ECSE 325

LAB 3 REPORT

Martin Kruchinski(260915767)
Menad Kessaci(260807381)

GROUP 35

G35_mod_exp:

Introduction:

The circuit below implements a basic modular exponentiation that is going to be used in our public-key based message time stamping system. Given the flowing as inputs:

- A 14-bit signal for the exponent *d*
- A 10-bit signal representing a message *c*

The circuit calculates the following:

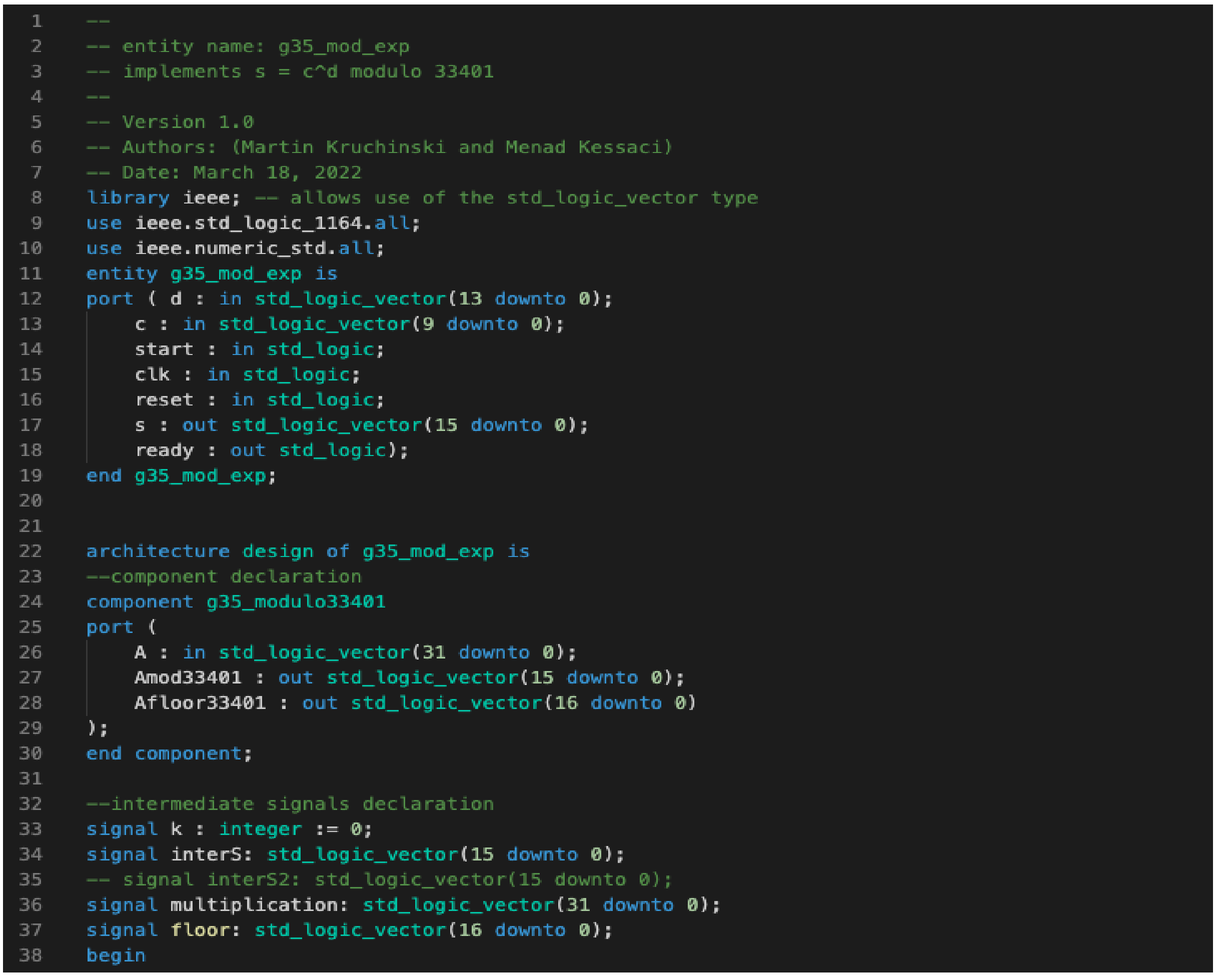
$$s = c^d \bmod n$$

(where *n* is 33401)

