



GEBZE TEKNİK ÜNİVERSİTESİ
ELEKTRONİK MÜHENDİSLİĞİ

ELM235

LOJİK DEVRE TASARIM LABORATUVARI

LAB 5 Deney Raporu

Sıralı Mantık Tasarım

Hazırlayanlar
1) 200102002031 – Beyza Duran
2) 200102002043 – Senanur Ağaç

TEORİK BİLGİLER

Sayıcı Devresi Nedir?

Flip-flopların uygun şekillerde bağlanmasıyla, sayıları saymak için kullanılması sayıcı ismi verilir. Sayıcı devreleri, giriş darbelerine bağlı olarak belirli bir durum dizisini tekrarlayan lojik devrelerdir. Sayıcılar genellikle saat sinyalinin flip-floplara uygulanmış şekline göre ikiye ayrılırlar. Bunlar ; **asenkron sayıcılar** ve **senkron sayıcılar**dır.

Senkron Sayıcılar

Sayma işlemi için kullanılan saat sinyali bütün flip-floplara eş zamanlı olarak aynı anda uygulanıyorsa bu tür sayıcılara da Senkron Sayıcılar denir. Burada saat girişi bütün flip-floplar için ortaktır. Bu sayede bütün flip-floplar birlikte tetiklenir.

Saat sinyalinin uygulanışıyla oluşan bu farklılık ile birlikte asenkron sayıcılar senkron sayıcılara göre daha yavaş çalışmaktadır. Bunun sebebi ise asenkron sayıcılarda flip-flopların birbirini tetiklemesidir.

PROBLEMLER

Problem 1 - Yukarı serbest sayıcı devresi

Bu problemde aktif edildiğinde (en sinyali) her clock rising edge de 0'dan verilen psc değerine kadar sayıp, her psc değerine ulaştığında tick sinyali oluşturan bir devre tasarlayacaksınız. Devrenin disable edilmesi counter değerinizi etkilemeyecek. psc sinyali 5 bitlik olarak 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.

```
module p1 (  
    input logic clk, reset, en,  
    input logic [4:0] psc,
```

output tick);

Problem 1 : TESTBENCH

```
module tb_lab5_g15_p1 ();

logic clk, reset, en;
logic [4:0] psc;
logic tick;
logic [4:0] count;

lab5_g15_p1 lab5(.clk(clk), .reset(reset), .en(en), .psc(psc),
.tick(tick), .count(count));

always begin
clk = 0; #5; clk = 1; #5;

end

initial begin
reset = 0; #100; reset = 1;
en=1;

end

initial begin
psc={5{1'b1}}; #1000;

$stop;

end

endmodule
```

