

# ddApp-10

## Sayısal Tasarım ve FPGA Uygulamaları

Amaç  
Kazanımlar  
Çalışma Prensibi  
Kodun Açıklaması  
Deneyin Yapılışı  
Telif Hakkı

## Clock Bölme Uygulaması

<b>Durum</b>	Tamamlandı <input checked="" type="checkbox"/>
<b>Güncelleme</b>	9 Kasım 2023
<b>Hazırlayan</b>	Emre İŞSEVER
<b>E-Posta</b>	emre.issever@fpgaturkey.com info@beti.com.tr

## Amaç

Bu doküman Clock Divider mantığını göstermek amacıyla 1 Hz frekansla LED yakma projesi anlatmaktadır.

## Kazanımlar

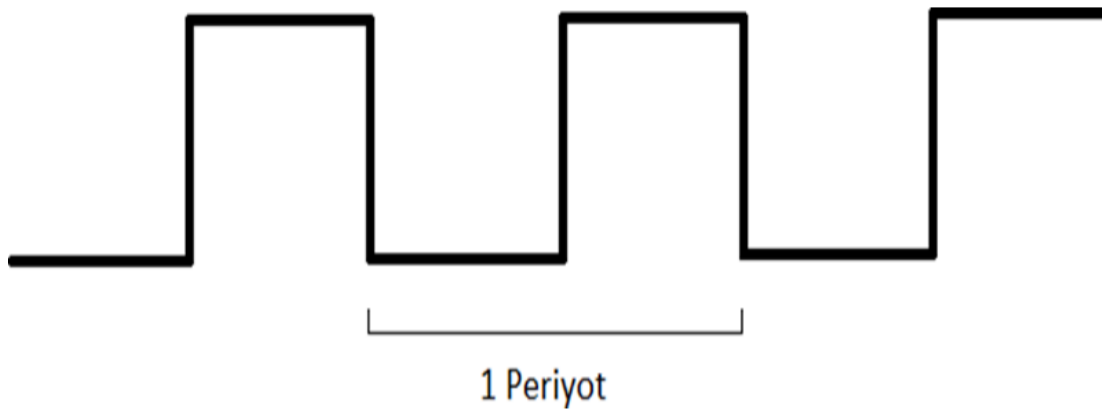
Clock Bölmenin temel prensipleri, yüksek frekanslı sinyalleri istenilen frekans seviyesine düşürme pratiği.

## Uygulamanın Çalışma Prensibi

Clock Divider yapabilmek FPGA uygulamaları için çok önemlidir. Her FPGA tipinin kendi clock frekansı vardır. Örneğin, Baysy3'ün clock hızı 100 MHz'dir. Bu, clock sinyalinin 1 saniyede 100,000,000 kere '1'den '0'a geçiş yaptığı anlamına gelir. Aynı zamanda clock sinyalinin '0' dan '1'e ya da '1'den '0'a geçişin 0.00000001 saniyede gerçekleştiği anlamına gelir. Eğer bir uygulamada farklı bir frekans gerekirse, bu clock sinyalini bölmek gerekmektedir. Sayısal elektronik sistemlerinin büyük bir çoğunluğu senkronize sistemlerdir. Senkronize olması, bütün devre elemanlarının bir clock sinyaline göre çalışması anlamına gelmektedir.

Hatırlatma: 100 Hz = saniyede 100 periyot = periyot süresi 1/100 saniye

Bu proje 1Hz periyodu olan bir sinyal yaratıp, board üzerindeki bir LED'i 1 Hz periyotla yakmayı amaçlamaktadır.



1Hz frekansı olan bir sinyal yaratmak için 0,5 saniye LOW ve 0,5 saniye HIGH olan bir dalga yapmak gerekmektedir. Bu sayede periyodu 1 saniye olan bir dalga oluşmaktadır.

