dscal
DIGITAL SYSTEMS & COMPUTER ARCHITECTURE LABORATORY

10ο Εργαστήριο Σχεδίασης Ψηφιακών Συστημάτων

4ο Εργαστηριακό Μάθημα

Βασιλόπουλος Διονύσης

Ε.ΔΙ.Π. Τμήματος Πληροφορικής & Τηλεπικοινωνιών

ΤΜΗΜΑ
ΠΛΗΡΟΦΟΡΙΚΗΣ & ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

3η Εργαστηριακή Άσκηση

Ανάλυση άσκησης

Δημιουργία νέου Ρολογιού
=
100 εκ. χτύποι του CLK της κάρτας

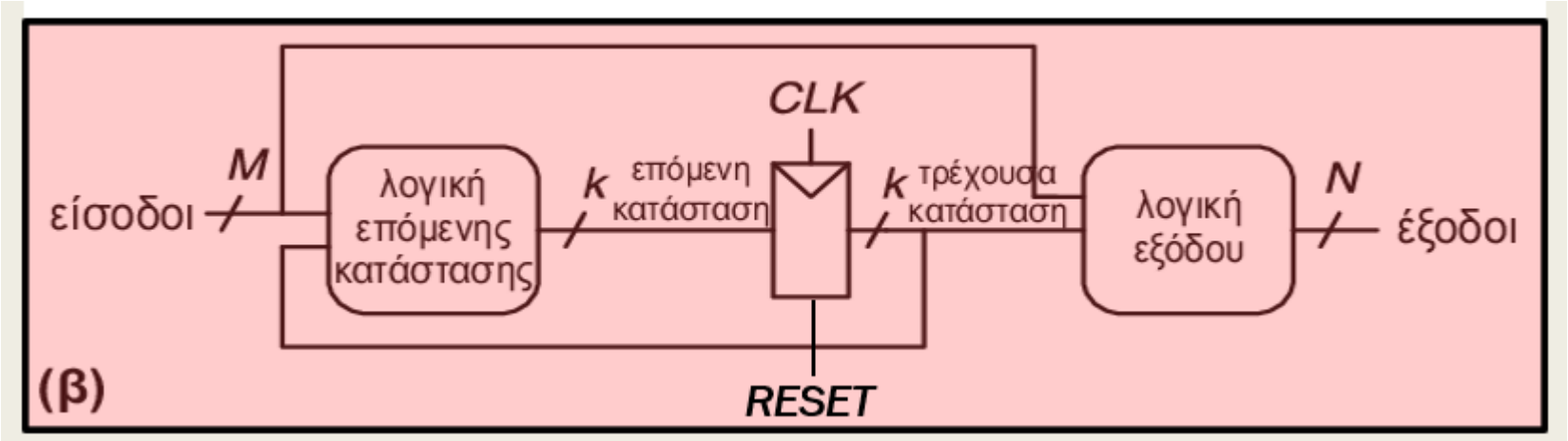
Υπολογισμός επόμενης κατάστασης

Ενημέρωση Τρέχουσας Κατάστασης

Υπολογισμός εξόδου

4η Εργαστηριακή Άσκηση

Ανάλυση άσκησης



4η Εργαστηριακή Άσκηση

ΠΡΟΧΩΡΗΣΤΕ ΣΤΗΝ ΑΣΚΗΣΗ

4η Εργαστηριακή Άσκηση

Αρχιτεκτονική (1/4) – Count 1 sec (Περίοδος = 1 sec)

```
constant TICKS: positive  :=  1024;
constant ONESEC: positive :=100000000;

signal sum: unsigned(6 downto 0);
signal new_sum: unsigned(6 downto 0);
signal right_digit: unsigned(3 downto 0);
signal left_digit: unsigned(3 downto 0);
signal seven_segment_left: std_logic_vector(6 downto
0);
signal seven_segment_right: std_logic_vector(6 downto
0);
signal digit_selection: std_logic;
signal new_digit_selection: std_logic;
signal clk_ticks : integer range 0 to ONESEC:=0;
signal new_clk_ticks : integer range 0 to ONESEC:=0;
```

```
new_clk_ticks<=clk_ticks+1 when clk_ticks<ONESEC else 1;

count_ticks : process (clk, reset) is
begin
        if reset = '1' then
    clk_ticks <=0;
        elsif rising_edge(clk) then
                clk_ticks <= new_clk_ticks;
        end if; --reset
end process count_ticks;
```

