

ddApp-10

Sayısal Tasarım ve FPGA Uygulamaları

Amaç
Kazanımlar
Çalışma Prensibi
Kodun Açıklaması
Deneyin Yapılışı
Telif Hakkı

Led - Switch Uygulaması

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

Amaç

Basys3 FPGA kartı üzerindeki anahtarlar (switch) ile LED'leri (ışık yayan diyot) kontrol edebilmek, bu işlemi yaparken de FPGA'nın nasıl programlanacağını ve FPGA içindeki lojik blokların kullanımını öğrenmek. Bu uygulama ile dijital devre tasarımının temel prensiplerini kavrama ve VHDL (VHSIC Hardware Description Language) programlama dilinin temellerini öğrenme amaçlanmaktadır.

Kazanımlar

- Basys3 FPGA kartı hakkında temel bilgi edinme.
- VHDL programlama dilinin temel yapılarını ve kullanımını öğrenme.
- Basys3 FPGA içindeki I/O pin yapılandırmasını (Constraints) anlama ve uygulama.
- Basit bir dijital devre tasarımını gerçekleştirme ve simüle etme.

Uygulamanın Çalışma Prensibi

Bu uygulamada, Basys3 FPGA kartının üzerinde yer alan 16 adet switch (anahtar) ve 16 adet LED'in her biri, kartın üzerinde fiziksel olarak yerleşiktir; fakat switch'ler ile LED'ler arasında doğrudan elektriksel bir bağlantı bulunmamaktadır. Bunun yerine, switch'lerden alınan giriş bilgileri ve LED'lere gönderilecek çıkış sinyalleri, FPGA içindeki programlanabilir lojik bloklar tarafından işlenir.

FPGA, Field-Programmable Gate Array (Alan Programlanabilir Kapı Dizisi) kısaltmasından gelir ve içerisindeki lojik kapılar kullanıcının ihtiyacına göre programlanabilir özelliğe sahiptir. Bu uygulama kapsamında, switch'ler FPGA'ye bağlıdır ve kullanıcının bu switch'leri açıp kapamasına bağlı olarak FPGA içindeki lojik devreler, switch'lerin durumunu okuyup karşılık gelen LED'lere uygun sinyalleri gönderir.

Örneğin, kullanıcı bir switch'i 'açık' (mantıksal '1') konuma getirdiğinde, FPGA içindeki ilgili giriş pinine bu durum okunur. FPGA üzerine yüklenen VHDL programı, bu bilgiyi alır ve programda tanımlanan işlemleri uygulayarak, ilgili LED'i 'yanacak' (mantıksal '1') şekilde çıkış pinine sinyal gönderir. Benzer şekilde, switch 'kapalı' (mantıksal '0') olduğunda, ilgili LED söner (mantıksal '0').

Bu süreçte önemli olan, switch'lerin ve LED'lerin her ikisinin de doğrudan FPGA'ye bağlı olması ve aralarındaki bu etkileşimin tamamen FPGA üzerinde yüklenen VHDL programı ile kontrol edilmesidir. Bu durum, FPGA'nın çok yönlülüğünü ve programlanabilir yapılar kullanarak çeşitli dijital devreler tasarlayabilme yeteneğini gösterir. Bu yaklaşım sayesinde, FPGA üzerindeki herhangi bir giriş-çıkış pininin işlevi, yüklenen program ile tamamen yeniden tanımlanabilir.

Kodun Açıklaması

[Bu doküman içerisindeki kodlar Vivado 2020.03 üzerinde gerçekleştirilip test edilmiştir.](#)

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

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;

entity top is
    Port (
        switch : in  std_logic_vector (15 downto 0);
        led     : out std_logic_vector (15 downto 0));
end top;
architecture Behavioral of top is
begin
    led <= switch;
end Behavioral;
```

[Projeyi İndir](#)

Bu VHDL kod örneği, dijital bir devre tasarımının temel elemanlarını tanımlamakta ve VHDL'nin donanım tanımlama kabiliyetlerini göstermektedir. VHDL, donanımın davranışsal özelliklerini, işlevselliğini ve etkileşimlerini betimlemek için tasarlanmış bir dildir. Bu dil, mantıksal işlemleri, veri akışını ve sistem zamanlamasını, yüksek seviyede bir soyutlama ile ifade etmeye olanak tanır.

Kodda tanımlanan entity, top ismiyle belirtilen ve bir FPGA devresinin dış dünya ile olan arayüzünü simgeler. Burada, devrenin giriş ve çıkış portları tanımlanmıştır: switch ve led. Her iki port da 16 bit genişliğinde std_logic_vector veri türünü kullanarak, çoklu bit bilgisi taşıyabilir şekilde tanımlanmıştır.

architecture Behavioral of top bölümü ise, tanımlanan bu `entity'nin içsel davranışını açıklar. VHDL kodunun bu kısmı, switch portundan alınan verinin nasıl işleneceğini ve led portuna nasıl iletilmesi gerektiğini betimler. Bu örnekteki davranışsal tanım, son derece basit bir veri akışı modeli kullanarak, giriş portundan okunan veriyi doğrudan çıkış portuna aktarır. Bu akış, led <= switch; ifadesiyle temsil edilmiştir ve bu atama, her bir switch'in durumunu doğrudan ilgili LED'e yansıtır.

Bu uygulamanın amacı, VHDL ile donanım betimleme ve FPGA'lerin programlanması konusunda temel bir giriş sağlamaktır. VHDL dilinin, donanımın lojik ve işlevsel özelliklerini nasıl modelleyebileceğini ve FPGA içindeki fiziksel donanım kaynaklarının nasıl kontrol edilebileceğini anlama açısından, bu örnek kod, basit ancak etkili bir öğrenme aracıdır.

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_Complate'** 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.