Problem 1 : .sv DOSYASI

```

/*
Hazırlayanlar : Senanur Ağaç-Beyza Duran
*/
module lab5_g15_p1(                                     //devrenin input ve
output pinleri tanımlanmıştır
input logic clk, reset, en,
input logic [4:0] psc,
output logic [4:0] count,
output logic tick);

always_ff @(posedge clk)

begin
if (reset==0)                                     //eğer reset pini aktif ise sayaç
sıfırlanır
count <={5{1'b0}};
else if(en)                                       // Devrenin aktif olduğu durumda ;

begin
if (count==psc)

begin                                           //Psc ve sayaç aynı olduğu zaman tick
sinyali gönderilmiş

count<=0;
tick =1;

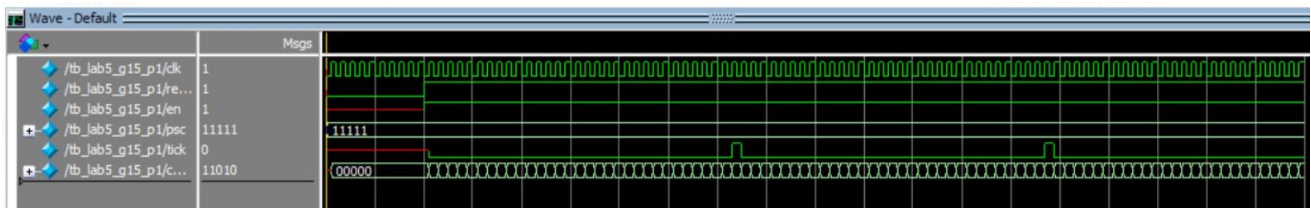
end

else

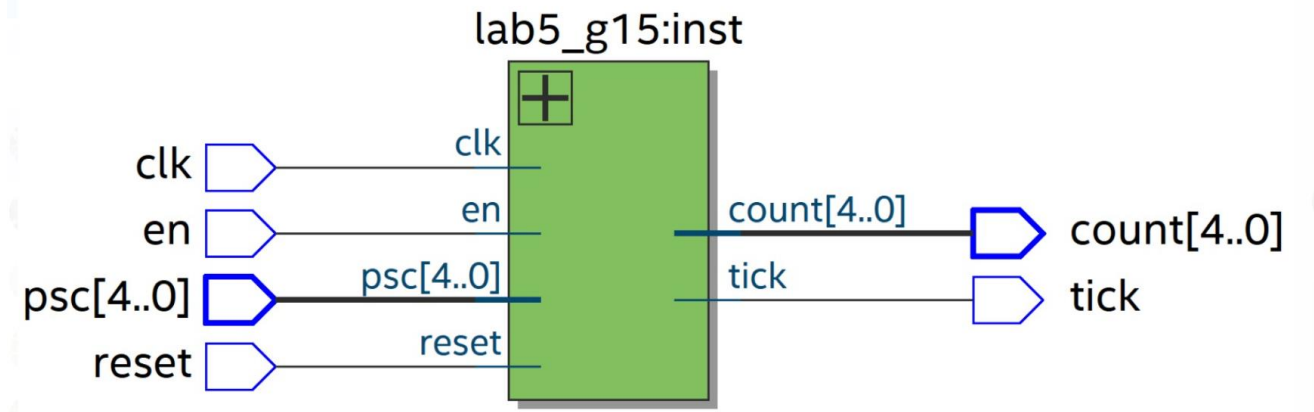
begin                                           // Clock 0->1 olduğunda sayaç 1 artmıştır.
count<=count+1'b1;
tick=0;

end
end
end
endmodule

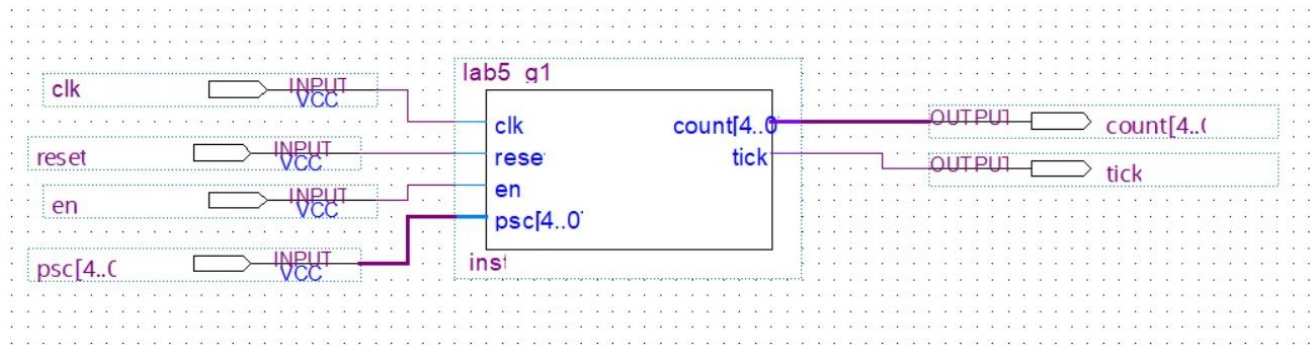
```

