



GEBZE TEKNİK ÜNİVERSİTESİ
ELEKTRONİK MÜHENDİSLİĞİ

ELM235

LOJİK DEVRE TASARIM LABORATUVARI

LAB 0x5 Deney Raporu

Lojik Devreler ve Tasarım Laboratuvarı Dersi

Hazırlayanlar
1) 171024086-Berat KIZILARMUT
2) 1801022037-Ömer Emre POLAT

1. Giriş

Bu deneyde SystemVerilog HDL dilini kullanarak sıralı mantık devreleri oluşturacağız. Tasarlanan bir devreyi başka bir devrede alt parça olarak kullanacağız. Ardından bu oluşturulan devreyi, devrenin istenilen çalışma senaryolarına göre sınavacak, devreye uyumlu ve kaliteli test yapabilecek bir testbench hazırlayacağız.

Bu tasarlayacağımız devreler Counter devreleri olacak. Belli kriterlere uyacak şekilde, belli bit büyüklüklerinde sayma işlemleri gerçekleştirecekler.

2. Problemler

2.1. Problem I – Yukarı Serbest Sayıcı Devresi

A. HDL Yazılımı

Counter devresi SystemVerilog dilinde, posedge clk, negedge reset hassasiyet kuralında tanımlandı. If statementları, reset aktif olduğunda direkt countı sıfırlayacak şekilde, değilse count'a bir artı ekleyecek şekilde ayarlandı. Count ile Psc eşit olduğu durumlarda Tick verecek şekilde yazıldı.

```
/* lab5_g41_p1.sv
 * Hazırlayanlar: Berat Kızıllarmut, Ömer Emre Polat
 * Notlar: Counter Devresi */
module lab5_g41_p1 (
input logic clk, reset, en,
input logic [4:0] psc,
output tick);
logic [4:0] count = 5'b1;
always_ff @(posedge clk, negedge reset)
begin
    if(!reset)
    begin
        count <= 5'b1;
    end
    else if(en)
    begin
        count <= count + 1'b1;
        if(count == psc)
        begin
            count <= 5'b1;
        end
    end
end
end
assign tick = (count == psc);
endmodule
```

B. Testbench Kurulumu

Kurulan devreyi sınamak için duruma ve devreye uygun bir testbench devresi kuruldu. Devre explicit instantiation şekilde çağırıldı. 10 ps periyoda sahip bir Clock sinyali tanımlandı. Logic inputlar sıfırlandı. Farklı kombinasyonlarda inputlar sınandı. Testbench devreyi istenildiği gibi doğru bir şekilde sınavı.

```

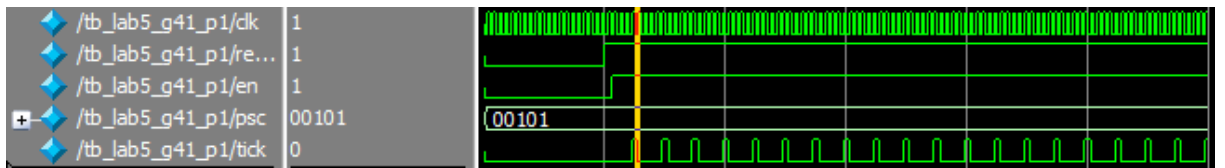
/* tb_lab5_g41_p1.sv
 * Hazırlayanlar: Berat Kızıllarmut, Ömer Emre Polat
 * Notlar: Counter devrensinin testbenchi */
`timescale 1ns/1ps
module tb_lab5_g41_p1();
logic clk, reset, en;
logic [4:0] psc;
logic tick;
lab5_g41_p1 dut0(.clk(clk), .reset(reset), .en(en), .psc(psc),
.tick(tick));
always begin
clk = 0; #5; clk = 1; #5;
end

initial begin
en = 0; reset = 0; #200;
reset = 1; #10; en = 1;
end
initial begin
psc <= 5'd5;
#1200;
$stop;
end
endmodule

```

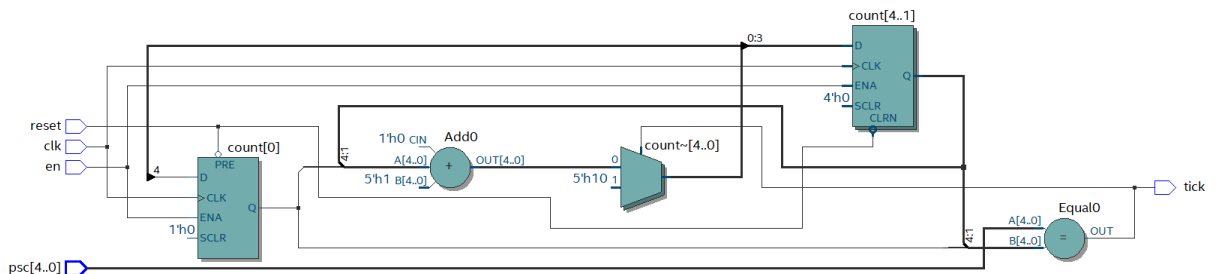
C. Waveshape

Testbench sonuçlarını gösteren dalga formu. Devre beklenildiği sonuçları vermiştir. istenilen tick sinyalizimiz iç değişken olan counter psc'ye ulaştığında 1 gönderiyor ve ulaştıktan sonraki clock cycle da 1 arttırıldığı için bir sonraki clockda 1 yollamıyor. Yapılan bu 1 arttırma işlemi counter değişkeni overflow olup başa dönene kadar devam etmekte. Başa dönen counter yeniden psc değerine ulaştığında tick sinyalizimiz 1 olmakta.



