



GEBZE TEKNİK ÜNİVERSİTESİ
ELEKTRONİK MÜHENDİSLİĞİ

ELM 235

LOJİK DEVRE TASARIM LABORATUVARI

Sıralı Mantık Tasarım

LAB 0x3 Deney Raporu

Lojik Devreler ve Tasarım Laboratuvarı

Hazırlayanlar
1) 1801022024 – M. Cemal Eryiğit
2) 1801022077 – Burak Kamil Çiftci

Bu Labın Amacı

- Donanım tanıma dillerini (DTD) kullanarak devre tasarımı yapmak.
- Sıralı mantık devreleri oluşturabilmek.
- Devrelerin çalışma prensiplerine ve isterlere göre çalıştığını doğrulamak için senaryo oluşturup onları test edebilmek.

1. Giriş

Bu deney kapsamında sıralı mantık tasarımına uygun devreler oluşturması üzerinde çalışılmıştır. Ayrıca bir devrenin diğer devreyle nasıl senkronize edilebileceği üzerinde durulmuştur.

1.1 Teorik Araştırma

Counterlar ve çalışma prensiplerine dair araştırma yapıp gerekli bilgiler edinilmiştir. Yukarı ve aşağı sayıcı olmak üzere iki farklı türü öğrenilmiştir.

1.2. Deneyin Yapılışı

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.

```
/* Hazirlayanlar:
 * M.Cemal Eryigit
 * Burak Kamil Ciftci
 * ELM235 2021 Bahar Lab3 - Problem 1
 */

module lab3_g24_p1(
input logic clk, reset, en, yon,
input logic [7:0]psc,
output logic tick,
logic [7:0] count,
logic [7:0]cnt,
logic done
);

always_ff @ (posedge clk )
begin
if(!yon)
begin
if(en)
begin
if(psc==count)
begin
tick<=1;
end
else
begin
count<=count+1'b1;
tick<=0;
end
end
else if(!reset)
count <= 8'b00000000;
end
else if(yon)

begin

if(en==1)
begin
if(done==0)
begin
cnt<=psc;
done<=1;
end
```

```
begin

if(en==1)
begin
if(done==0)
begin
cnt<=psc;
done<=1;
end
```

```

        if(done==1 )
            cnt=cnt-1'b1;
            if(cnt==0 && done==1)
                begin
                    tick<=1;
                    done<=0;
                end
            end
        else if (!reset)
            cnt<=8'b00000000;

    end
end
endmodule

```

```

/* Hazirlayanlar:
* M.Cemal Eryigit
* Burak Kamil Ciftci
* ELM235 2021 Bahar Lab3 - Problem 1
*/
module tb_lab3_g24_p1();
    logic [7:0]count;
    logic [7:0] psc;
    logic clk,reset,en,tick,yon,done;

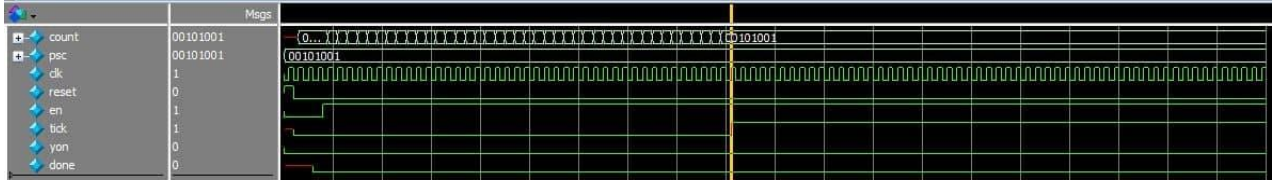
    lab3_g24_p1 dut0 ( .count(count), .yon(yon), .en(en), .clk(clk),
        .reset(reset), .psc(psc), .tick(tick), .done(done));

    initial begin
        en=0; reset =1; #10;
        reset =0; tick=0; #10;
        count =0; #10;
        done=0; #10;
        en=1;
    end
    always begin
        clk=0; #5;
        clk=1; #5;
    end
    always begin
        yon=1; psc <= 8'b00101001; #1000;

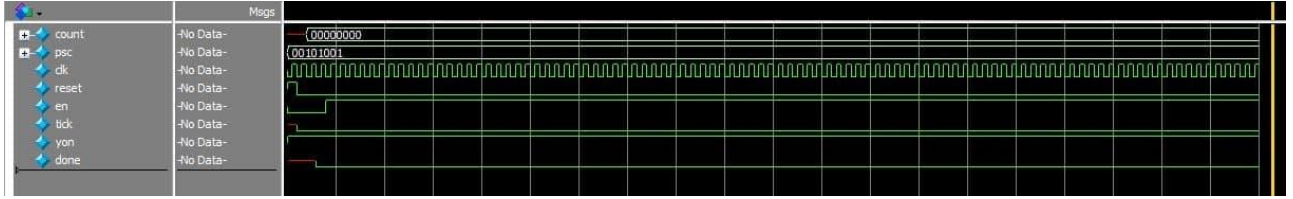
    $stop;
    end
endmodule