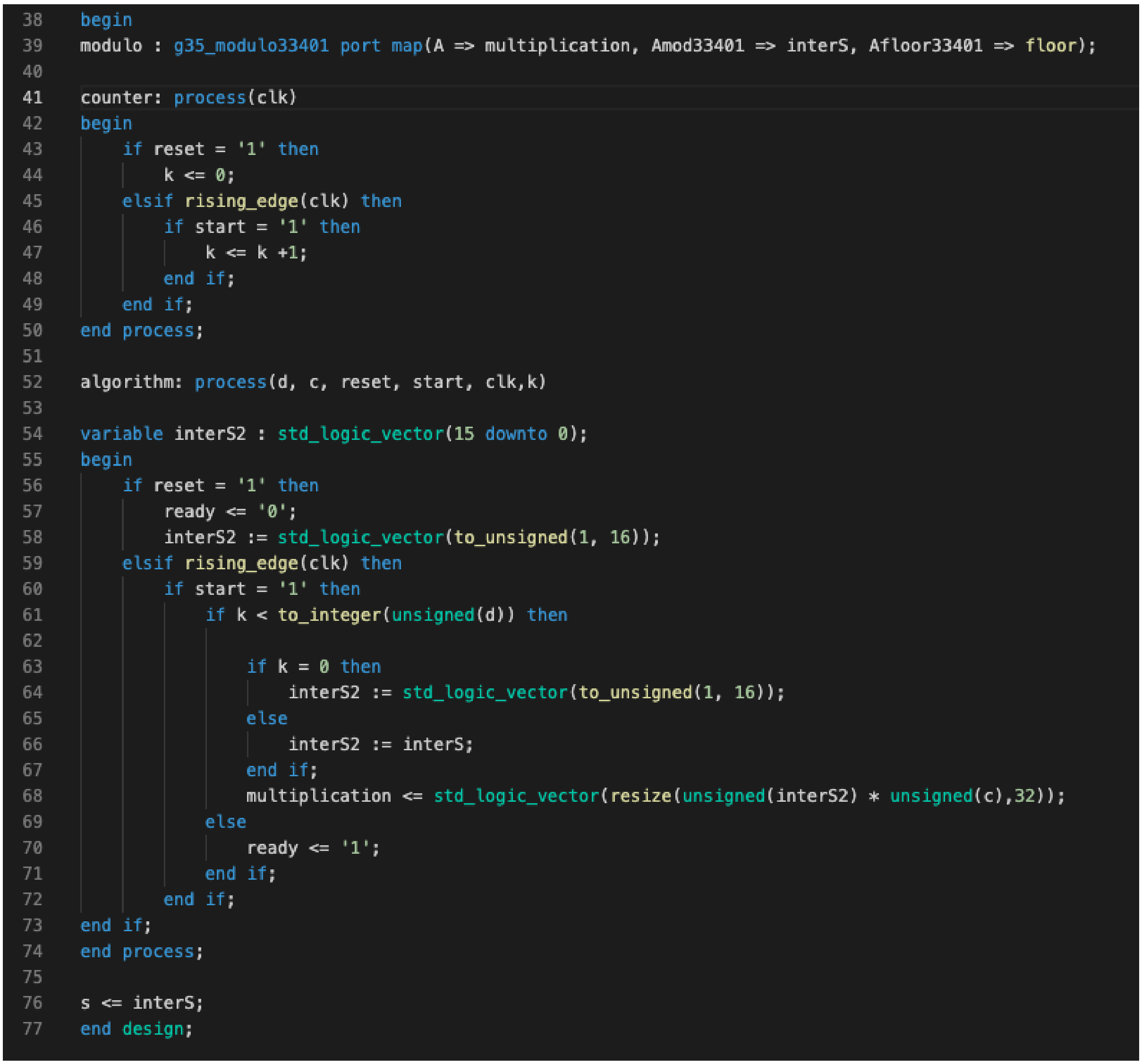
This means that the following circuit uses the circuit from Lab 2 as a component to calculate the modulus operation.

