;

#5 cancel = 1;

#10 cancel = 0;

#10 lid_open = 1;

#15 lid_open = 0;

#5 mode3 = 1;

#100 start = 0;


// Test cancellation during different states

#20 start = 1;

#5 cancel = 1;

#10 cancel = 0;

#100 start = 0;


// Test lid open during different states

#20 start = 1;

#10 lid_open = 1;

#15 lid_open = 0;
```

```
    #100 start = 0;


    #150 $finish; // End the simulation after 150 time units

  end


  // Display outputs

  always @(posedge clk) begin

    $display("Time = %0t, Water Inlet = %b, Idle Op = %b, Ready Op = %b, Soak Op = %b, Wash Op = %b, Rinse Op = %b, Spin Op = %b, Coin Return = %b", $time, water_inlet, idle_op, ready_op, soak_op, wash_op, rinse_op, spin_op, coin_rtrn);

  end


endmodule
```

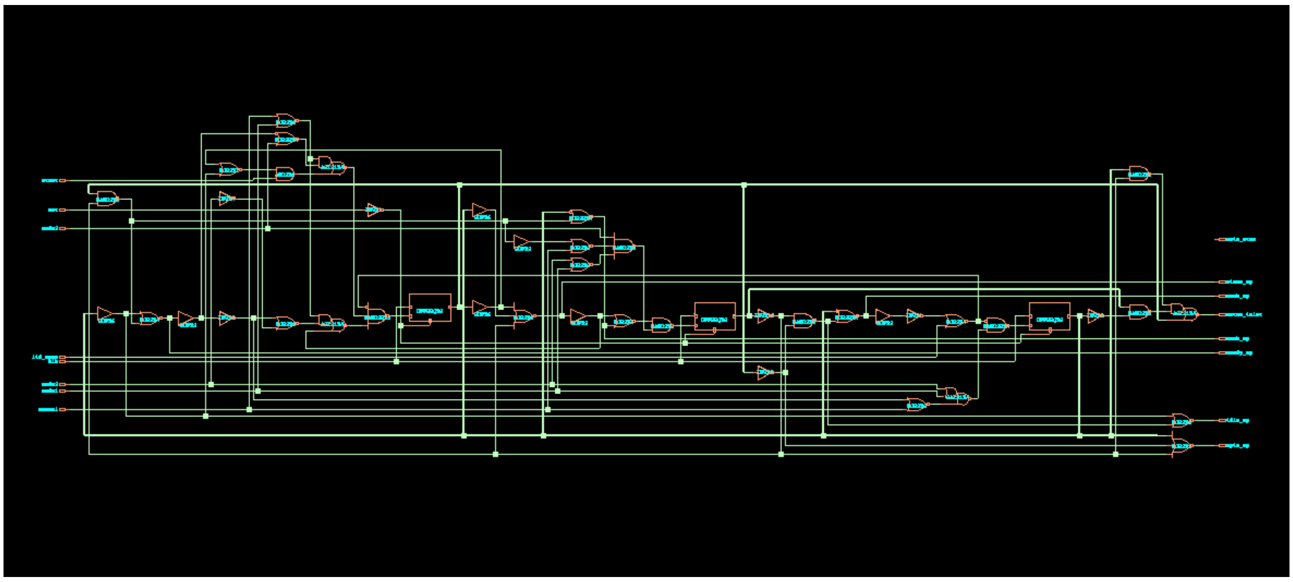
Simulation Waveform

The simulation of the washing machine controller was performed using the ncverilog simulator. The waveform generated provides insights into the system's behavior and interactions between different components



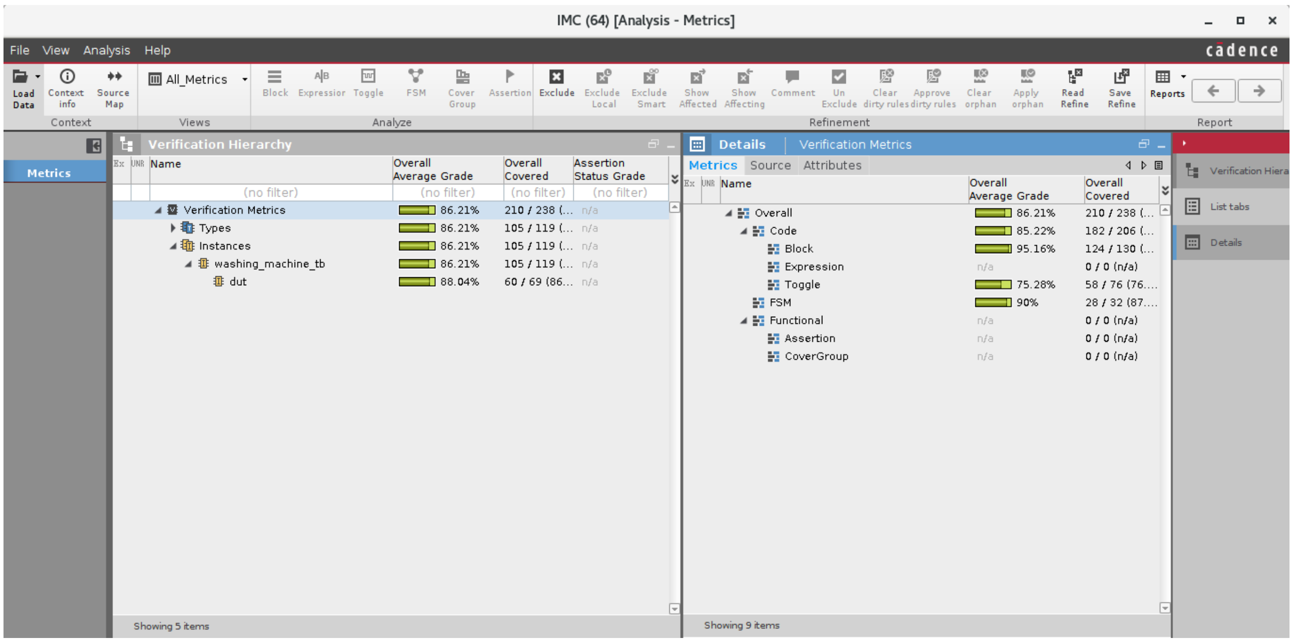
Synthesis Overview

The synthesis phase, conducted using the Genus tool from the Cadence Tool Suite, involved the translation of the high-level washing machine controller design into a gate-level representation. This process ensures that the digital logic is optimized for implementation on hardware. The successful synthesis is evident from the generated schematic, showcasing the efficient transformation of the design into synthesizable logic elements.