4η Εργαστηριακή Άσκηση

Αρχιτεκτονική (2/4) – Tick Counter per second

```
count_sec: process(clk_ticks) is
begin
if clk_ticks=100000000 then
   if direction='1' then
      if sum<99 then
         new_sum<=sum+1;
      else
         new_sum<=(others=>'0');
      end if; -- sum<99
   else
      if sum>0 then
         new_sum<=sum-1;
      else
         new_sum<="1100011";
      end if; -- sum>0
   end if; --direction
else
   new_sum<=sum;
end if;
end process count_sec;
```

Register

Υπολογισμός εισόδου Register

```
count: process (clk, reset) is
begin
if (reset='1') then
   sum<=(others=>'0');
elsif rising_edge(clk) then
   sum<=new_sum;
end if; --reset
end process count;
```

4η Εργαστηριακή Άσκηση

Αρχιτεκτονική (3/4) – Switch every TICKS*10 ns to each digit

```
new_digit_selection<= not digit_selection when (clk_ticks mod TICKS)=0 else
                      digit_selection ;

select_digit: process (clk, reset) is
begin
if (reset='1') then
   digit_selection<='0';
elsif rising_edge(clk) then
   digit_selection<=new_digit_selection;
end if; --reset
end process select_digit;

digit_selection_out<=digit_selection;
```

4η Εργαστηριακή Άσκηση

Αρχιτεκτονική (4/4) – Rest of Output signals

```
led_result<=std_logic_vector(sum);

left_digit<=resize(sum/10, left_digit'length);
right_digit<=resize(sum mod 10, right_digit'length);

right_digit_values: process (right_digit) begin
case right_digit is
   when X"0" => seven_segment_right   <="0111111";   -- 0
   when X"1" => seven_segment_right   <="0000110";   -- 1
   when X"2" => seven_segment_right   <="1011011";   -- 2
   when X"3" => seven_segment_right   <="1001111";   -- 3
   when X"4" => seven_segment_right   <="1100110";   -- 4
   when X"5" => seven_segment_right   <="1101101";   -- 5
   when X"6" => seven_segment_right   <="1111101";   -- 6
   when X"7" => seven_segment_right   <="0000111";   -- 7
   when X"8" => seven_segment_right   <="1111111";   -- 8
   when X"9" => seven_segment_right   <="1101111";   -- 9
   when others => seven_segment_right <="1000000";
end case;
end process right_digit_values;
```

```
left_digit_values: process (left_digit) begin
case left_digit is
   when X"0" => seven_segment_left   <="0111111";   -- 0
   when X"1" => seven_segment_left   <="0000110";   -- 1
   when X"2" => seven_segment_left   <="1011011";   -- 2
   when X"3" => seven_segment_left   <="1001111";   -- 3
   when X"4" => seven_segment_left   <="1100110";   -- 4
   when X"5" => seven_segment_left   <="1101101";   -- 5
   when X"6" => seven_segment_left   <="1111101";   -- 6
   when X"7" => seven_segment_left   <="0000111";   -- 7
   when X"8" => seven_segment_left   <="1111111";   -- 8
   when X"9" => seven_segment_left   <="1101111";   -- 9
   when others => seven_segment_left <="0000000";
end case;
end process left_digit_values;

seven_segment<=seven_segment_right when digit_selection='0' else      s
                  even_segment_left when digit_selection='1' else
                 "1000000";
```

4η Εργαστηριακή Άσκηση

Constraints (1/2) – CLK + Switches + Leds

Constraints →

```
# CLK - Zedboard 100MHz oscillator
set_property -dict { PACKAGE_PIN Y9 IOSTANDARD LVCMOS33 } [get_ports {clk}]

##############################
# On-board Slide Switches  #
##############################
set_property -dict { PACKAGE_PIN M15   IOSTANDARD LVCMOS33 } [get_ports { reset }];
set_property -dict { PACKAGE_PIN F22   IOSTANDARD LVCMOS33 } [get_ports { direction }];

##############################
# On-board LEDS  #
##############################
set_property -dict { PACKAGE_PIN T22   IOSTANDARD LVCMOS33 } [get_ports { led_result[0] }];
set_property -dict { PACKAGE_PIN T21   IOSTANDARD LVCMOS33 } [get_ports { led_result[1] }];
set_property -dict { PACKAGE_PIN U22   IOSTANDARD LVCMOS33 } [get_ports { led_result[2] }];
set_property -dict { PACKAGE_PIN U21   IOSTANDARD LVCMOS33 } [get_ports { led_result[3] }];
set_property -dict { PACKAGE_PIN V22   IOSTANDARD LVCMOS33 } [get_ports { led_result[4] }];
set_property -dict { PACKAGE_PIN W22   IOSTANDARD LVCMOS33 } [get_ports { led_result[5] }];
set_property -dict { PACKAGE_PIN U19   IOSTANDARD LVCMOS33 } [get_ports { led_result[6] }];
```