WAVE

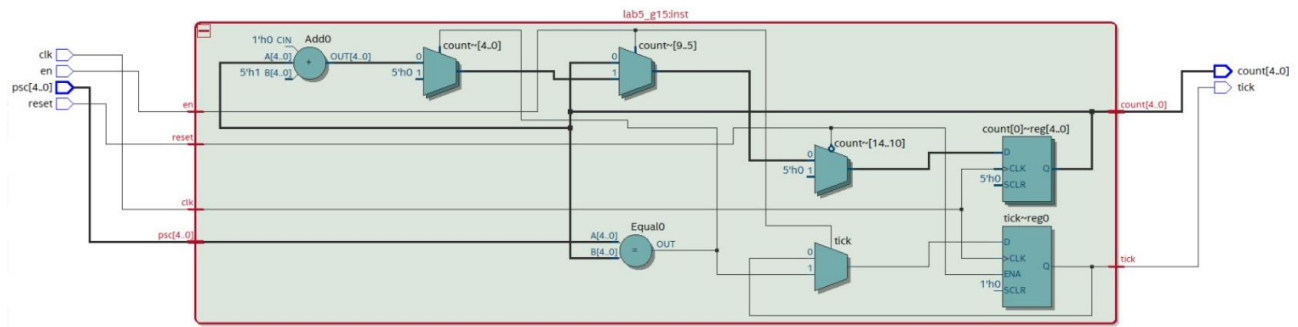
Şekil 1: Problem 1 için dalga şeması

RTL

Şekil 2: Problem 1 için RTL şeması



Şekil 3: Problem 1 için Quartus Prime şeması



Şekil 4: Problem 1 için ayrıntılı RTL şeması

Slow 1200mV 85C Model Fmax Summary

<<Filter>>

	Fmax	Restricted Fmax	Clock Name	Note
1	383.73 MHz	250.0 MHz	clk	limit due to minimum period restriction (max I/O toggle rate)

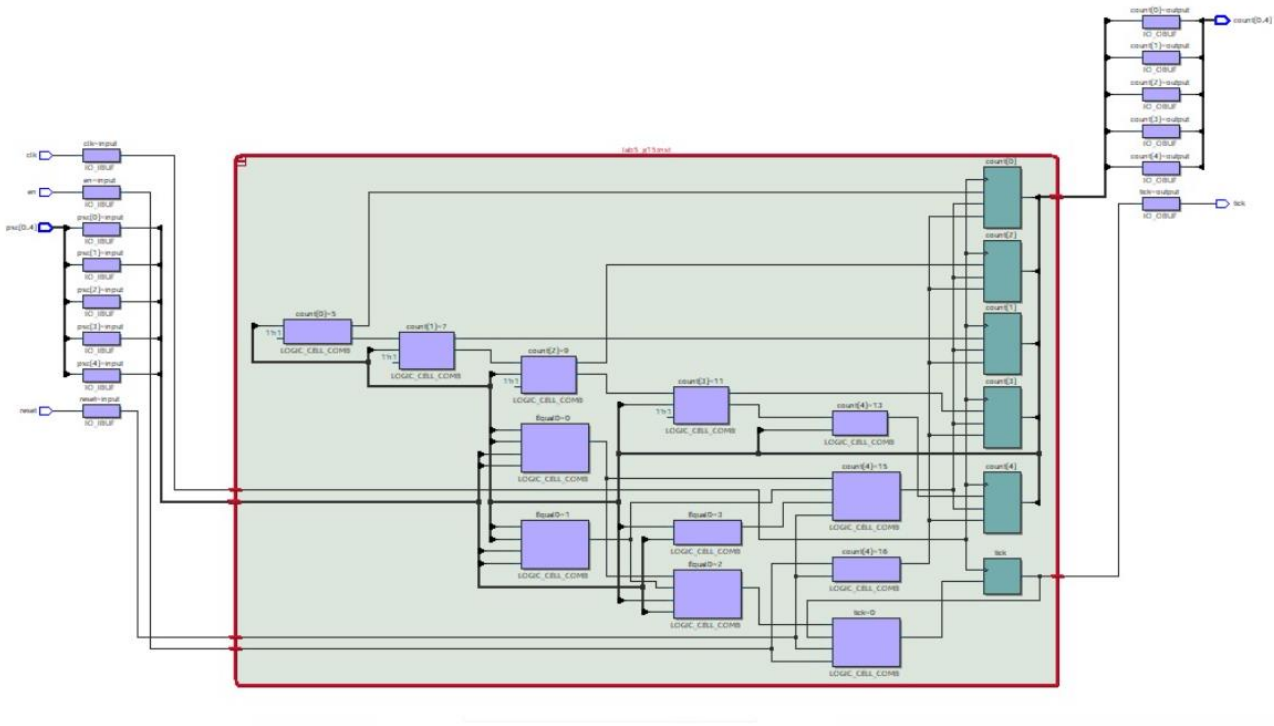
Slow 1200mV OC Model Fmax Summary

<<Filter>>				
	Fmax	Restricted Fmax	Clock Name	Note
1	414.94 MHz	250.0 MHz	clk	limit due to minimum period restriction (max I/O toggle rate)

Flow Summary

<<Filter>>	
Flow Status	Successful - Sun Apr 24 20:53:30 2022
Quartus Prime Version	19.1.0 Build 670 09/22/2019 SJ Lite Edition
Revision Name	examp_top
Top-level Entity Name	examp_top
Family	MAX 10
Device	10M08DAF484C8G
Timing Models	Final
Total logic elements	12
Total registers	6
Total pins	14
Total virtual pins	0
Total memory bits	0
Embedded Multiplier 9-bit elements	0
Total PLLs	0
UFM blocks	0
ADC blocks	0

Şekil 5: Problem 1 için akış şeması



Bu deneyde disable edilmesi counter değerini etkilememektedir. Psc sinyali 5 bitlik olarak alınmıştır. Ek olarak active-low sinyali eklendi. Resetin eklenmesiyle counterlar 0 oldu. Tick değeri 1 olunca sayı elde edildi. Bunlar dışında devre içinde 14 adet input/output pin kullanılmıştır. Fmax değeri 467.51 Mhz olarak bulunmuştur.