```

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





Şekil 1 Yukarı Sayma Sayacı

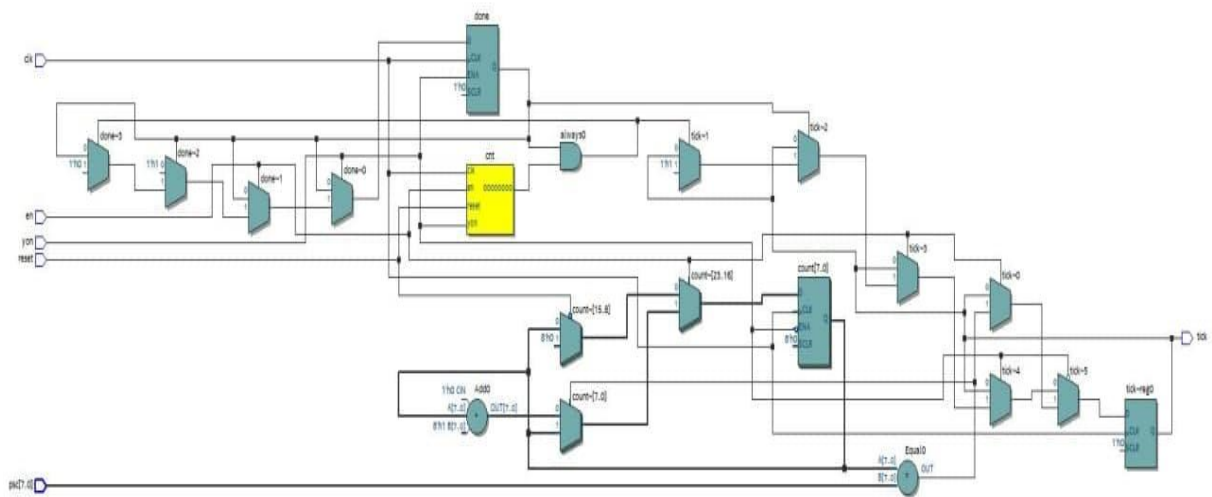


Şekil 2 Aşağı Sayma Sayacı

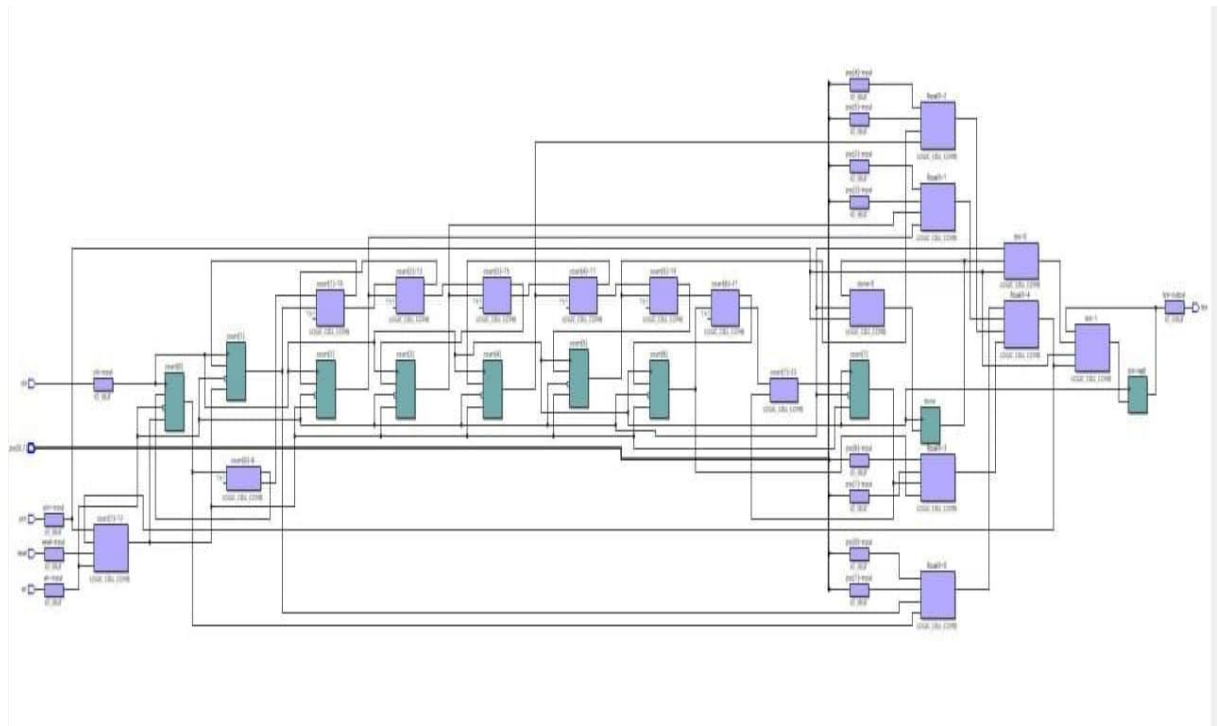
- C. Devreyi Quartus'da sentezleyerek devrenin ne kadar yer kapladığını (resource utilization report), sentezlenen RTL ve eşleştirme ardı devre şemalarını ekleyerek yorumlayınız.

Flow Summary						Analysis & Synthesis Resource Usage Summary																					
 <<Filter>>						 <<Filter>>																					
Flow Status		Successful - Fri Jul 02 01:01:28 2021						Resource				Usage															
Quartus Prime Version		20.1.1 Build 720 11/11/2020 SJ Lite Edition				1		Estimated Total logic elements				17															
Revision Name		lab3_g24_p1				2																					
Top-level Entity Name		lab3_g24_p1				3		Total combinational functions				17															
Family		MAX 10				4		▼ Logic element usage by number of LUT inputs																			
Device		10M50DAF484C6GES				1		-- 4 input functions				7															
Timing Models		Preliminary				2		-- 3 input functions				2															
Total logic elements		18 / 49,760 (< 1 %)				3		-- <=2 input functions				8															
Total registers		10				5																					
Total pins		13 / 360 (4 %)				6		▼ Logic elements by mode																			
Total virtual pins		0				1		-- normal mode				10															
Total memory bits		0 / 1,677,312 (0 %)				2		-- arithmetic mode				7															
Embedded Multiplier 9-bit elements		0 / 288 (0 %)				7																					
Total PLLs		0 / 4 (0 %)				8		▼ Total registers				10															
UFM blocks		0 / 1 (0 %)				1		-- Dedicated logic registers				10															
ADC blocks		0 / 2 (0 %)				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				en~input															
						15		Maximum fan-out				11															
						16		Total fan-out				99															
						17		Average fan-out				1.87															
Compilation Hierarchy Node		Combinational ALUTs		Dedicated Logic Registers		Memory Bits		UFM Blocks		DSP Elements		DSP 9x9		DSP 18x18		Pins		Virtual Pins		ADC blocks		Full Hierarchy Name		Entity Name		Library Name	