4η Εργαστηριακή Άσκηση

Constraints (1/2) – Pmod + CLK

Constraints →

```
##############################
# PmodSSO                    #
##############################

set_property -dict { PACKAGE_PIN Y11   IOSTANDARD LVCMOS33 } [get_ports { seven_segment[0] }];
set_property -dict { PACKAGE_PIN AA11  IOSTANDARD LVCMOS33 } [get_ports { seven_segment[1] }];
set_property -dict { PACKAGE_PIN Y10   IOSTANDARD LVCMOS33 } [get_ports { seven_segment[2] }];
set_property -dict { PACKAGE_PIN AA9   IOSTANDARD LVCMOS33 } [get_ports { seven_segment[3] }];
set_property -dict { PACKAGE_PIN W12   IOSTANDARD LVCMOS33 } [get_ports { seven_segment[4] }];
set_property -dict { PACKAGE_PIN W11   IOSTANDARD LVCMOS33 } [get_ports { seven_segment[5] }];
set_property -dict { PACKAGE_PIN V10   IOSTANDARD LVCMOS33 } [get_ports { seven_segment[6] }];

set_property -dict { PACKAGE_PIN W8   IOSTANDARD LVCMOS33 } [get_ports { digit_selection_out }];

##############################################################
##ZedBoard Timing Constraints
##############################################################
# define clock and period
create_clock -period 10 -name CLK -waveform {0.000 5.000} [get_ports {clk}]
```