Coverage Analysis Summary

Coverage analysis, performed using the IMC tool from the Cadence Tool Suite, assesses the effectiveness of the test cases in exercising different parts of the washing machine controller. The coverage results indicate the comprehensiveness of the testing strategy, ensuring that critical functional aspects are thoroughly validated.


IMC (64) [Analysis - Metrics]

Verification Hierarchy

Ex	UNR	Name	Overall Average Grade	Overall Covered	Assertion Status Grade
		(no filter)	(no filter)	(no filter)	(no filter)
		Verification Metrics	86.21%	210 / 238 (…	n/a
		Types	86.21%	105 / 119 (…	n/a
		Instances	86.21%	105 / 119 (…	n/a
		washing_machine_tb	86.21%	105 / 119 (…	n/a
		dut	88.04%	60 / 69 (86…	n/a

Showing 5 items

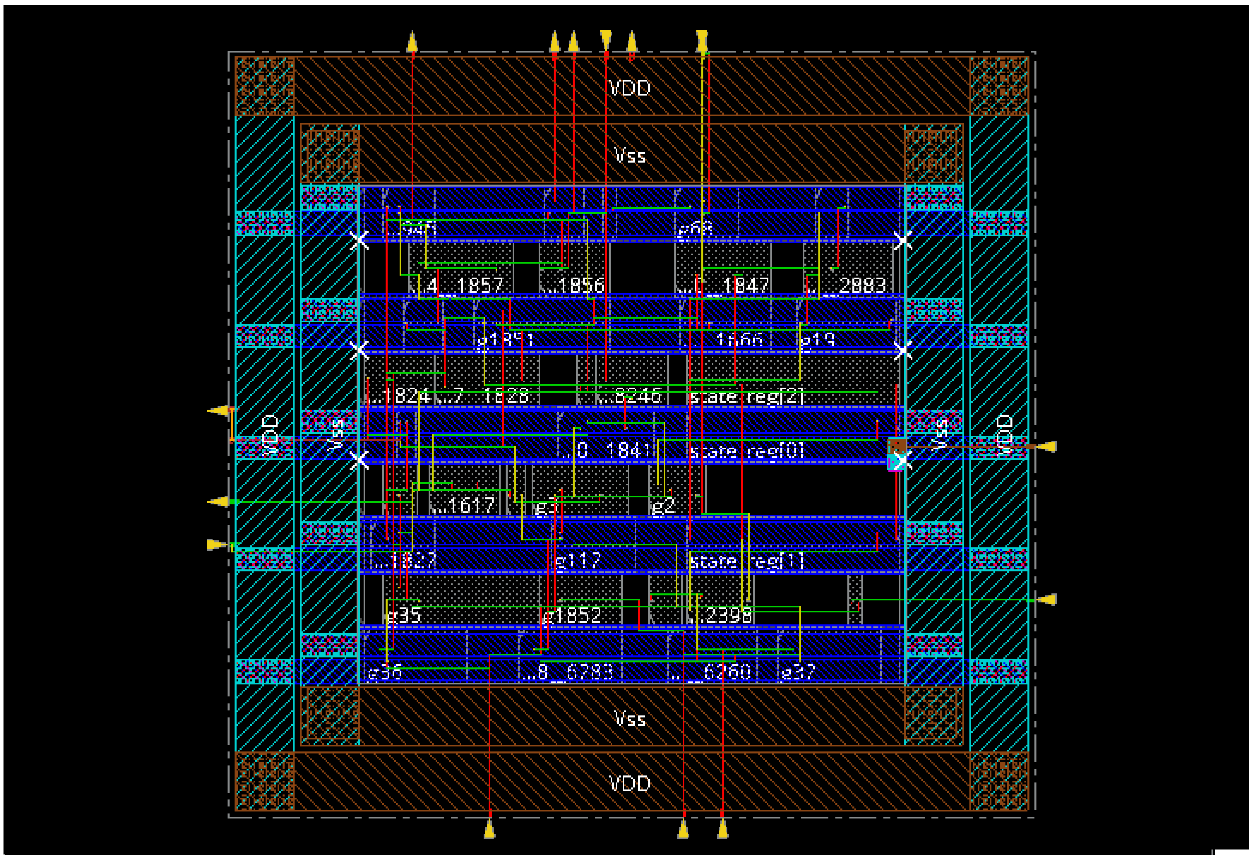
Details | Verification Metrics

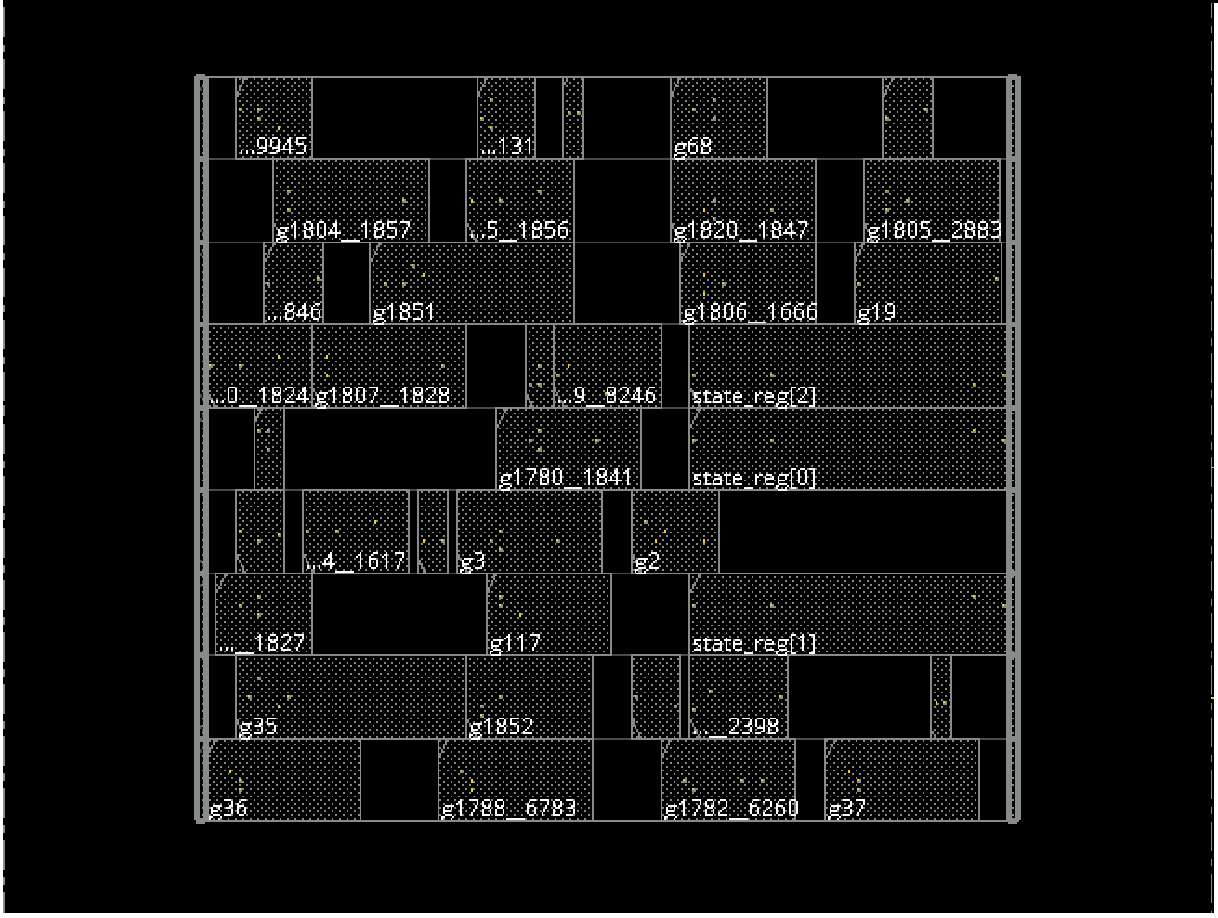
Ex	UNR	Name	Overall Average Grade	Overall Covered
		Overall	86.21%	210 / 238 (…
		Code	85.22%	182 / 206 (…
		Block	95.16%	124 / 130 (…
		Expression	n/a	0 / 0 (n/a)
		Toggle	75.28%	58 / 76 (76.…
		FSM	90%	28 / 32 (87.…
		Functional	n/a	0 / 0 (n/a)
		Assertion	n/a	0 / 0 (n/a)
		CoverGroup	n/a	0 / 0 (n/a)

Showing 9 items

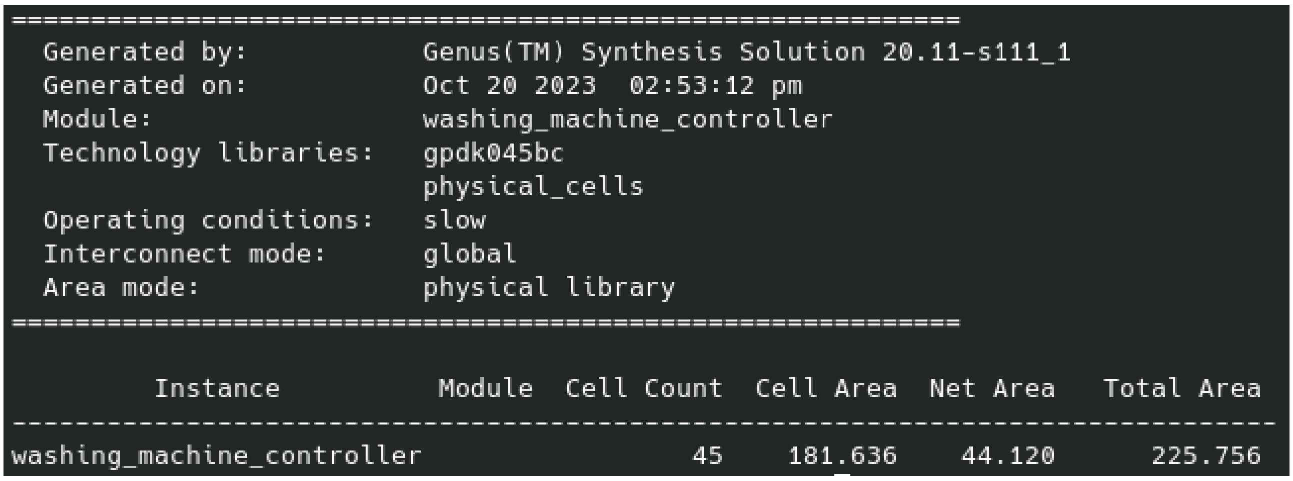
Physical Design Insights

The physical design phase, executed with the Innovus tool from the Cadence Tool Suite, involves translating the synthesized logic into a physical layout suitable for fabrication. The