S is the 16-bit output of the circuit, which represents the result of the whole exponentiation and modulus operation.

VHDL description of the circuit:

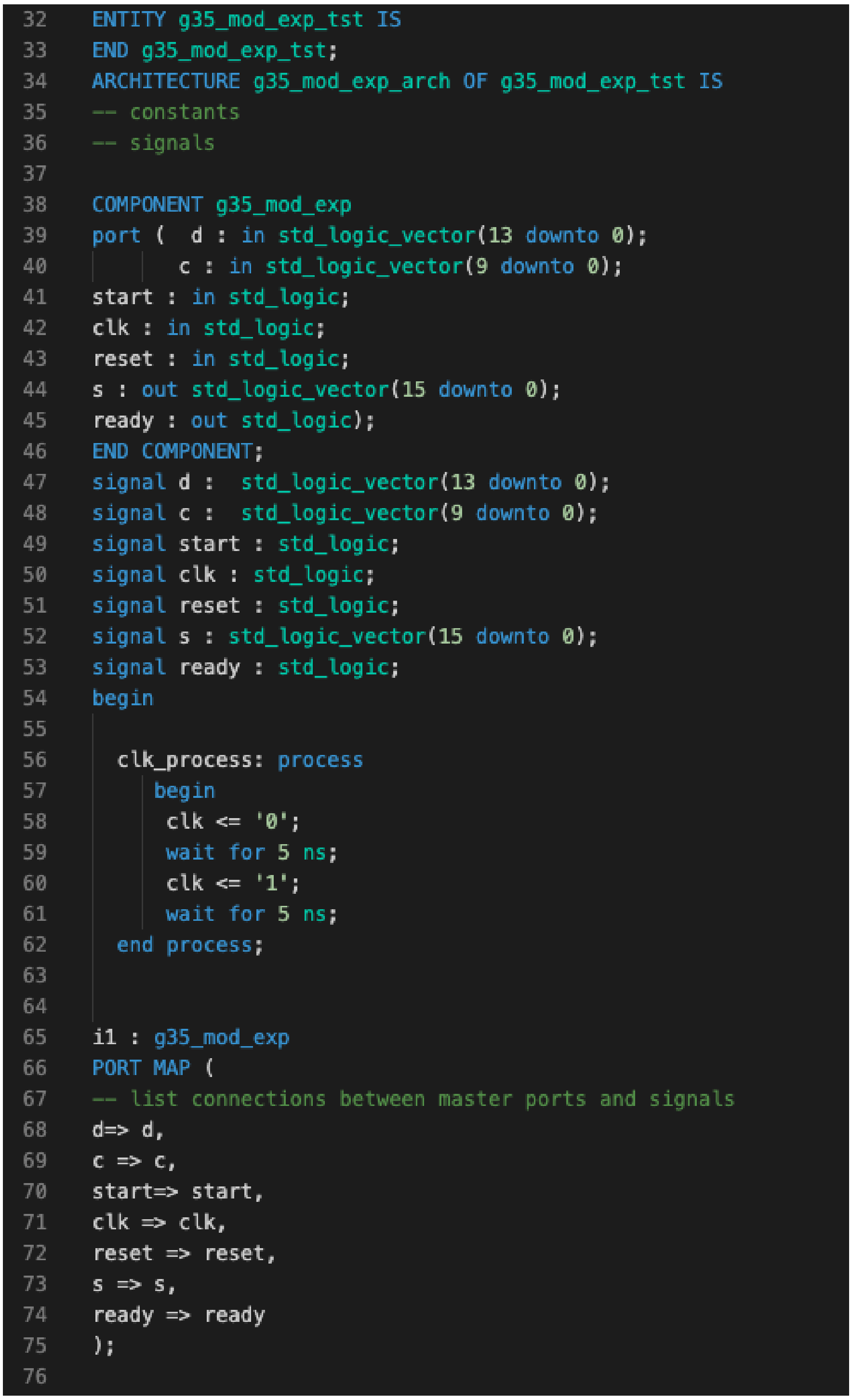


```vhdl
--
-- entity name: g35_mod_exp
-- implements s = c^d modulo 33401
--
-- Version 1.0
-- Authors: (Martin Kruchinski and Menad Kessaci)
-- Date: March 18, 2022
library ieee; -- allows use of the std_logic_vector type
use ieee.std_logic_1164.all;
use ieee.numeric_std.all;
entity g35_mod_exp is
port ( d : in std_logic_vector(13 downto 0);
    c : in std_logic_vector(9 downto 0);
    start : in std_logic;
    clk : in std_logic;
    reset : in std_logic;
    s : out std_logic_vector(15 downto 0);
    ready : out std_logic);
end g35_mod_exp;


architecture design of g35_mod_exp is
--component declaration
component g35_modulo33401
port (
    A : in std_logic_vector(31 downto 0);
    Amod33401 : out std_logic_vector(15 downto 0);
    Afloor33401 : out std_logic_vector(16 downto 0)
);
end component;

--intermediate signals declaration
signal k : integer := 0;
signal interS: std_logic_vector(15 downto 0);
-- signal interS2: std_logic_vector(15 downto 0);
signal multiplication: std_logic_vector(31 downto 0);
signal floor: std_logic_vector(16 downto 0);
begin
```

