



GEBZE TEKNİK ÜNİVERSİTESİ
ELEKTRONİK MÜHENDİSLİĞİ

ELM235

LOJİK DEVRE TASARIM LABORATUVARI

LAB 0x3 Deney Raporu

Sıralı Mantık Tasarım

Hazırlayanlar
1) 1801022035 – Ruveyda Dilara Günal
2) 200102002087 – Alican Bayındır

1. Problem

1.1 Serbest sayıcı devresi

Bu problemde devre aktif iken (en sinyali) her clock rising edge de yon girişine bağlı olarak 0 sayısından verilen psc değerine veya psc değerinden 0 a kadar sayıp, hedefe ulaştığında tick sinyali oluşturan bir devre tasarlayacaksınız. (Aşağı/yukarı sayıcı)

- Devrenin disable edilmesi registerlarınızı değiştirmeyecek fakat sayma işlemini donduracak.
- Devre geri active edildiğinde, o anda registerların değeri neyse aynı şekilde çalışmaya devam edecek.
- psc sinyalini 8 bitlik alınız. Ayrıca bir active-low reset sinyali ekleyip, reset geldiğinde counterınızı 0 layınız. Örnek modül portları aşağıda verilmiştir. Modülünüz için basit bir testbench oluşturup, birkaç farklı psc değerlerine göre test ediniz.
- yon girişi 0 ise yukarı doğru, 1 ise aşağı doğru sayınız.

A. Devreyi HDL ile tasarlayın

B. Farklı psc, yon, en ve reset girişlerine göre testbench oluşturarak, devrenin farklı kombinasyonlarda doğru çalıştığını gözlemleyin.

C. Devreyi Quartus'da sentezleyerek devrenin ne kadar yer kapladığını (resource utilization report), sentezlenen RTL ve eşleştirme ardı devre şemalarını ekleyerek yorumlayınız.

```
module lab3_g29_p1 (  
    input logic clk, rst, en,  
    input logic yon,  
    input logic [7:0] psc,  
    output logic tick  
);  
  
    logic [7:0] counter;  
  
    always @(posedge clk)  
        if (yon)  
            begin  
                counter <= 8'b0;  
  
                if (en)
```

```

begin
    assign tick = (counter == psc)? 1:0;
    if(tick)
        counter <= 0;
    else
        counter <= counter + 1 ;
    end
end
else
begin
    assign counter = psc;
    if (en)
        begin
            assign tick = (counter == 0)? 1:0;
            if(tick)
                counter <= psc;
            else
                counter <= counter - 1;
            end
        end
    end
end
endmodule