D. Problem I Analiz ve Yorumlar

Systemverilog yazılımımız beklenildiği gibi çalışmıştır. Dalga formunda da gözlenildiği gibi, gerektiği yerlerde tick sinyalini vermiştir. Reset active low olarak çalışmaktadır. Latch oluşmaması için reset sinyali always_ff bloğuna negedge reset şeklinde eklenmişti. Bu önlemler sayesinde devre beklenildiği gibi sentezlendi.



	Resource	Usage
1	Estimated Total logic elements	9
2		
3	Total combinational functions	9
4	▼ Logic element usage by number of LUT inputs	
1	-- 4 input functions	3
2	-- 3 input functions	0
3	-- <=2 input functions	6
5		
6	▼ Logic elements by mode	
1	-- normal mode	5
2	-- arithmetic mode	4
7		
8	▼ Total registers	5
1	-- Dedicated logic registers	5
2	-- I/O registers	0
9		
10	I/O pins	9
11		
12	Embedded Multiplier 9-bit elements	0
13		
14	Maximum fan-out node	Equal0~2
15	Maximum fan-out	6
16	Total fan-out	61
17	Average fan-out	1.91

Devremizde 3 adet 4 girişli, 6 adet iki ve daha az girişli lojik ünite kullanılmıştır. Devrede 9 ALUT bulunmasına rağmen, 5 Logic Register kullanılmaktadır. Devremizin Fmax'ı 375.09 MHz çıkmıştır. Restricted Fmax'imi 250 MHz gözükmemektedir, bunun sebebi devrenin sentezlendiği kartın input output değiştirme sıklığından dolayı limitlenmesidir.

	Compilation Hierarchy Node	Combinational ALUTs	Dedicated Logic Registers
1	lab5_g41_p1	9 (9)	5 (5)

	Fmax	Restricted Fmax	Clock Name	Note
1	375.09 MHz	250.0 MHz	clk	limit due to minimum period restriction (max I/O toggle rate)

2.2 Problem II – Yavaşlatılabilir aşağı sayıcı tasarımı

A. HDL Yazılımı

Bir önceki adımdaki gibi bu Counter devresi SystemVerilog dilinde, posedge clk, negedge reset hassasiyet kuralında tanımlandı. Bir önceki devre instantiate edilerek kullanıldı. Laboratuvar föyünde istenilen fonksiyonlar ve durumlar if statementları ile oluşturuldu.

```

/* lab5_g41_p2.sv
 * Hazırlayanlar: Berat Kızıllarmut, Ömer Emre Polat
 * Notlar: Yavaşlatılabilir aşağı sayıcı tasarımı */
module lab5_g41_p2 (
    input logic clk, reset, en,
    input logic [4:0] psc,
    input logic [15:0] reload,
    output logic [15:0] cnt,
    output logic done
);

```

```

        logic tick;
        lab5_g41_p1 c1(.clk(clk), .reset(reset), .en(en), .psc(psc),
        .tick(tick));
        always_ff @(posedge clk, negedge reset)
        begin
            if(!reset)
            begin
                cnt <= 16'b0;
            end
            else if(en & tick)
            begin
                cnt <= cnt - 1'b1;
            end
            else if(!en & (reload != 16'b0))
            begin
                cnt <= reload;
            end
            else if(en & (cnt == 16'b0))
            begin
                done <= 1'b1;
                cnt <= reload;
            end
        end
    end
endmodule

```

B. Testbench Kurulumu

Devreyi sınamak için bize sunulan aşağıdaki testbench devresi kullanıldı.

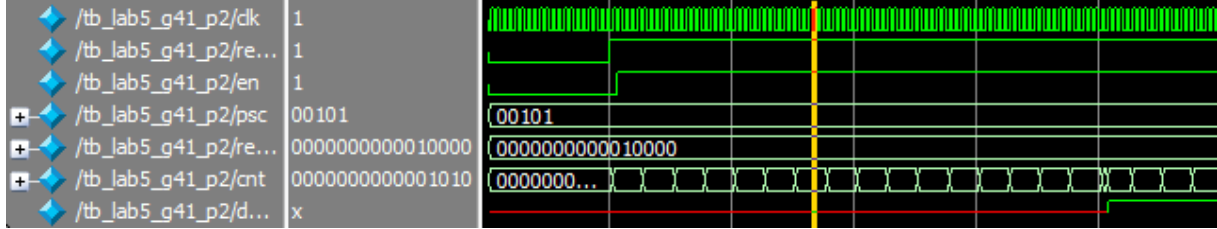
```

/* tb_lab5_g41_p2.sv
 * Hazırlayanlar: Berat Kızıllarmut
 * Notlar: Half adder yardımı ile fulladder tasarımı */
`timescale 1ns/1ps
module tb_lab5_g41_p2();
    logic clk, reset, en;
    logic [4:0] psc;
    logic [15:0] reload;
    logic [15:0] cnt;
    logic done;
    lab5_g41_p2 dut0(.clk(clk), .reset(reset), .en(en),
    .psc(psc), .reload(reload), .cnt(cnt), .done(done));
    always begin
        clk = 0; #5; clk = 1; #5;
    end
    initial begin
        en = 0; reset = 0; #200;
        reset = 1; #10; en = 1;
    end
    initial begin
        psc <= 5'd5;
        reload <= 16'd16;
        #1200;
        $stop;
    end
end
endmodule