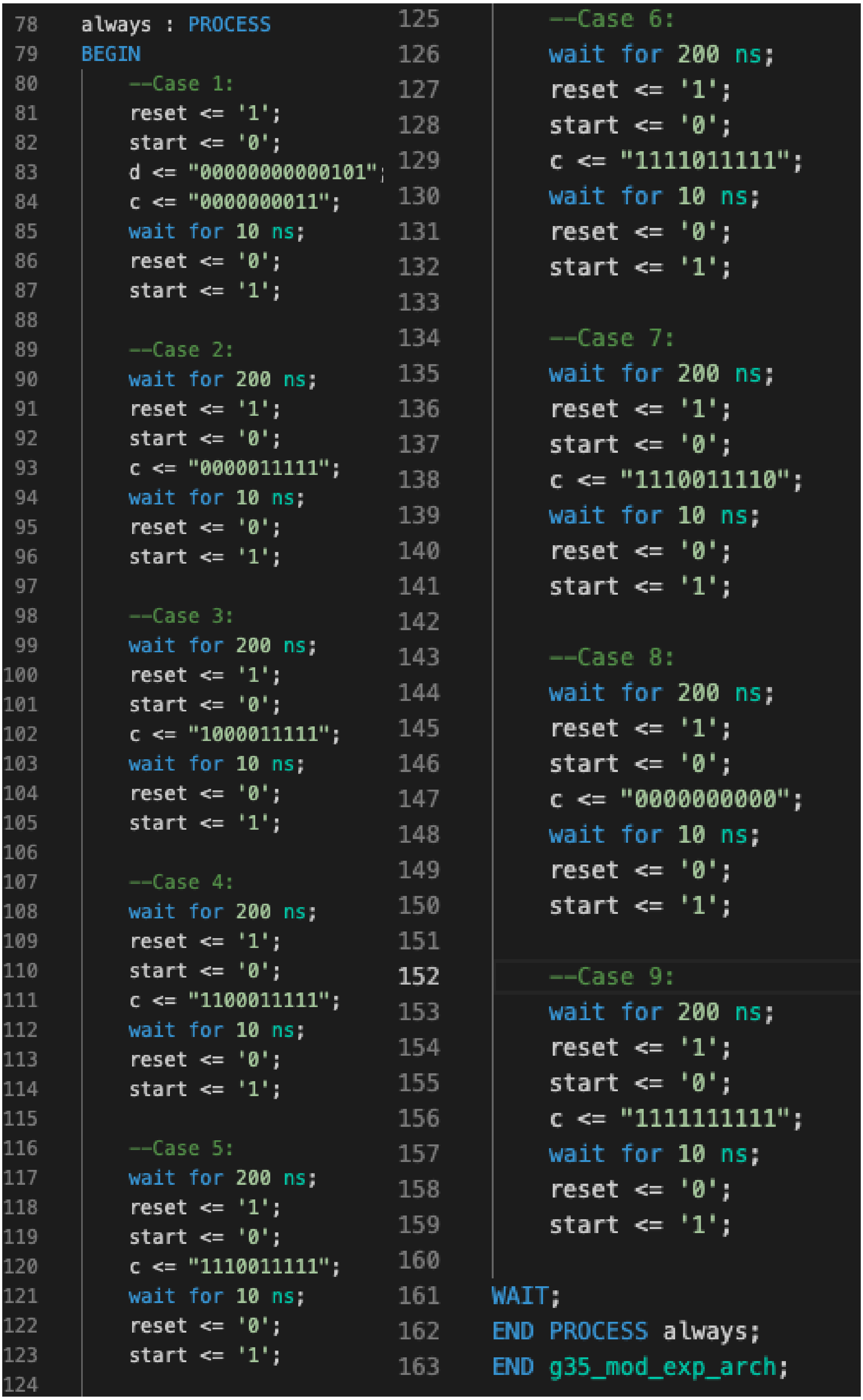


```
begin
modulo : g35_modulo33401 port map(A => multiplication, Amod33401 => interS, Afloor33401 => floor);

counter: process(clk)
begin
    if reset = '1' then
        k <= 0;
    elsif rising_edge(clk) then
        if start = '1' then
            k <= k +1;
        end if;
    end if;
end process;

algorithm: process(d, c, reset, start, clk,k)

variable interS2 : std_logic_vector(15 downto 0);
begin
    if reset = '1' then
        ready <= '0';
        interS2 := std_logic_vector(to_unsigned(1, 16));
    elsif rising_edge(clk) then
        if start = '1' then
            if k < to_integer(unsigned(d)) then

                if k = 0 then
                    interS2 := std_logic_vector(to_unsigned(1, 16));
                else
                    interS2 := interS;
                end if;
                multiplication <= std_logic_vector(resize(unsigned(interS2) * unsigned(c),32));
            else
                ready <= '1';
            end if;
        end if;
end if;
end process;

s <= interS;
end design;
```