screenshots reveal the floorplan and placement of cells, highlighting the careful arrangement of components to meet performance and area requirements.



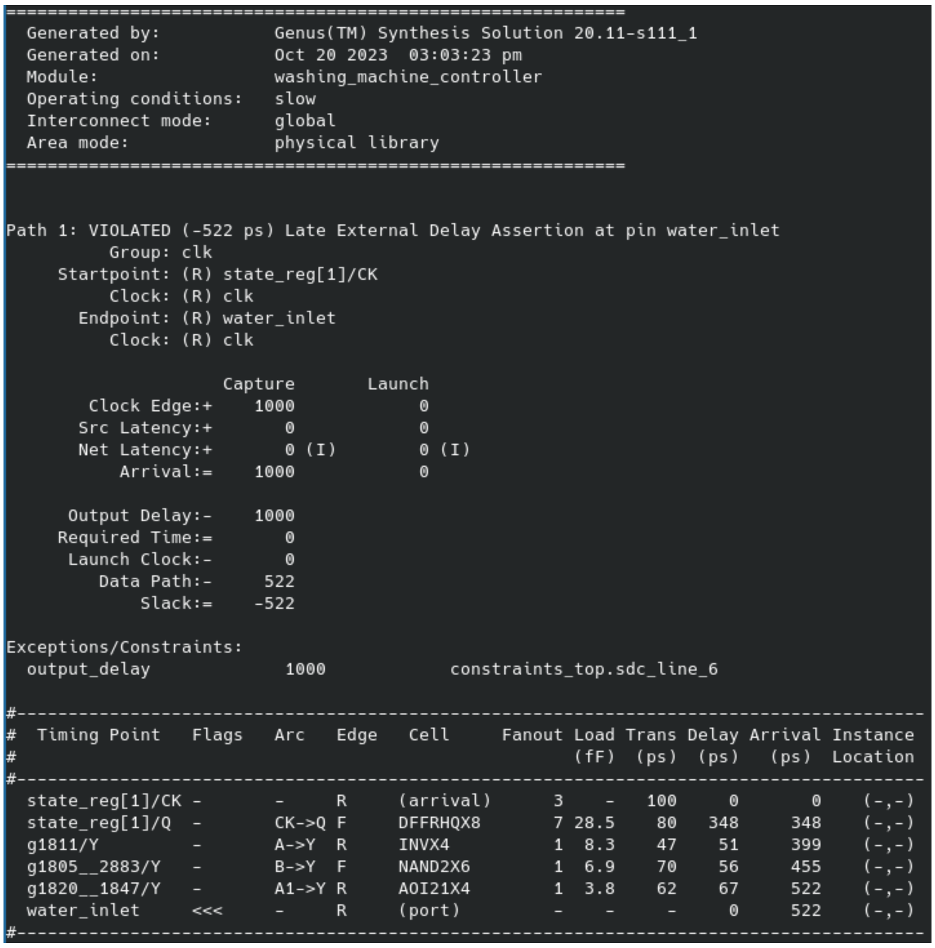


Area Report



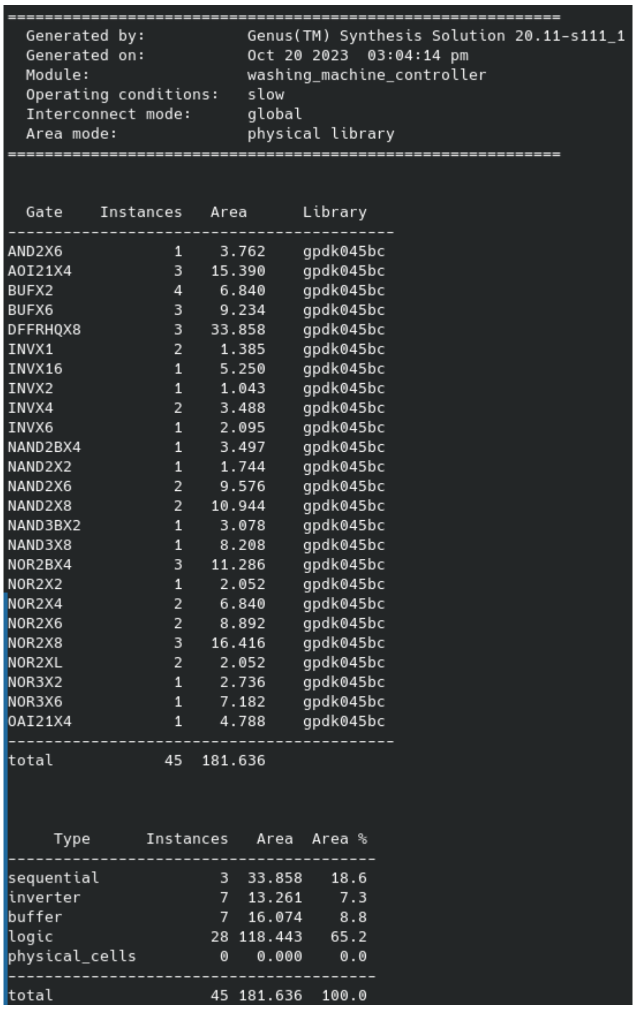
```
============================================================
  Generated by:           Genus(TM) Synthesis Solution 20.11-s111_1
  Generated on:           Oct 20 2023  02:53:12 pm
  Module:                 washing_machine_controller
  Technology libraries:   gpdk045bc
                          physical_cells
  Operating conditions:   slow
  Interconnect mode:      global
  Area mode:              physical library
============================================================