4η Εργαστηριακή Άσκηση

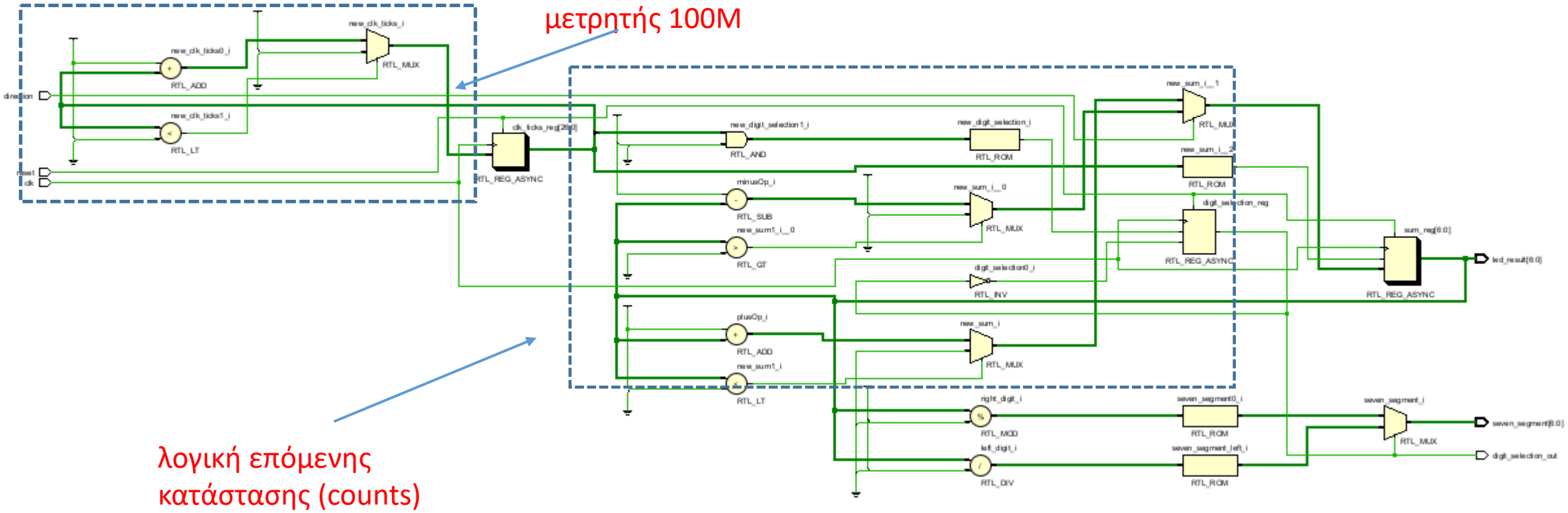
Pmod Manual

Manual Pmod

https://reference.digilentinc.com/reference/pmod/pmodssd/reference-manual

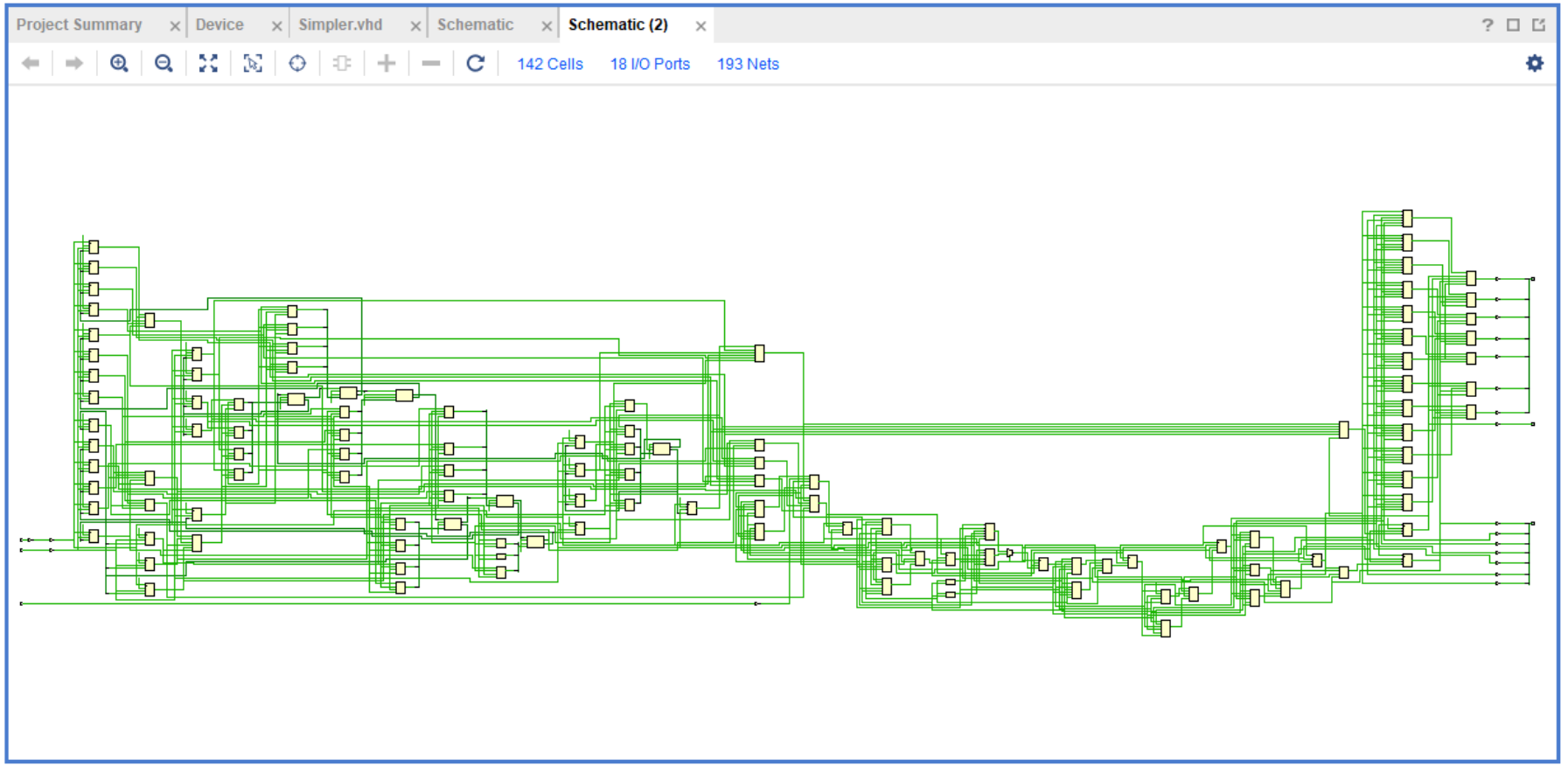