```

C. Waveshape

İlk devremizden aldığımız tick sinyali ile her tick sinyalinde devremizin içindeki count değişkenini birer azaltmaktayız. Bu değerimiz sıfıra ulaştığında ise done sinyalimizi vermekteyiz.



D. Problem II Analiz ve Yorumlar

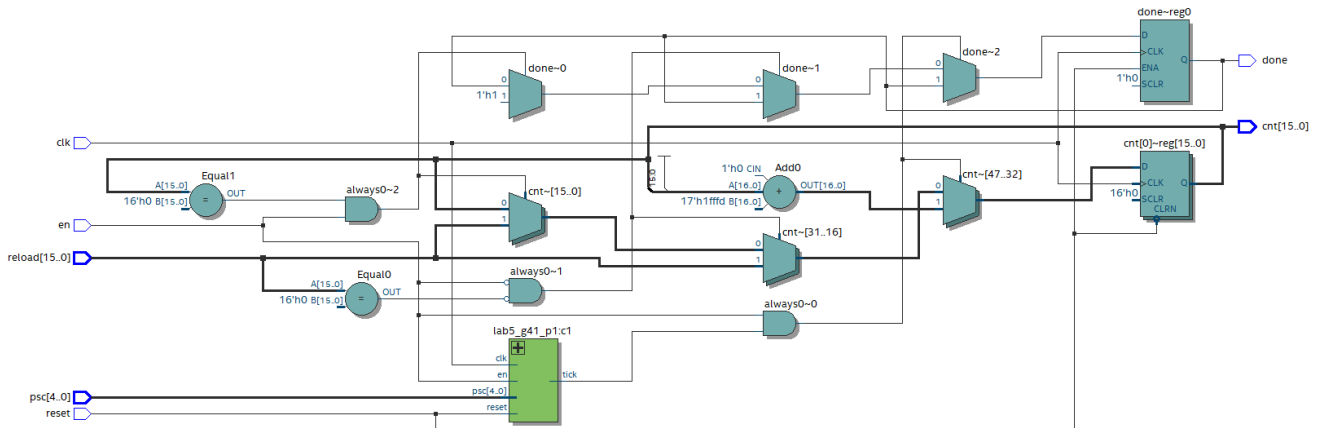
	Resource	Usage
1	Estimated Total logic elements	37
2		
3	Total combinational functions	37
4	Logic element usage by number of LUT inputs	
1	-- 4 input functions	14
2	-- 3 input functions	0
3	-- <=2 input functions	23
5		
6	Logic elements by mode	
1	-- normal mode	18
2	-- arithmetic mode	19
7		
8	Total registers	21
1	-- Dedicated logic registers	21
2	-- I/O registers	0
9		
10	I/O pins	41
11		
12	Embedded Multiplier 9-bit elements	0
13		
14	Maximum fan-out node	clk~input
15	Maximum fan-out	21
16	Total fan-out	281
17	Average fan-out	2.01

Yavaşlatılabilir aşağı sayıcı devremiz, bir önceki devrenin sayma özelliğini kullanarak, bir birim zaman bekleme ile ikinci devrede işlem yapılabilmesini sağlamaktadır. Kısacası Problem1'deki devremiz bizim için bir zamanlayıcı yardımcısı olmaktadır.

Devremizde toplam 14 adet 4 girişli, 23 adet iki veya daha az girişli lojik element bulunmaktadır.

Devrenimizin Clock Speedi 261.44 MHz'dir. Restricted Fmax bir önceki problemde de olduğu gibi, FPGA boardundan kaynaklı olarak 250 MHz çıkmıştır.

	Fmax	Restricted Fmax	Clock Name
1	261.44 MHz	250.0 MHz	clk



3. Genel Yorumlar ve Kazanımlar

Bu deneyde counter tasarlamayı, farklı işlevli counterlar tasarlamayı ve counterların birbirinden faydalanmasını öğrendik. Birden fazla counterı, structural design ve sequential design kurallarına uyarak tasarladık.

Testbench hazırlanışında birden fazla initial bloğu kullanarak aynı anda farklı değişkenleri değiştirmeyi öğrendik.

4. Kaynaklar

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