```

Yukarıdaki kod soruda istenen temel isterleri – her clockta ileri sayma, geri sayma, tick sinyalinin 1-0 yapma, gibi isterleri yerine getirmektedir. Bütün bu isterleri yapmasını sağlayan testbench kodu ise aşağıda verildiği gibidir;

```
`timescale 1ps/1ps

module tb_lab3_g29_p1();

    logic clk, rst, en;

    logic yon;

    logic [7:0] psc;

    logic tick;

    logic [7:0] counter;

    lab3_g29_p1 dut0(clk, rst, en, yon, psc, tick);

    always begin

        clk = 1;          #5;

        clk = 0;          #5;

        en = 1;

    end

    initial begin

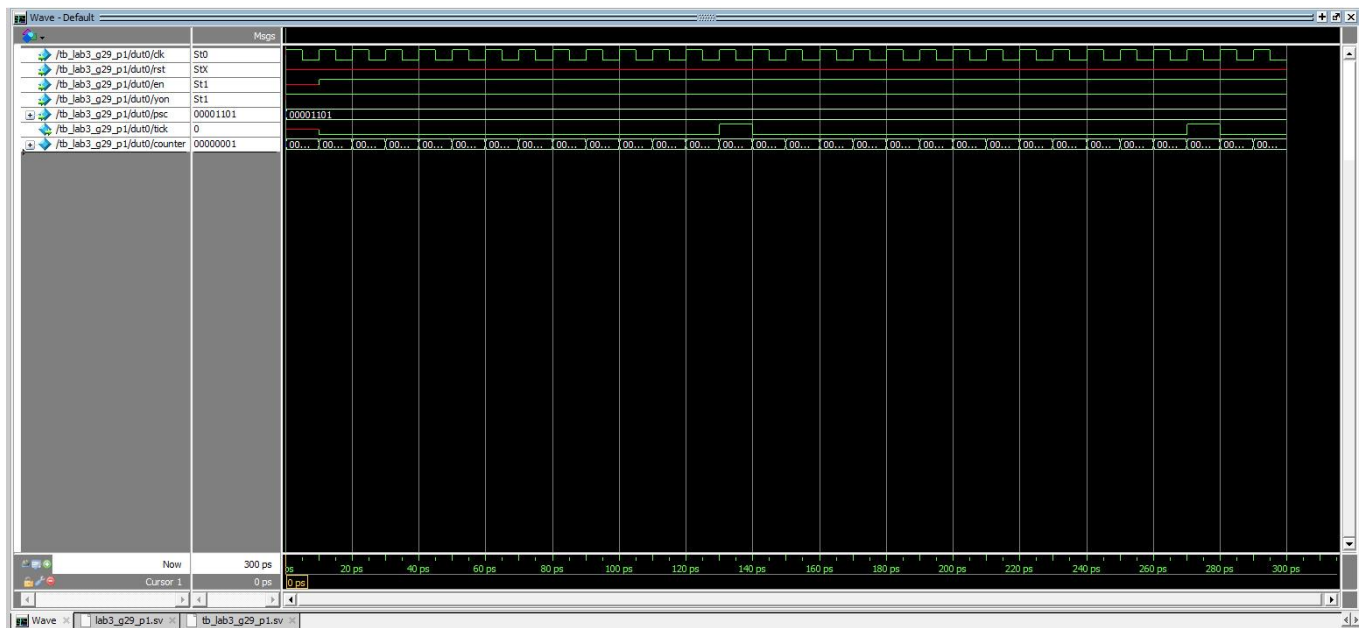
        yon = 1; psc = 8'b00001101; #300;

        $stop;

    end

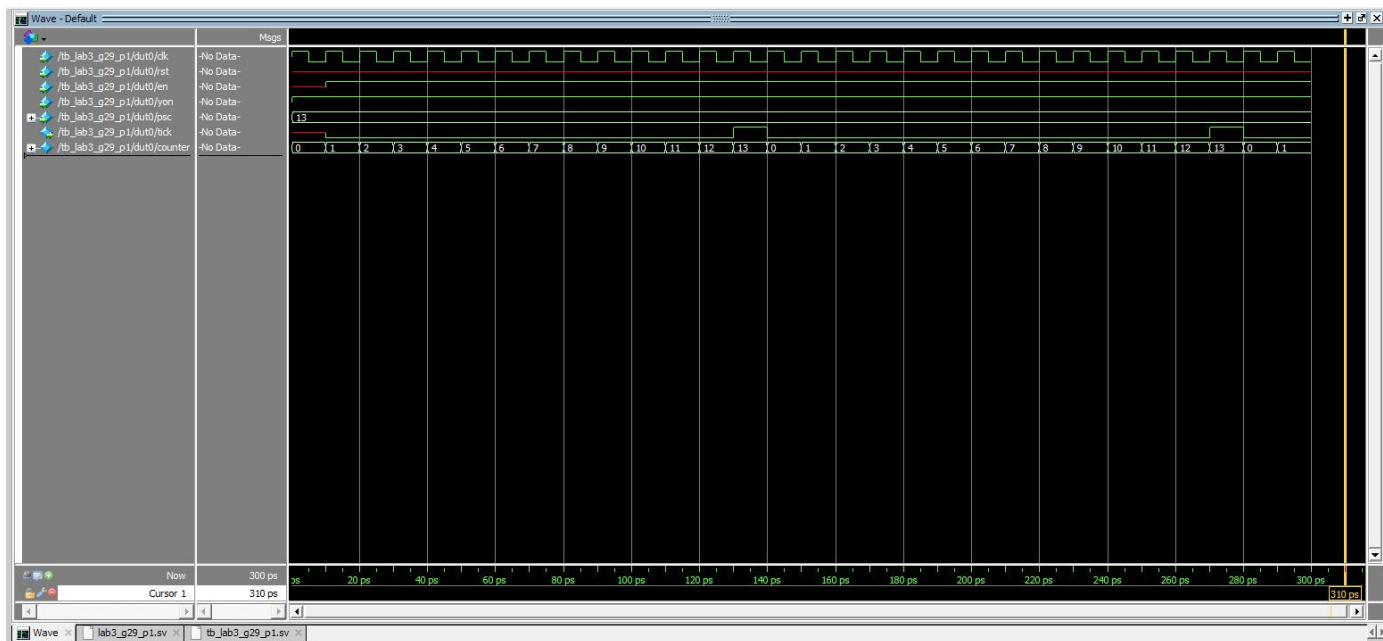
endmodule
```