4η Εργαστηριακή Άσκηση

RTL Design



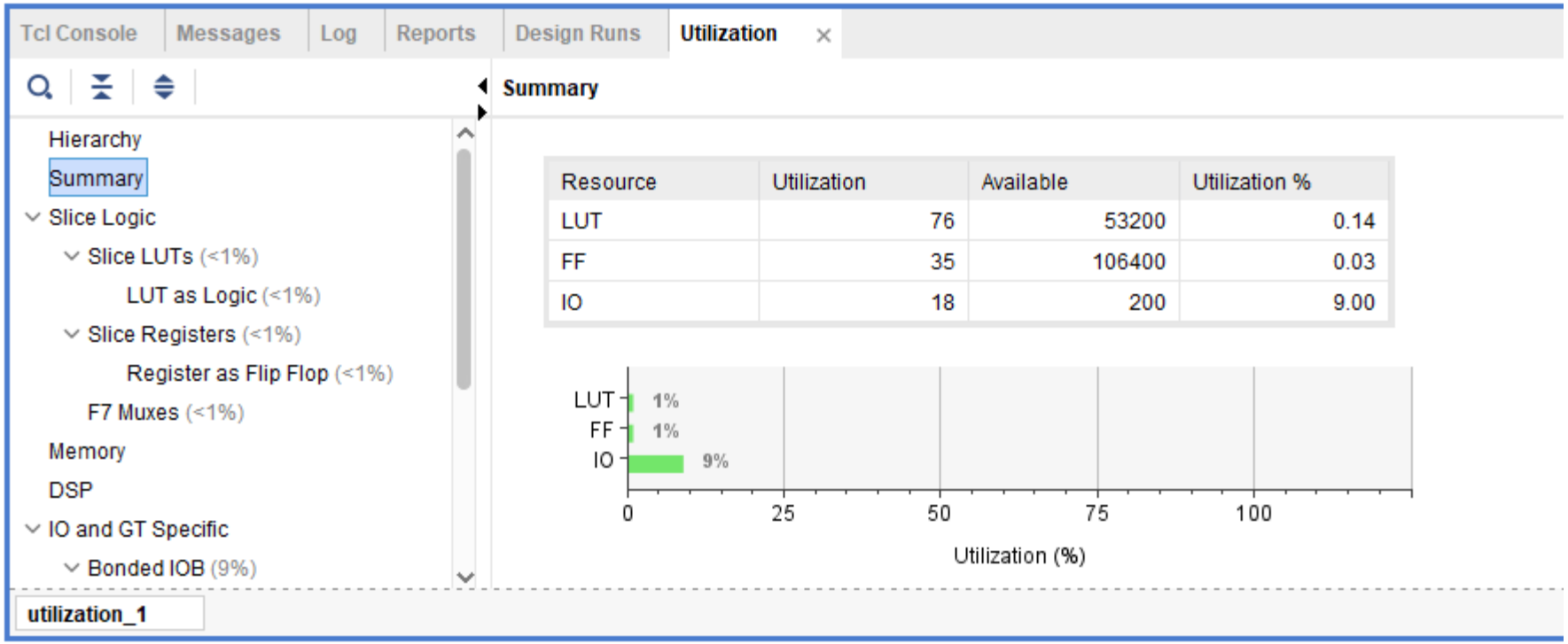
4η Εργαστηριακή Άσκηση

Synthesis/Implementation – Schematic



4η Εργαστηριακή Άσκηση