Testbench:

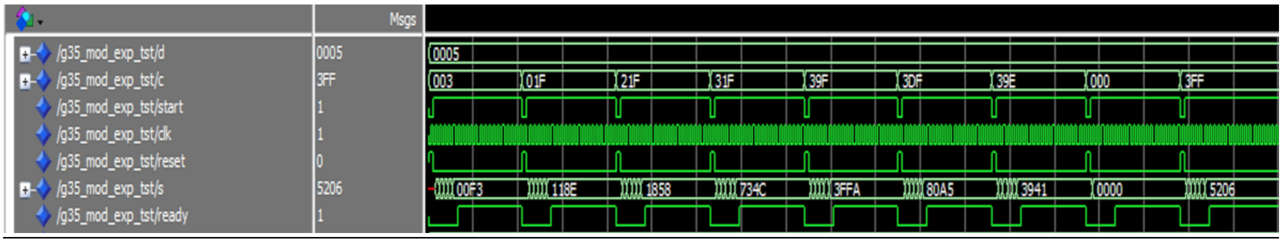


```vhdl
ENTITY g35_mod_exp_tst IS
END g35_mod_exp_tst;
ARCHITECTURE g35_mod_exp_arch OF g35_mod_exp_tst IS
-- constants
-- signals

COMPONENT g35_mod_exp
port (  d : in std_logic_vector(13 downto 0);
       c : in std_logic_vector(9 downto 0);
start : in std_logic;
clk : in std_logic;
reset : in std_logic;
s : out std_logic_vector(15 downto 0);
ready : out std_logic);
END COMPONENT;
signal d :  std_logic_vector(13 downto 0);
signal c :  std_logic_vector(9 downto 0);
signal start : std_logic;
signal clk : std_logic;
signal reset : std_logic;
signal s : std_logic_vector(15 downto 0);
signal ready : std_logic;
begin

  clk_process: process
     begin
      clk <= '0';
      wait for 5 ns;
      clk <= '1';
      wait for 5 ns;
  end process;


i1 : g35_mod_exp
PORT MAP (
-- list connections between master ports and signals
d=> d,
c => c,
start=> start,
clk => clk,
reset => reset,
s => s,
ready => ready
);