        Instance          Module  Cell Count  Cell Area  Net Area   Total Area
--------------------------------------------------------------------------------
washing_machine_controller                45    181.636    44.120      225.756
```

Timing Report



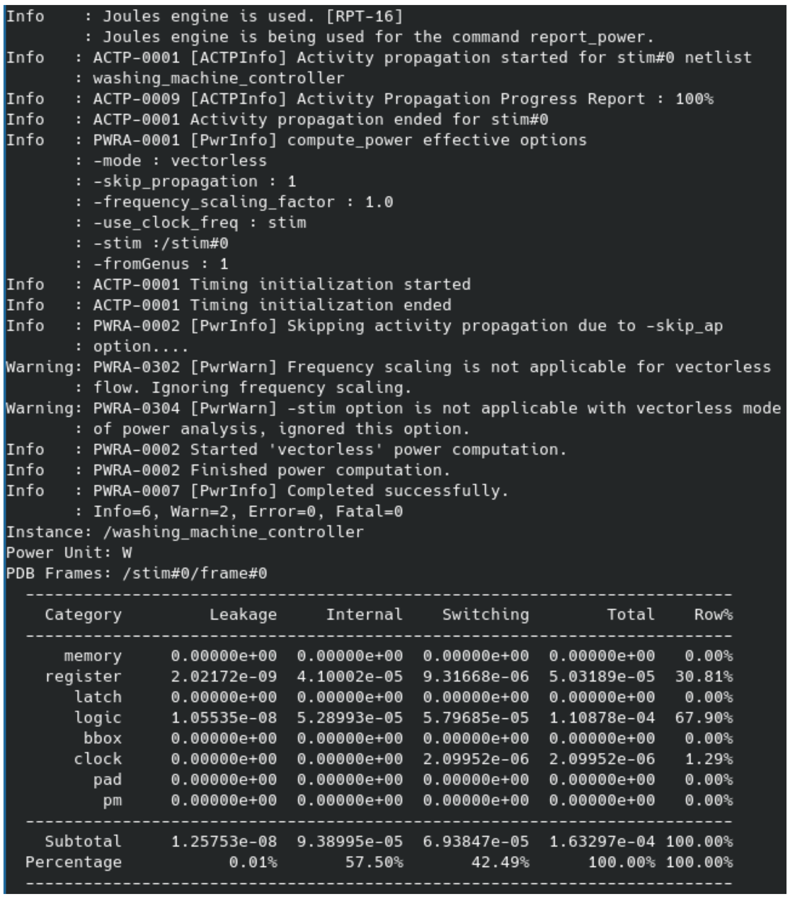
```
============================================================
  Generated by:           Genus(TM) Synthesis Solution 20.11-s111_1
  Generated on:           Oct 20 2023  03:03:23 pm
  Module:                 washing_machine_controller
  Operating conditions:   slow
  Interconnect mode:      global
  Area mode:              physical library
============================================================