Problem 2 - Yavaşlatılabilir aşağı sayıcı tasarımı

Bu problemde, aktif edildiğinde verilen 16-bitlik reload değerinden 0 a kadar sayıp, 0 a ulaştığında done sinyali oluşturan bir devre tasarlayacaksınız. Ayrıca bir reset sinyali ekleyip, reset geldiğinde counterınızı 0 layınız. Bu devreyi tasarlarken, Problem 1 de tasarladığınız devreyi instantiate edip, counterınızı her tick geldiğinde düşürünüz.

1. Tasarımınızı yaparken senkron tasarım prensiplerine bağlı kalınız.
2. active-low reset kullanınız.
3. counter değerinizi, devre aktifse ve problem 1 deki devredeki tick sinyali geldiğinde düşürülecektir.
4. reload değerinizi 0 dan farklı ise ve devre aktif değilse reload değeri counterınıza atanacak.
5. Eğer devre aktif ise reload değerinin değişmesi counterınızı etkilemeyecek
6. Bu devreye verdiğiniz psc değeri Problem 1 de tasarladığınız devreye iletilecek
7. Devre aktifse ve counter 0 a ulaşmışsa, done sinyali verilecek
8. 0 a ulaştıktan sonra o anki reload değeri countera yüklenip aynı şekilde çalışmaya devam edecek.

Örnek olarak psc 5 ve reload değerinizi 16 ise devreyi aktif ettiğiniz zaman, her 5 cycle da bir tick sinyali oluşacak ve her tick sinyalinde counter değerinizi 16 dan bir düşecek. Dolayısıyla 80 clock cycle sonra $(16 \cdot 5)$ done sinyali görmeniz lazım. (+- 1 cycle farklılık gösterebilir)

```
module p2 (  
input logic clk, reset, en,  
input logic [ 4:0] psc,  
input logic [15:0] reload,  
output logic [15:0] cnt,  
output logic done  
);
```

Bu devreyi test etmek için oluşturacağınız testbench, farklı giriş kombinasyonlarını göz önünde bulundurmalıdır. Örnek olarak isterlerde belirtilen normal operasyon dışı reload değeri en sinyali aktif iken değişebilir. Bu ve bunun gibi durumda devrenin normal çalıştığını gözlemlemeniz gerekmektedir. Raporunuza testbench oluştururken düşündüğünüz senaryoları ve onlara karşı devrenin nasıl çalıştığını da ekleyiniz.

Problem 2 : TESTBENCH

```
`timescale 1ns/1ps  
module tb_lab5_g15_s2 ();  
  
logic clk, reset, en;  
logic [4:0] psc;  
logic [15:0] reload;  
logic tick, d;  
logic [15:0] cnt;  
logic [4:0] count;  
lab5_g15_s2 labb5(.clk(clk),  
  
.reset(reset), .en(en), .psc(psc),  
.tick(tick), .count(count),  
.reload(reload), .cnt(cnt), .d(d));  
  
always begin  
clk = 0; #5; clk = 1; #5;  
end  
  
initial begin  
reset = 0; #20; reset = 1;  
en=1;  
end  
  
initial begin  
psc={5{1'b1}};  
  
reload=16'b0000000000000001; #2000;  
$stop;  
  
end  
  
endmodule
```