Basys3'ün 100MHz clock'undan 1 saniye yapabilmek için kaç tane periyot gerektiği belirlenmelidir. 100MHz clock sinyaliyle bir periyot 0.00000001 saniye olduğu için, 0,5 saniye için gereken periyot sayısı:

$(0.5 \text{ saniye}) / (0.00000001 \text{ saniye}) = 50,000,000$

Bulunan sayı iki tabanına çevrilmelidir:

$100,000,000_{10} = 10111110101111000010000000_2$

Bir saniye elde edebilmek için, 100MHz clock frekansı ile 50,000,000'e kadar saymak gerekmektedir. Bu yarım saniyedir. Sayaç bu sayıya geldiğinde LED'in durumu değiştirilmelidir. Örneğin LED '0' durumundaysa '1', '1' durumundaysa '0' durumuna getirilmelidir. Bu sayede yurakıdaki şekilde verilmiş sinyal örneği yaratılmış olur. Bu mantık kodala şu şekilde yazılır:

## Kodun Açıklaması

*Bu doküman içerisindeki kodlar Vivado 2020.03 üzerinde gerçekleştirilip test edilmiştir.*

Verilen VHDL kodu, FPGA üzerinde bir clock bölme işlemi gerçekleştiren basit bir dijital devre tasarımıdır.

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL; --toplama çıkarma işlemi yapılması için gereken library
use IEEE.STD_LOGIC_UNSIGNED.ALL;

entity clockdivider is
    Port ( in_100MHz: in std_logic;
          led: out std_logic);
end clockdivider;

architecture Behavioral of clockdivider is
    signal clk_1Hz: std_logic;
    signal counter: std_logic_vector(28 downto 0);
    begin
        process(in_100MHz)
        begin
            if rising_edge(in_100MHz) then
                if counter < "0101111010111100001000000000" then -- 100.000.000'in 2 tabanlı
                    counter <= counter + 1; -- counter 100.000.000'e varana kadar counter'ı say
                else
                    clk_1Hz <= not clk_1Hz; -- eğer counter 100.000.000 olursa clk_1Hz'i değiştir
                end if;
            end if;
        end process;
    end;
end;
```

Projeyi İndir

## Deneyin Yapılışı

- Kişisel bilgisayarınıza Proje Dosyasını indiriniz.
- İndirdiğiniz '.zip' dosyasını, belirteceğiniz bir çalışma klasöre ayıklayınız.
- Dosyaların içindeki '.xpr' Vivado Project File dosyasına çift tıklayarak veya Vivado arayüzündeki üst panelden 'File > Open' yolunu takip ederek açınız.
- Projeniz açıldığında proje ekranının sağ üst köşesinde 'write\_bitstream\_Complete' ve  sembolünü görmelisiniz.
- Basys3 kartının üzerindeki güç anahtarını açarak, 'Power' ledinin yandığına emin olun.
- Vivado'da sol taraftaki 'Flow Navigator' penceresinden 'Open Hardware Manager' altındaki 'Open Target > Auto Connect' yolunu takip edin. Bu cihazınız ile FPGA kartı arasındaki bağlantıyı sağlayacaktır.
- Cihazı programlamak için 'Open Hardware Manager' altındaki 'Program Device' butonuna tıklayın.

## Telif Hakkı

Bu doküman, **Beti Elektronik San. ve Tic. Ltd. Şti**'nin fikri mülkiyetine tabidir ve firmanın izni olmadan kopyalanması, çoğaltılması, dağıtılması veya yeniden yayınlanması yasaktır. Kitabın herhangi bir bölümünün veya içeriğinin izinsiz kullanılması, telif hakkı ihlali anlamına gelir ve yasal işlemlere yol açabilir. Dokümanın içeriği veya herhangi bir kısmı kullanılacaksa, yazılı izin alınmalıdır. İzin talepleri için info@beti.com.tr ile iletişime geçiniz. Makul alıntılar yapılabilir, ancak bu alıntılar mutlaka kaynak gösterme ile birlikte sunulmalıdır. Kaynak göstermeden yapılan alıntılar veya içerik kullanımları, yine telif hakkı ihlali olarak kabul edilir.