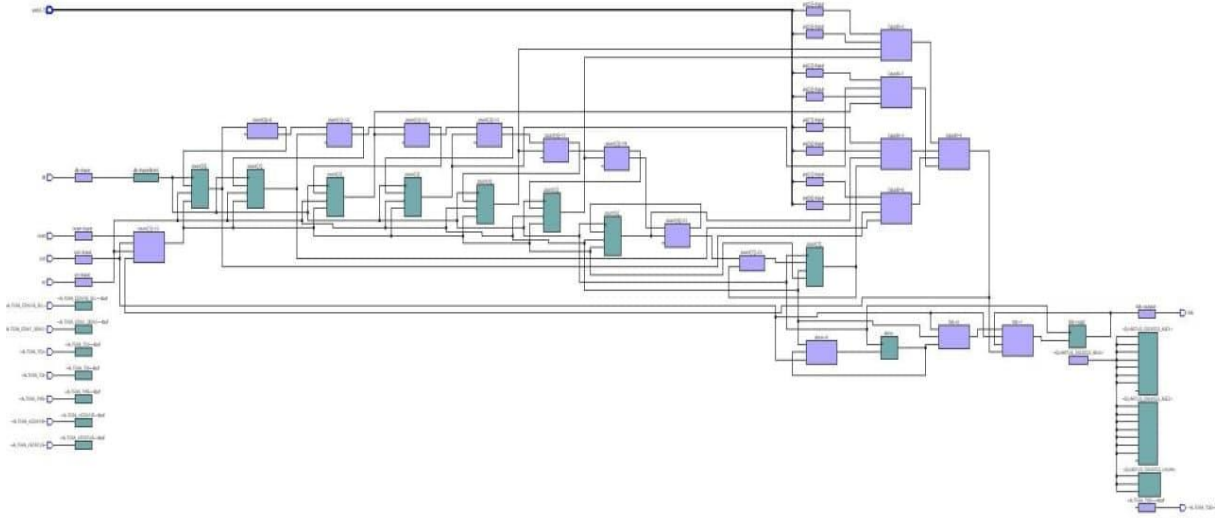
Şekil 3 Analiz ve Kaynak Sentez Özetleri



Şekil 4 RTL Devresi



Şekil 5.a Techmap Şeması



Şekil 6.b Techmap Şeması

1.3 Deneyin Yorumu

Deneyin bize öğretmek istediği sayıcı devreler hakkında ön araştırma yapıldı. Deney yapılırken birlaç sorunla karşılaşıldı. Karşılaştığımız sorunlara örnek olarak, serbest sayıcı devrede “psc” değerinden başlayıp 0’a kadar sayma işlemi tam olarak gerçekleşmemektedir. Çeşitli yöntemler denenmesine rağmen çözüme ulaşamadık. Denenen çözümler arasında atama yapılan değerleri reg ile tanımlayıp register yapma veya if bloğu yerine case ile daha doğru sayma işlemi yapabilme gibi varyasyonlar denendi. Ekstra done ve cnt gibi değer tutan logic tanımlamalar yapılarak sonuca ulaşılmaya çalışıldı. Bu uğraşlar sonucunda teslim süresinin gelmesinden ötürü aşağı sayaç tamamlanamadı.

Dizaynın quartus sentezi sonucunda 17 adet logic blok kullanıldığını, 10 adet register kullanıldığını, 18 adet logic element kullanıldığını ve 13 adette pin kullanıldığını sonucuna ulaşıldı.

4. Referanslar

[1] Ders Slaytları

[2] Harris and D.Harris, Digital Design and Computer Architecture: ARM Edition, 1st edition. Morgan Kaufmann, 2015.