```

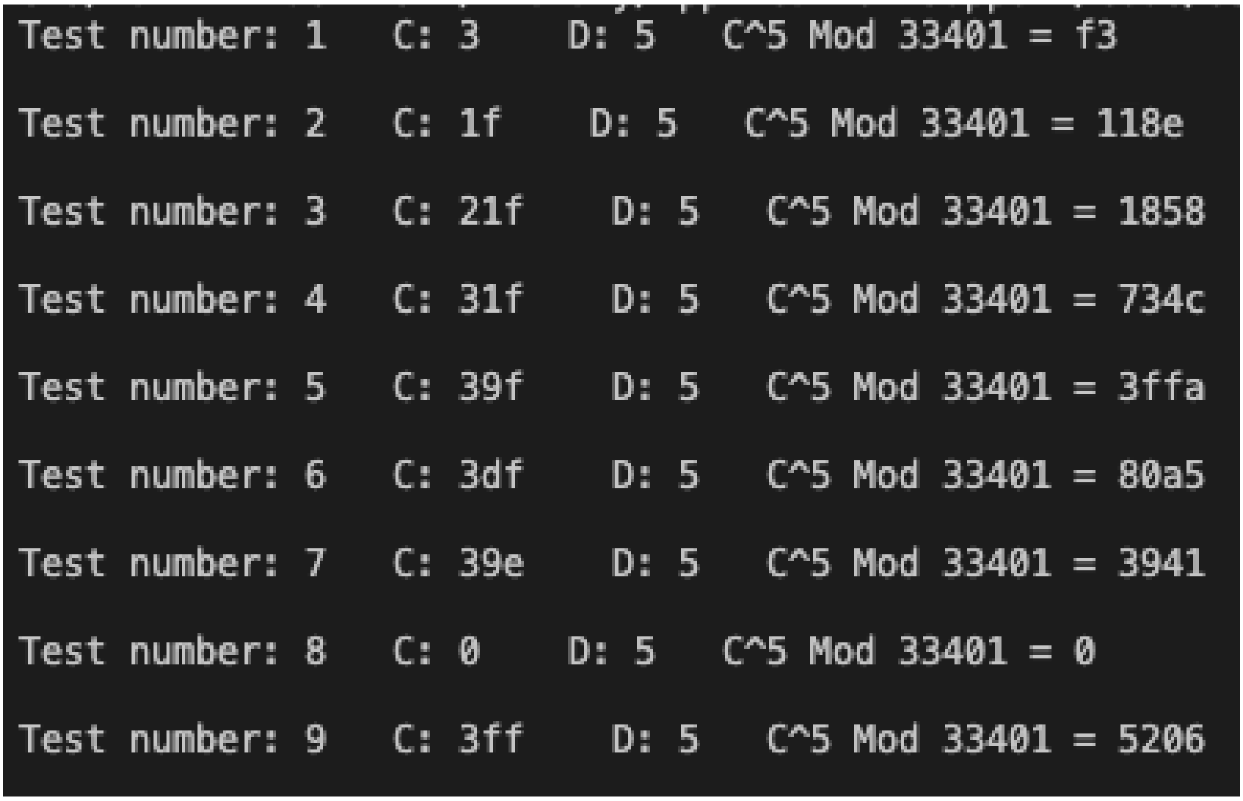


```
always : PROCESS
BEGIN
    --Case 1:
    reset <= '1';
    start <= '0';
    d <= "00000000000101";
    c <= "0000000011";
    wait for 10 ns;
    reset <= '0';
    start <= '1';

    --Case 2:
    wait for 200 ns;
    reset <= '1';
    start <= '0';
    c <= "0000011111";
    wait for 10 ns;
    reset <= '0';
    start <= '1';

    --Case 3:
    wait for 200 ns;
    reset <= '1';
    start <= '0';
    c <= "1000011111";
    wait for 10 ns;
    reset <= '0';
    start <= '1';

    --Case 4:
    wait for 200 ns;
    reset <= '1';
    start <= '0';
    c <= "1100011111";
    wait for 10 ns;
    reset <= '0';
    start <= '1';

    --Case 5:
    wait for 200 ns;
    reset <= '1';
    start <= '0';
    c <= "1110011111";
    wait for 10 ns;
    reset <= '0';
    start <= '1';

    --Case 6:
    wait for 200 ns;
    reset <= '1';
    start <= '0';
    c <= "1111011111";
    wait for 10 ns;
    reset <= '0';
    start <= '1';

    --Case 7:
    wait for 200 ns;
    reset <= '1';
    start <= '0';
    c <= "1110011110";
    wait for 10 ns;
    reset <= '0';
    start <= '1';

    --Case 8:
    wait for 200 ns;
    reset <= '1';
    start <= '0';
    c <= "0000000000";
    wait for 10 ns;
    reset <= '0';
    start <= '1';

    --Case 9:
    wait for 200 ns;
    reset <= '1';
    start <= '0';
    c <= "1111111111";
    wait for 10 ns;
    reset <= '0';
    start <= '1';

WAIT;
END PROCESS always;
END g35_mod_exp_arch;
```