Yukarıdaki örnek kod parçaları çalıştırıldığı zaman çıkan dalga formları aşağıdaki gibidir;



Şekil 1 Modelsim dalga formları – Binary

Çıkan dalga formları daha kolay kontrol için Binary'den Decimal'e çevirdiğimizde;



Şekil 2 Modelsim dalga formu - Decimal

Geri ve ileri sayma – tick atma fonksiyonlarının çalıştığı görülmektedir. Aynı kod parçası “Quartus” uygulamasında çalıştırıldığı zaman aşağıdaki hata alınmıştır;

Type	ID	Message

		Running Quartus Prime Analysis & Elaboration
		Command: quartus_map --read_settings_files=on --write_settings_files=off lab3_g29_p1 -c lab3_g29_p1_top --analysis_and_elaboration
	18236	Number of processors has not been specified which may cause overloading on shared machines. Set the global assignment NUM_PARALLEL_PROCE
	20030	Parallel compilation is enabled and will use 4 of the 4 processors detected
	12021	Found 1 design units, including 1 entities, in source file lab3_g29_p1.sv
	12021	Found 1 design units, including 1 entities, in source file lab3_g29_p1_top.bdf
	12127	Elaborating entity "lab3_g29_p1_top" for the top level hierarchy
	12128	Elaborating entity "lab3_g29_p1" for hierarchy "lab3_g29_p1:inst"
	10043	Verilog HDL unsupported feature error at lab3_g29_p1.sv(16): Procedural Continuous Assignment to register is not supported
	12152	Can't elaborate user hierarchy "lab3_g29_p1:inst"
		Quartus Prime Analysis & Elaboration was unsuccessful. 2 errors, 1 warning

Şekil 3 Quartus'ta devre sentezlenirken alınan hata.

14	if (en)
15	begin
16	if (counter == psc)
17	assign tick = 1;
18	else
19	assign tick = 0;
20	if(tick)
21	counter <= 0;
22	else
23	counter <= counter + 1 ;
24	end
25	end
26	
27	else
28	begin
29	assign counter = psc;
30	if (en)
31	begin
32	if (counter == 0)
33	assign tick = 1;

Şekil 4 Hata oluşturan kod parçası.

Bu hatanın sebebi olarak Quartus uygulamasında aşağıdaki bilgiler verilmiştir;

- CAUSE:** In an Always Construct, at the specified location in a [Verilog Design File \(.v\)](#), you used a Procedural Continuous Assignment Statement to specify an assignment to a register. However, although the Procedural Continuous Assignment Statement, which overrides a regular assignment, is supported in Verilog HDL, it is not supported in the Quartus II software.
- ACTION:** Either remove the Procedural Continuous Assignment Statement, or try to merge the assignment into other Always Constructs that contain assignments to the same register, by using an If Statement to specify the assignment conditions instead of using a Procedural Continuous Assignment Statement.