Path 1: VIOLATED (-522 ps) Late External Delay Assertion at pin water_inlet
          Group: clk
     Startpoint: (R) state_reg[1]/CK
          Clock: (R) clk
       Endpoint: (R) water_inlet
          Clock: (R) clk

                     Capture       Launch
        Clock Edge:+    1000            0
       Src Latency:+       0            0
       Net Latency:+       0 (I)        0 (I)
           Arrival:=    1000            0

      Output Delay:-    1000
     Required Time:=       0
      Launch Clock:-       0
         Data Path:-     522
             Slack:=    -522

Exceptions/Constraints:
  output_delay             1000            constraints_top.sdc_line_6

#-----------------------------------------------------------------------------------------
#  Timing Point   Flags   Arc   Edge   Cell     Fanout Load Trans Delay Arrival Instance
#                                                       (fF)  (ps)  (ps)   (ps)  Location
#-----------------------------------------------------------------------------------------
  state_reg[1]/CK -       -     R     (arrival)      3    -   100     0       0    (-,-)
  state_reg[1]/Q  -       CK->Q F     DFFRHQX8       7 28.5    80   348     348    (-,-)
  g1811/Y         -       A->Y  R     INVX4          1  8.3    47    51     399    (-,-)
  g1805__2883/Y   -       B->Y  F     NAND2X6        1  6.9    70    56     455    (-,-)
  g1820__1847/Y   -       A1->Y R     AOI21X4        1  3.8    62    67     522    (-,-)
  water_inlet     <<<     -     R     (port)         -    -     -     0     522    (-,-)
#-----------------------------------------------------------------------------------------
```

Gate Report



```
============================================================
  Generated by:           Genus(TM) Synthesis Solution 20.11-s111_1
  Generated on:           Oct 20 2023  03:04:14 pm
  Module:                 washing_machine_controller
  Operating conditions:   slow
  Interconnect mode:      global
  Area mode:              physical library
============================================================