Implementation – Report Utilization

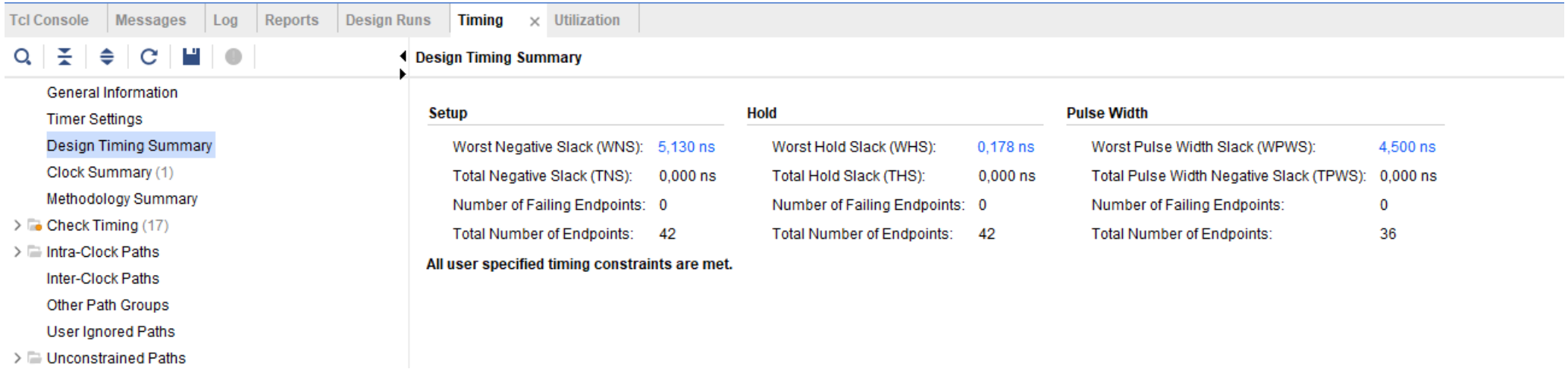


Και από Project Summary

4η Εργαστηριακή Άσκηση

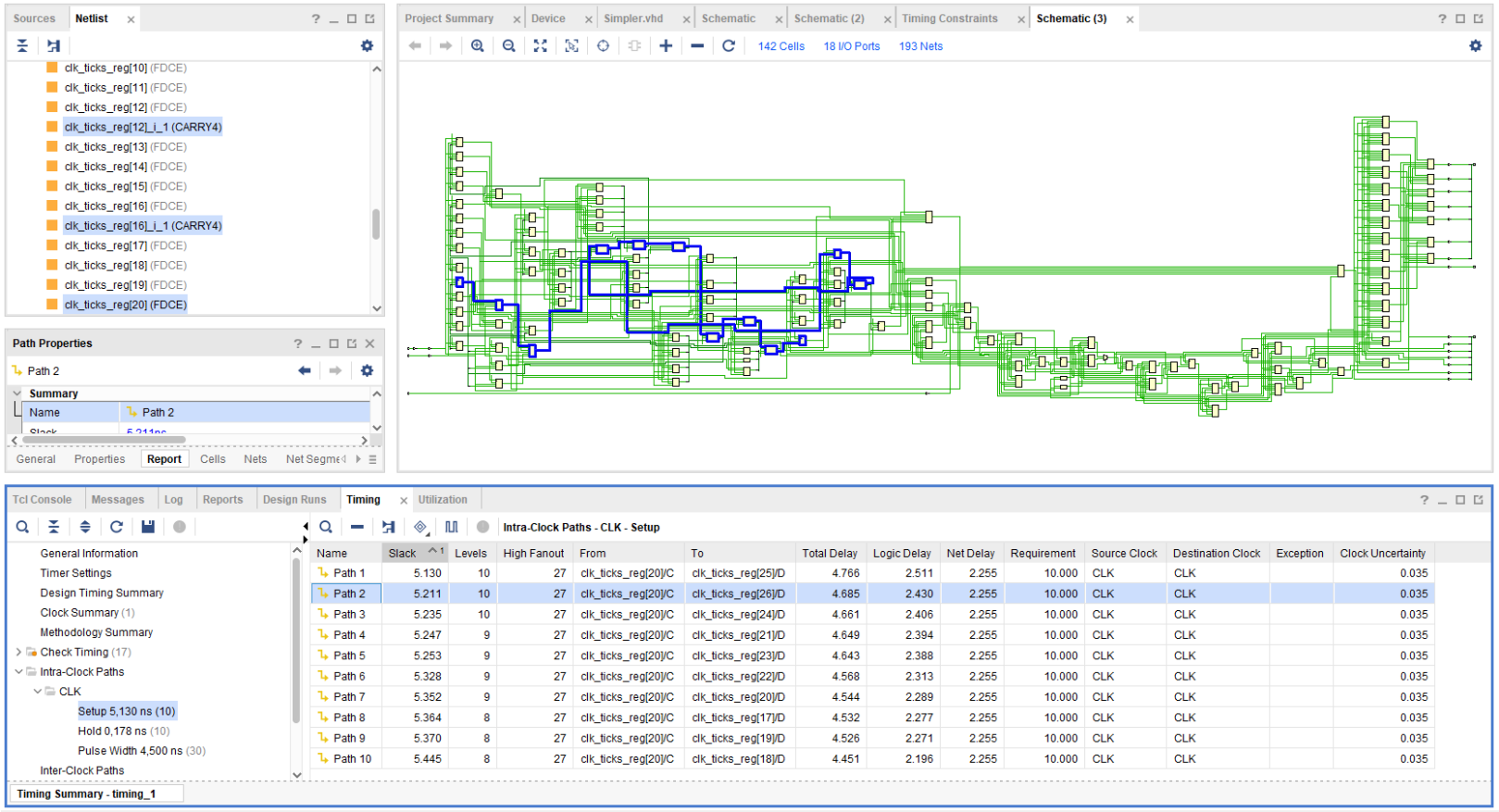
Implementation – Timing Reports (1/4)