Kod üzerinde kullanılan bazı blokların Quartus 2 tarafından desteklenmediği öğrenildiği zaman aşağıdaki bazı değişiklikler koda yapılmıştır;

```
if (counter == psc)

    tick <= 1;

else

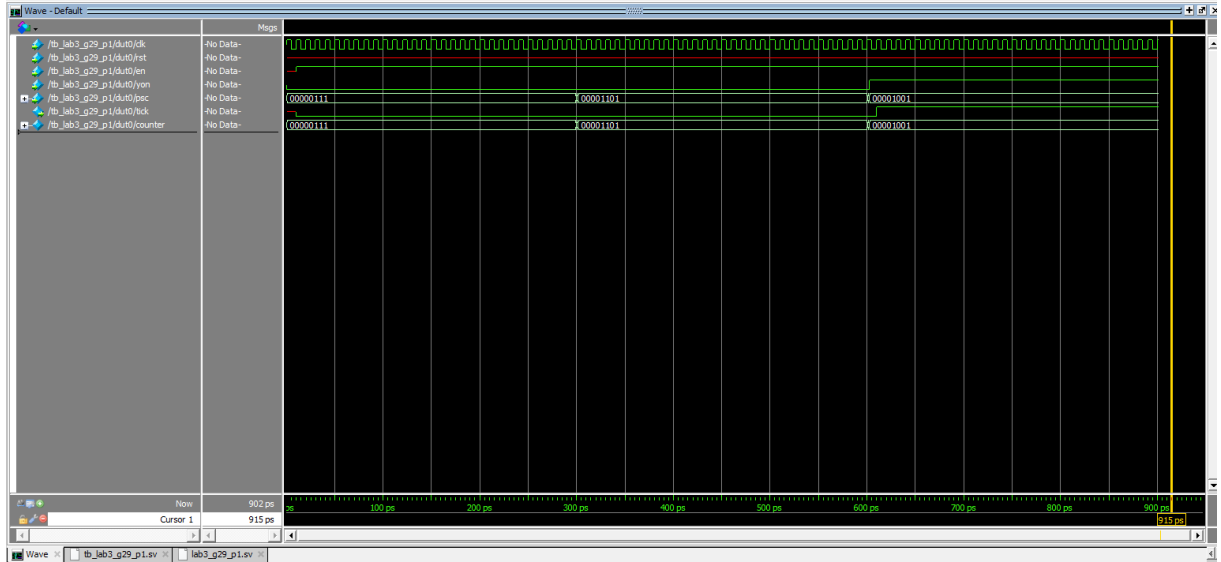
    tick <= 0;
```

Uygulanan değişikliklerden sonra hala aynı hata devam etmektedir.

```
> [i]      running quartus prime Analysis & Elaboration
[!]      Command: quartus_map --read_settings_files=on --write_settings_files=off lab3_g29_p1 -c lab3_g29_p1_top --analysis_and_elaboration
[!]      18236 Number of processors has not been specified which may cause overloading on shared machines. Set the global assignment NUM_PARALLEL_PRC
[!]      20030 Parallel compilation is enabled and will use 4 of the 4 processors detected
> [i]      12021 Found 1 design units, including 1 entities, in source file lab3_g29_p1_top.bdf
> [i]      12021 Found 1 design units, including 1 entities, in source file lab3_g29_p1_top.bdf
[!]      12127 Elaborating entity "lab3_g29_p1_top" for the top level hierarchy
[!]      12128 Elaborating entity "lab3_g29_p1" for hierarchy "lab3_g29_p1:inst"
[!]      10230 Verilog HDL assignment warning at lab3_g29_p1.v(23): truncated value with size 32 to match size of target (8)
[!]      10043 Verilog HDL unsupported feature error at lab3_g29_p1.v(29): Procedural Continuous Assignment to register is not supported
[!]      12152 Can't elaborate user hierarchy "lab3_g29_p1:inst"
> [x]      Quartus Prime Analysis & Elaboration was unsuccessful. 2 errors, 2 warnings
```

Şekil 5 Quartus'ta alınan hata

Farklı sayılarla denendiği zaman çıkan dalga formu aşağıda görüldüğü gibidir ancak geri sayma fonksiyonu burada çalışmamıştır. Sebebi aşağıda açıklanmıştır.



Şekil 6 Modelsim'de geri sayım hatası.