Problem 2 : .sv DOSYASI

```
`timescale 1ns/1ps
module lab5_g15_s2(
input logic clk, reset, en,
input logic [4:0] psc,
output logic [4:0] count,
input logic [15:0] reload,

output logic [15:0] cnt,
output logic tick, done);

always_ff @(posedge clk)
begin

if (reset==0) begin
count <={5{1'b0}};
cnt<=0;

end

else if(en)
begin
if (count==psc)
begin
count<=0;
tick =1;
if(tick==1)begin

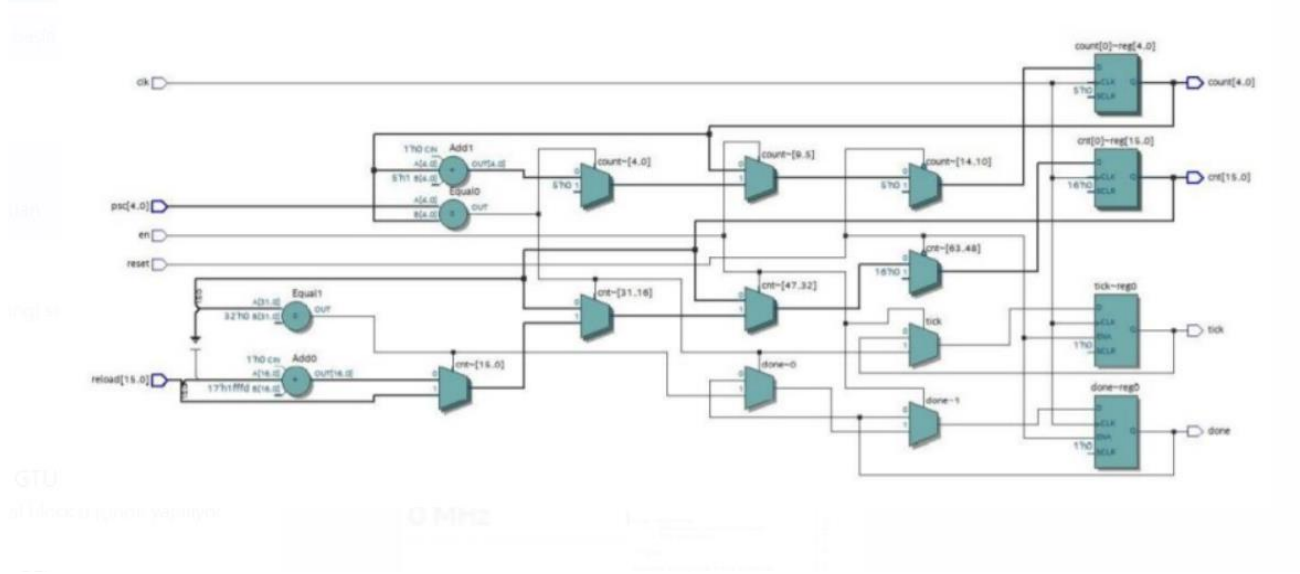
cnt=reload-1;
done<=0;
if(cnt==0) begin
cnt<=reload;
done = 1;

end

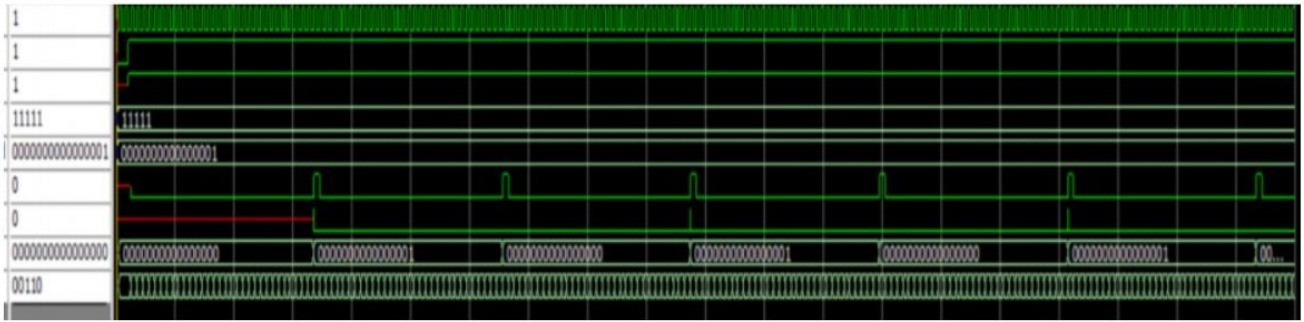
end

end

else
begin
count<=count+1'b1;
tick=0;
end
end
end
endmodule
```

RTL

Şekil 9: Problem 2 için ayrıntılı RTL şeması

WAVE

Şekil 10: Problem 2 için dalga şeması

	Fmax	Restricted Fmax	Clock Name	Note
1	263.16 MHz	250.0 MHz	clk	lim...te)

KAYNAKÇA

[Sayıcı Devresi Nedir ve Nasıl Yapılır? \(elektrikport.com\)](http://elektrikport.com)

[8-Bit Up-Down Counter \(asic-world.com\)](http://asic-world.com)

ELEKTRİK-ELEKTRONİK TEKNOLOJİSİ/SAYICI VE KAYDEDİCİ DEVRELERİ KİTABI