  Gate    Instances   Area      Library
------------------------------------------
AND2X6            1    3.762    gpdk045bc
AOI21X4           3   15.390    gpdk045bc
BUFX2             4    6.840    gpdk045bc
BUFX6             3    9.234    gpdk045bc
DFFRHQX8          3   33.858    gpdk045bc
INVX1             2    1.385    gpdk045bc
INVX16            1    5.250    gpdk045bc
INVX2             1    1.043    gpdk045bc
INVX4             2    3.488    gpdk045bc
INVX6             1    2.095    gpdk045bc
NAND2BX4          1    3.497    gpdk045bc
NAND2X2           1    1.744    gpdk045bc
NAND2X6           2    9.576    gpdk045bc
NAND2X8           2   10.944    gpdk045bc
NAND3BX2          1    3.078    gpdk045bc
NAND3X8           1    8.208    gpdk045bc
NOR2BX4           3   11.286    gpdk045bc
NOR2X2            1    2.052    gpdk045bc
NOR2X4            2    6.840    gpdk045bc
NOR2X6            2    8.892    gpdk045bc
NOR2X8            3   16.416    gpdk045bc
NOR2XL            2    2.052    gpdk045bc
NOR3X2            1    2.736    gpdk045bc
NOR3X6            1    7.182    gpdk045bc
OAI21X4           1    4.788    gpdk045bc
------------------------------------------
total            45  181.636


     Type      Instances   Area  Area %
----------------------------------------
sequential            3  33.858   18.6
inverter              7  13.261    7.3
buffer                7  16.074    8.8
logic                28 118.443   65.2
physical_cells        0   0.000    0.0
----------------------------------------
total                45 181.636  100.0
```

Power Report



```
Info    : Joules engine is used. [RPT-16]
        : Joules engine is being used for the command report_power.
Info    : ACTP-0001 [ACTPInfo] Activity propagation started for stim#0 netlist
        : washing_machine_controller
Info    : ACTP-0009 [ACTPInfo] Activity Propagation Progress Report : 100%
Info    : ACTP-0001 Activity propagation ended for stim#0
Info    : PWRA-0001 [PwrInfo] compute_power effective options
        : -mode : vectorless
        : -skip_propagation : 1
        : -frequency_scaling_factor : 1.0
        : -use_clock_freq : stim
        : -stim :/stim#0
        : -fromGenus : 1
Info    : ACTP-0001 Timing initialization started
Info    : ACTP-0001 Timing initialization ended
Info    : PWRA-0002 [PwrInfo] Skipping activity propagation due to -skip_ap
        : option....
Warning: PWRA-0302 [PwrWarn] Frequency scaling is not applicable for vectorless
        : flow. Ignoring frequency scaling.
Warning: PWRA-0304 [PwrWarn] -stim option is not applicable with vectorless mode
        : of power analysis, ignored this option.
Info    : PWRA-0002 Started 'vectorless' power computation.
Info    : PWRA-0002 Finished power computation.
Info    : PWRA-0007 [PwrInfo] Completed successfully.
        : Info=6, Warn=2, Error=0, Fatal=0
Instance: /washing_machine_controller
Power Unit: W
PDB Frames: /stim#0/frame#0
  ---------------------------------------------------------------------------
    Category         Leakage     Internal    Switching        Total    Row%
  ---------------------------------------------------------------------------
      memory     0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
    register     2.02172e-09  4.10002e-05  9.31668e-06  5.03189e-05  30.81%
       latch     0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       logic     1.05535e-08  5.28993e-05  5.79685e-05  1.10878e-04  67.90%
        bbox     0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
       clock     0.00000e+00  0.00000e+00  2.09952e-06  2.09952e-06   1.29%
         pad     0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
          pm     0.00000e+00  0.00000e+00  0.00000e+00  0.00000e+00   0.00%
  ---------------------------------------------------------------------------
    Subtotal     1.25753e-08  9.38995e-05  6.93847e-05  1.63297e-04 100.00%
  Percentage           0.01%       57.50%       42.49%      100.00% 100.00%
  ---------------------------------------------------------------------------
```

Conclusion

The Washing Machine Controller project demonstrates the application of digital system design concepts, utilizing a Finite State Machine and Timer block to manage washing cycles efficiently. The project successfully integrates various tools for simulation, synthesis, coverage analysis, and physical design, showcasing a comprehensive approach to digital system development.