Şekil 6'daki dalga formlarını çalıştırmak için gerekli kod parçası.

```
`timescale 1ps/1ps

module tb_lab3_g29_p1();

    logic clk, rst, en;

    logic yon;

    logic [7:0] psc;

    logic tick;

    logic [7:0] counter;

    lab3_g29_p1 dut0(clk, rst, en, yon, psc, tick);

    always begin

        clk = 1;          #5;

        clk = 0;          #5;

        en = 1;

    end

    initial begin

        yon = 0; psc = 8'b000000111; #300;

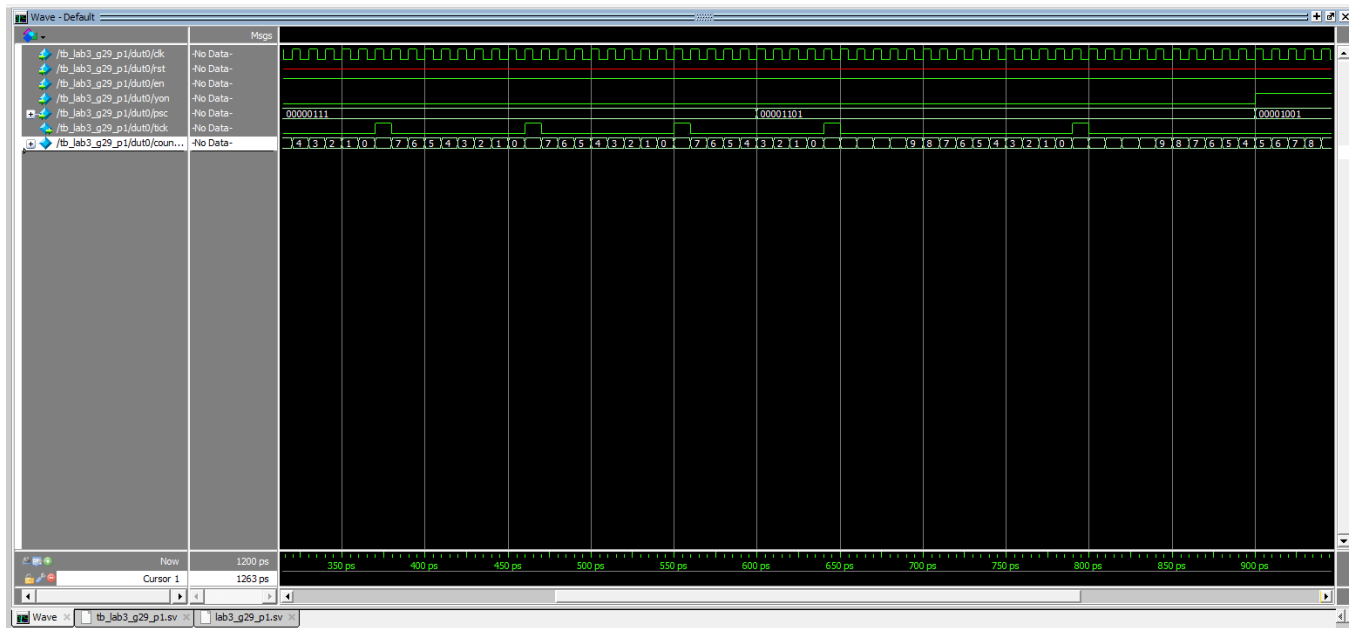
        yon = 0; psc = 8'b000001101; #300;

        yon = 1; psc = 8'b000001001; #300;

        $stop;

    end

endmodule
```

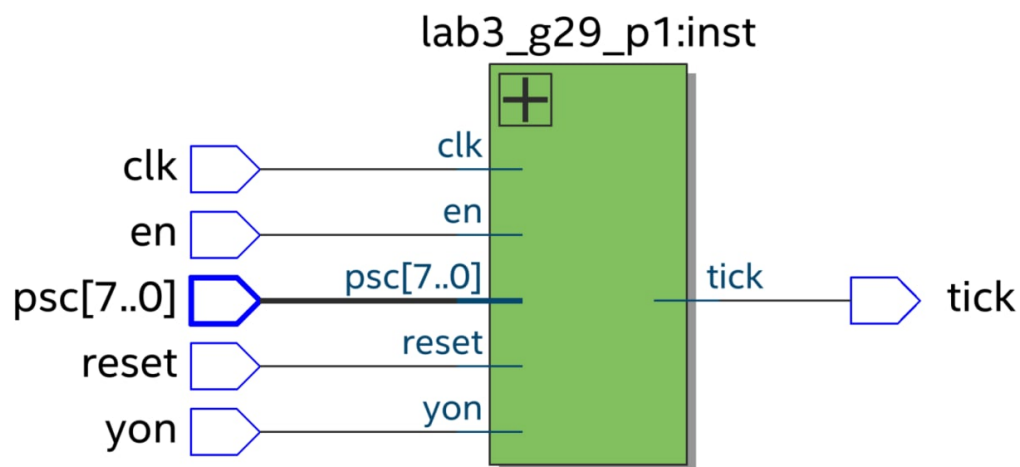
Aşağıdaki if bloğunun else kısmında kırmızı ile gösterilen işlem yapıldığı zaman sorun çözülmüş ve devre hatasız bir şekilde çalışmıştır.