<u>Simulation results:</u>

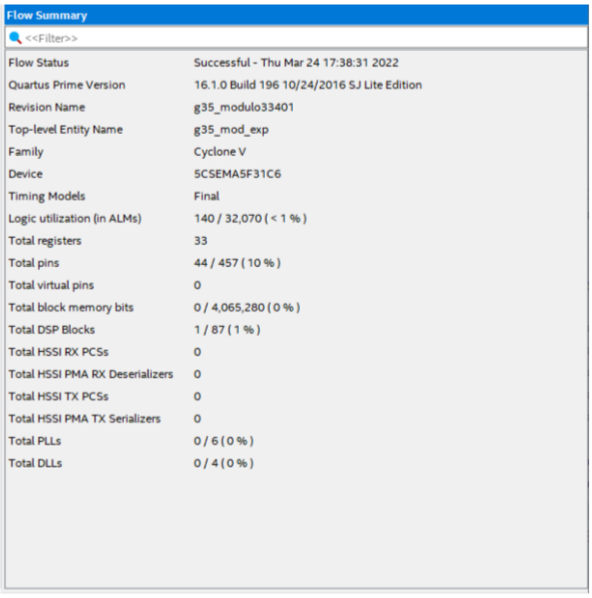


<u>Expected results:</u>



```
Test number: 1   C: 3    D: 5   C^5 Mod 33401 = f3

Test number: 2   C: 1f    D: 5   C^5 Mod 33401 = 118e

Test number: 3   C: 21f    D: 5   C^5 Mod 33401 = 1858

Test number: 4   C: 31f    D: 5   C^5 Mod 33401 = 734c

Test number: 5   C: 39f    D: 5   C^5 Mod 33401 = 3ffa

Test number: 6   C: 3df    D: 5   C^5 Mod 33401 = 80a5

Test number: 7   C: 39e    D: 5   C^5 Mod 33401 = 3941

Test number: 8   C: 0    D: 5   C^5 Mod 33401 = 0

Test number: 9   C: 3ff    D: 5   C^5 Mod 33401 = 5206
```

Flow Summary:

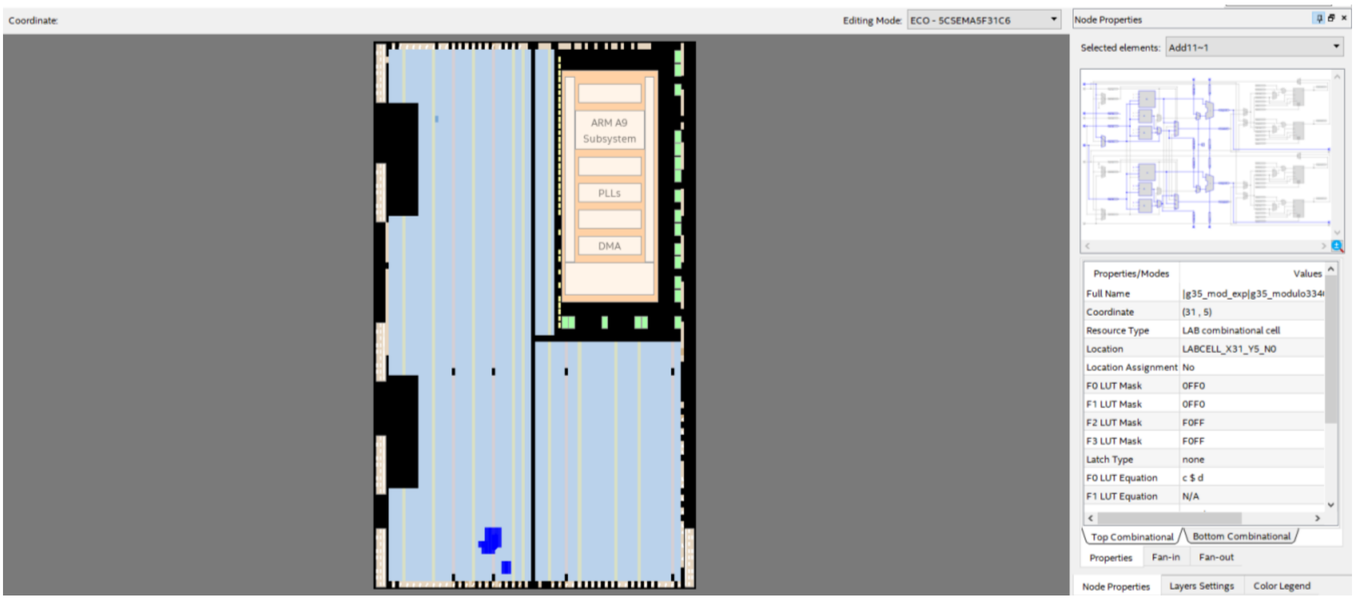


Flow Summary

<<Filter>>

Flow Status	Successful - Thu Mar 24 17:38:31 2022
Quartus Prime Version	16.1.0 Build 196 10/24/2016 SJ Lite Edition
Revision Name	g35_modulo33401
Top-level Entity Name	g35_mod_exp
Family	Cyclone V
Device	5CSEMA5F31C6
Timing Models	Final
Logic utilization (in ALMs)	140 / 32,070 (< 1 %)
Total registers	33
Total pins	44 / 457 (10 %)
Total virtual pins	0
Total block memory bits	0 / 4,065,280 (0 %)
Total DSP Blocks	1 / 87 (1 %)
Total HSSI RX PCSs	0
Total HSSI PMA RX Deserializers	0
Total HSSI TX PCSs	0
Total HSSI PMA TX Serializers	0
Total PLLs	0 / 6 (0 %)
Total DLLs	0 / 4 (0 %)

Chip planner layout:



Selected elements: Add11~1

Properties/Modes	Values
Full Name	|g35_mod_exp|g35_modulo334
Coordinate	(31 , 5)
Resource Type	LAB combinational cell
Location	LABCELL_X31_Y5_N0
Location Assignment	No
F0 LUT Mask	0FF0
F1 LUT Mask	0FF0
F2 LUT Mask	F0FF
F3 LUT Mask	F0FF
Latch Type	none
F0 LUT Equation	c $ d
F1 LUT Equation	N/A

Timing Analysis Summary:

For 4 period:

Requested Fmax = ___250 MHz_________________________

Fast 1100mV 0C Model Hold Slack Value =_________0.365___________

Slow 1100mV 85C Model Setup Slack Value = ________-14.543_________

Slow 1100mV 85C Model Fmax =__________53.93 MHz_________________

List the Worst-case Timing paths for the Setup times:

	Slack	From	To
1	-14.543	multiplication[16]	multiplication[0]
2	-14.543	multiplication[16]	multiplication[1]
3	-14.543	multiplication[16]	multiplication[2]
4	-14.543	multiplication[16]	multiplication[3]
5	-14.543	multiplication[16]	multiplication[4]

Logic utilization (in ALMs): 140 / 32,070 (< 1 %)

For 19 period -> passing

Requested Fmax = ___52.63 MHz_________________________

Fast 1100mV 0C Model Hold Slack Value =_________0.188___________

Slow 1100mV 85C Model Setup Slack Value = ________1.315_________

Slow 1100mV 85C Model Fmax =__________56.55 MHz_________________