Menu Reports->Timing-> Report Timing Summary
ή
Open Implemented Design->Report Timing Summary



4η Εργαστηριακή Άσκηση

Implementation – Timing Reports (2/4) - Setup

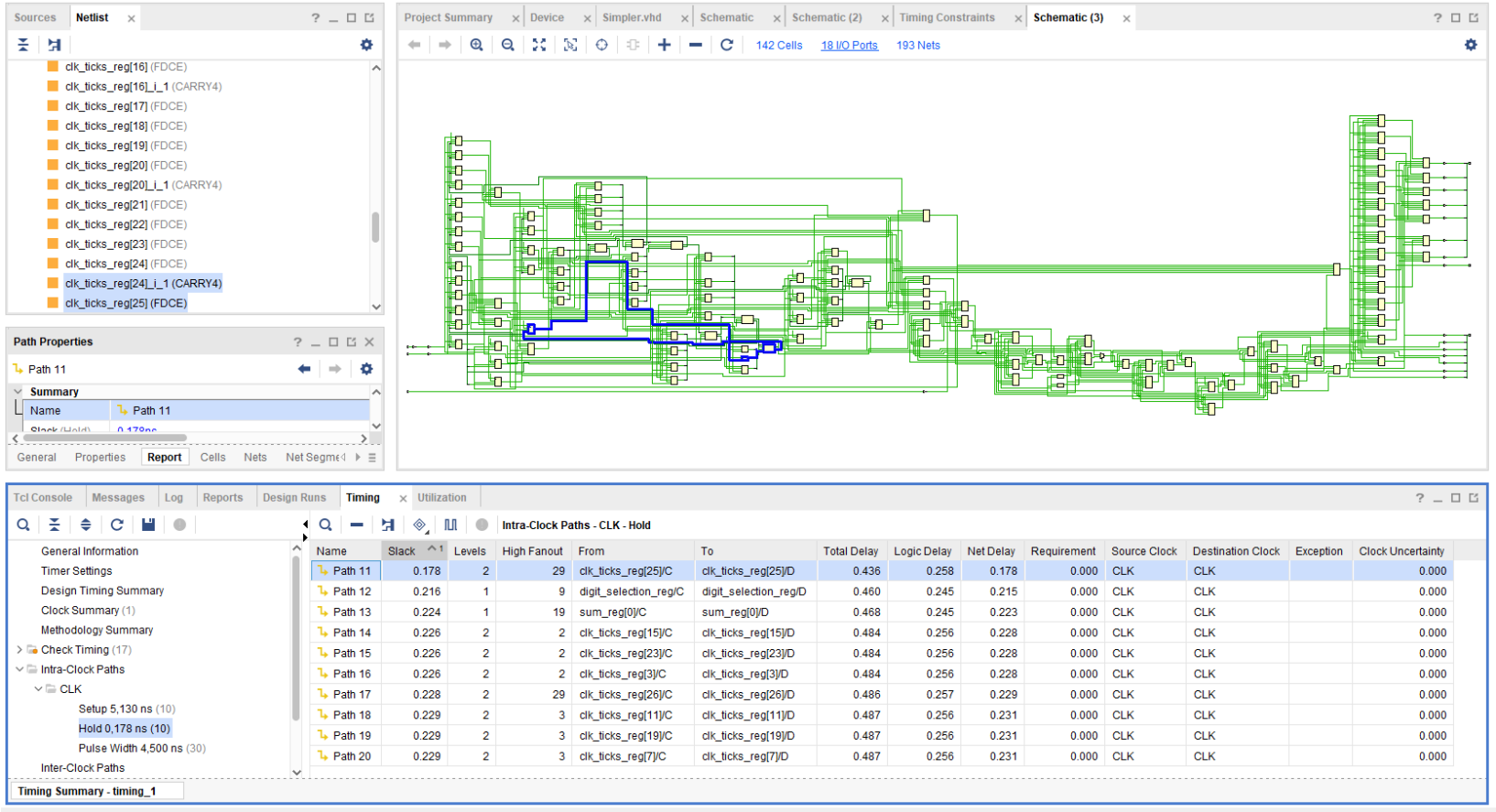


Name	Slack	Levels	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock	Destination Clock	Exception	Clock Uncertainty
Path 1	5.130	10	27	clk_ticks_reg[20]/C	clk_ticks_reg[25]/D	4.766	2.511	2.255	10.000	CLK	CLK		0.035
Path 2	5.211	10	27	clk_ticks_reg[20]/C	clk_ticks_reg[26]/D	4.685	2.430	2.255	10.000	CLK	CLK		0.035
Path 3	5.235	10	27	clk_ticks_reg[20]/C	clk_ticks_reg[24]/D	4.661	2.406	2.255	10.000	CLK	CLK		0.035
Path 4	5.247	9	27	clk_ticks_reg[20]/C	clk_ticks_reg[21]/D	4.649	2.394	2.255	10.000	CLK	CLK		0.035
Path 5	5.253	9	27	clk_ticks_reg[20]/C	clk_ticks_reg[23]/D	4.643	2.388	2.255	10.000	CLK	CLK		0.035
Path 6	5.328	9	27	clk_ticks_reg[20]/C	clk_ticks_reg[22]/D	4.568	2.313	2.255	10.000	CLK	CLK		0.035
Path 7	5.352	9	27	clk_ticks_reg[20]/C	clk_ticks_reg[20]/D	4.544	2.289	2.255	10.000	CLK	CLK		0.035
Path 8	5.364	8	27	clk_ticks_reg[20]/C	clk_ticks_reg[17]/D	4.532	2.277	2.255	10.000	CLK	CLK		0.035
Path 9	5.370	8	27	clk_ticks_reg[20]/C	clk_ticks_reg[19]/D	4.526	2.271	2.255	10.000	CLK	CLK		0.035
Path 10	5.445	8	27	clk_ticks_reg[20]/C	clk_ticks_reg[18]/D	4.451	2.196	2.255	10.000	CLK	CLK		0.035

Setup Time:
Αναφέρεται στις αργές διαδρομές (καθυστέρηση διάδοσης)

4η Εργαστηριακή Άσκηση