else begin

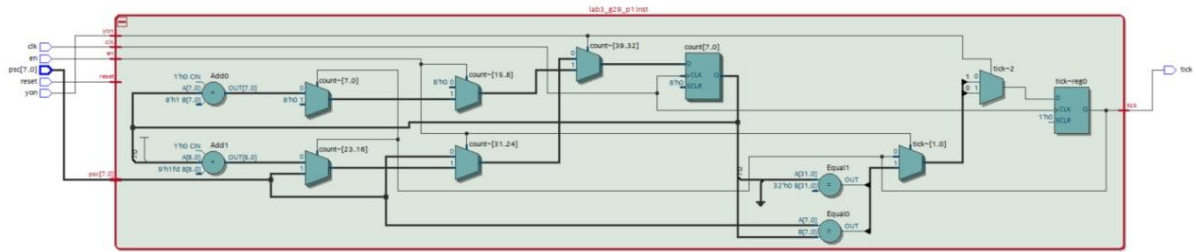
```
counter <= psc;
```

if (en) begin

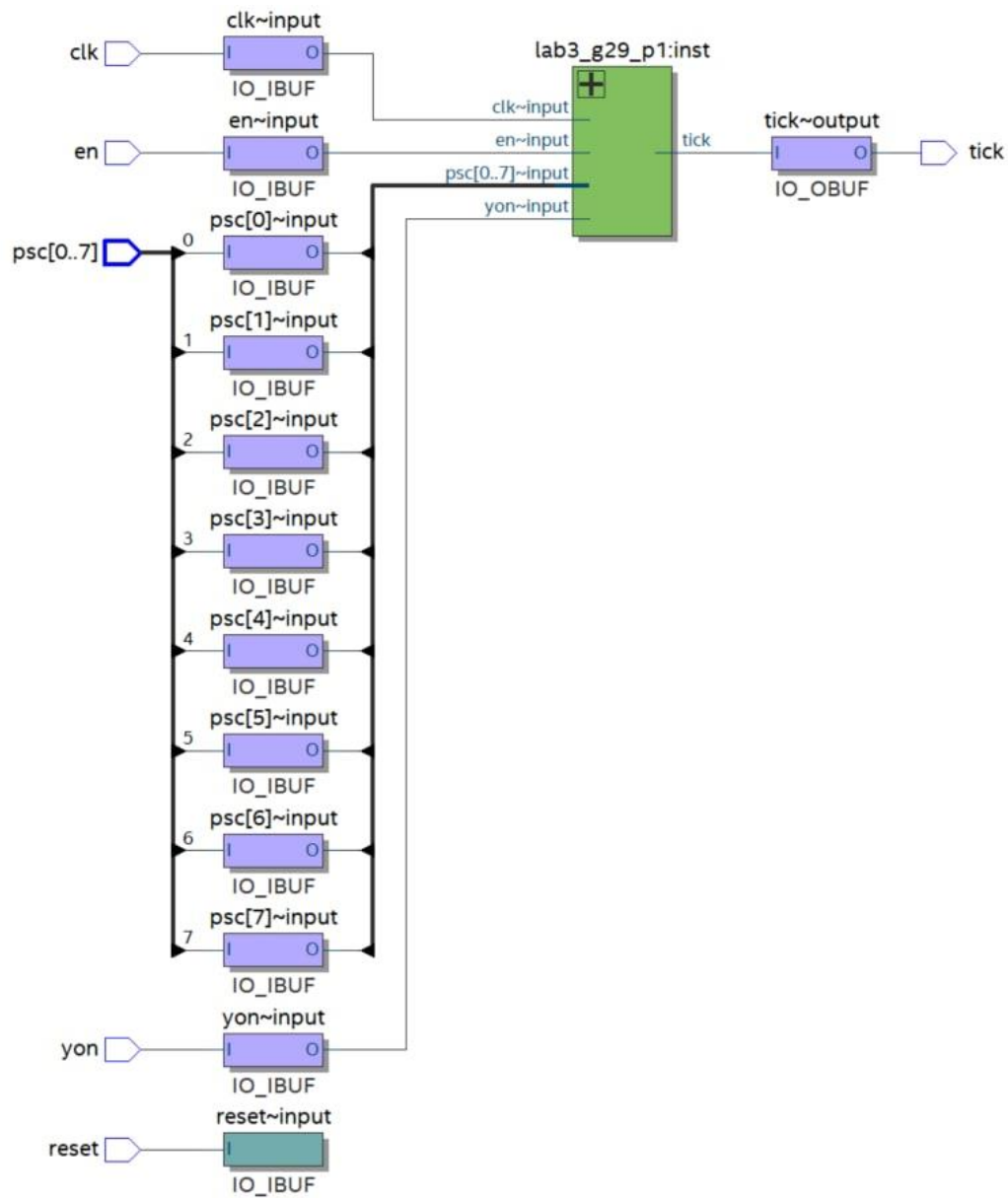
```
if (counter == 0)
```



Şekil 8 Quartus çıktısı



Şekil 9 RTL Şeması



Şekil 10 Quartus çıktısı - Post mapping

Analysis & Synthesis Resource Usage Summary		
<<Filter>>		
	Resource	Usage
1	Estimated Total logic elements	33
2		
3	Total combinational functions	33
4	▼ Logic element usage by number of LUT inputs	
1	-- 4 input functions	19
2	-- 3 input functions	10
3	-- <=2 input functions	4
5		
6	▼ Logic elements by mode	
1	-- normal mode	26
2	-- arithmetic mode	7
7		
8	▼ Total registers	9
1	-- Dedicated logic registers	9
2	-- I/O registers	0
9		
10	I/O pins	13
11		
12	Embedded Multiplier 9-bit elements	0
13		
14	Maximum fan-out node	yon~input
15	Maximum fan-out	21
16	Total fan-out	146
17	Average fan-out	2.15

Şekil 11 Quartus Resource Usage Summary

Analysis & Synthesis Resource Utilization by Entity						
<<Filter>>						
	Compilation Hierarchy Node	Combinational ALUTs	Dedicated Logic Registers	Memory Bits	UFM Blocks	DSP Elements
1	▼ lab3_g29_p1_top	33 (0)	9 (0)	0	0	0
1	lab3_g29_p1:inst	33 (33)	9 (9)	0	0	0

Şekil 12 Quartus Analysis & Synthesis Resource Utilization by Entity

by Entity							
DSP 9x9	DSP 18x18	Pins	Virtual Pins	ADC blocks	Full Hierarchy Name	Entity Name	Library Name
0	0	13	0	0	lab3_g29_p1_top	lab3_g29_p1_top	work
0	0	0	0	0	lab3_g29_p1_top lab3_g29_p1:inst	lab3_g29_p1	work

Şekil 13 Quartus çıktısı

Yazılan kodda hiçbir deęişiklik yapılmamıştır ancak bilinmeyen bir sebepten dolayı Quartus programı sürekli denemeler sonucunda bir anda çalışmış ve gerekli sonuçları vermiştir.

Sonuçlar ve Genel Yorumlar

Bu laboratuvar föyünde verilen up and down counter temel isterleri yerine getirebilecek ölçüde başarılı bir biçimde gerçekleştirilmiştir. Tasarlanan counter, ileri sayıp gerekli psc deęerinde tick = 1 yapabiliyor, psc sayısından başlayıp 0'a kadar geri sayıp 0 da Tick = 1 yapabiliyor, clock ve enable kontrolünü yapabiliyor, gerekli yerlerde 0'a veya psc deęerine dönüp baştan saymaya başlayabiliyor ve yön kontrolünü başarılı bir şekilde yapabiliyor.

Referanslar

[1] Intel® Quartus® Prime Software Suite. URL:

<https://www.intel.com.tr/content/www/tr/tr/software/programmable/quartus-prime/overview.html> Accessed: 21.02.2020

[2] Quartus Prime Introduction Using Schematic Designs. URL: ftp://ftp.intel.com/Pub/fpgaup/pub/Intel_Material/16.0/Tutorials/Schematic/Quartus_II_Introduction.pdf Accessed: 21.02.2020

[3] Harris, S. and Harris, D., 2015. Digital Design and Computer Architecture: ARM Edition. Morgan Kaufmann.

[4]

https://www.intel.com/content/www/us/en/programmable/quartushelp/13.0/mergedProjects/messages/evrfx_veri_procedural_assignment.html

[5] http://electronoobs.com/eng_circuitos_tut23_counter.php

[6] <https://github.com/fcayci/sv-digital-design/blob/master/rtl/counter.sv>

[7] <http://techmasterplus.com/verilog/verilog-counter.php?i=1>