Implementation – Timing Reports (3/4) - Hold



Name	Slack	Levels	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source Clock	Destination Clock	Exception	Clock Uncertainty
Path 11	0.178	2	29	clk_ticks_reg[25]/C	clk_ticks_reg[25]/D	0.436	0.258	0.178	0.000	CLK	CLK		0.000
Path 12	0.216	1	9	digit_selection_reg/C	digit_selection_reg/D	0.460	0.245	0.215	0.000	CLK	CLK		0.000
Path 13	0.224	1	19	sum_reg[0]/C	sum_reg[0]/D	0.468	0.245	0.223	0.000	CLK	CLK		0.000
Path 14	0.226	2	2	clk_ticks_reg[15]/C	clk_ticks_reg[15]/D	0.484	0.256	0.228	0.000	CLK	CLK		0.000
Path 15	0.226	2	2	clk_ticks_reg[23]/C	clk_ticks_reg[23]/D	0.484	0.256	0.228	0.000	CLK	CLK		0.000
Path 16	0.226	2	2	clk_ticks_reg[3]/C	clk_ticks_reg[3]/D	0.484	0.256	0.228	0.000	CLK	CLK		0.000
Path 17	0.228	2	29	clk_ticks_reg[26]/C	clk_ticks_reg[26]/D	0.486	0.257	0.229	0.000	CLK	CLK		0.000
Path 18	0.229	2	3	clk_ticks_reg[11]/C	clk_ticks_reg[11]/D	0.487	0.256	0.231	0.000	CLK	CLK		0.000
Path 19	0.229	2	3	clk_ticks_reg[19]/C	clk_ticks_reg[19]/D	0.487	0.256	0.231	0.000	CLK	CLK		0.000
Path 20	0.229	2	3	clk_ticks_reg[7]/C	clk_ticks_reg[7]/D	0.487	0.256	0.231	0.000	CLK	CLK		0.000

Hold Time:
Αναφέρεται στις γρήγορες διαδρομές (καθυστέρηση μόλυνσης)

4η Εργαστηριακή Άσκηση

Implementation – Timing Reports (4/4)

TCL Console>*report_timing_summary –datasheet*

(FULL REPORT)

Εμφανίζεται στη TCL console
(κοιτάξτε τα combinational delays)