

Mantıksal Sistem Tasarımı – BLM 201

Hafta 3: Kombinasyonel Lojik Bölüm III



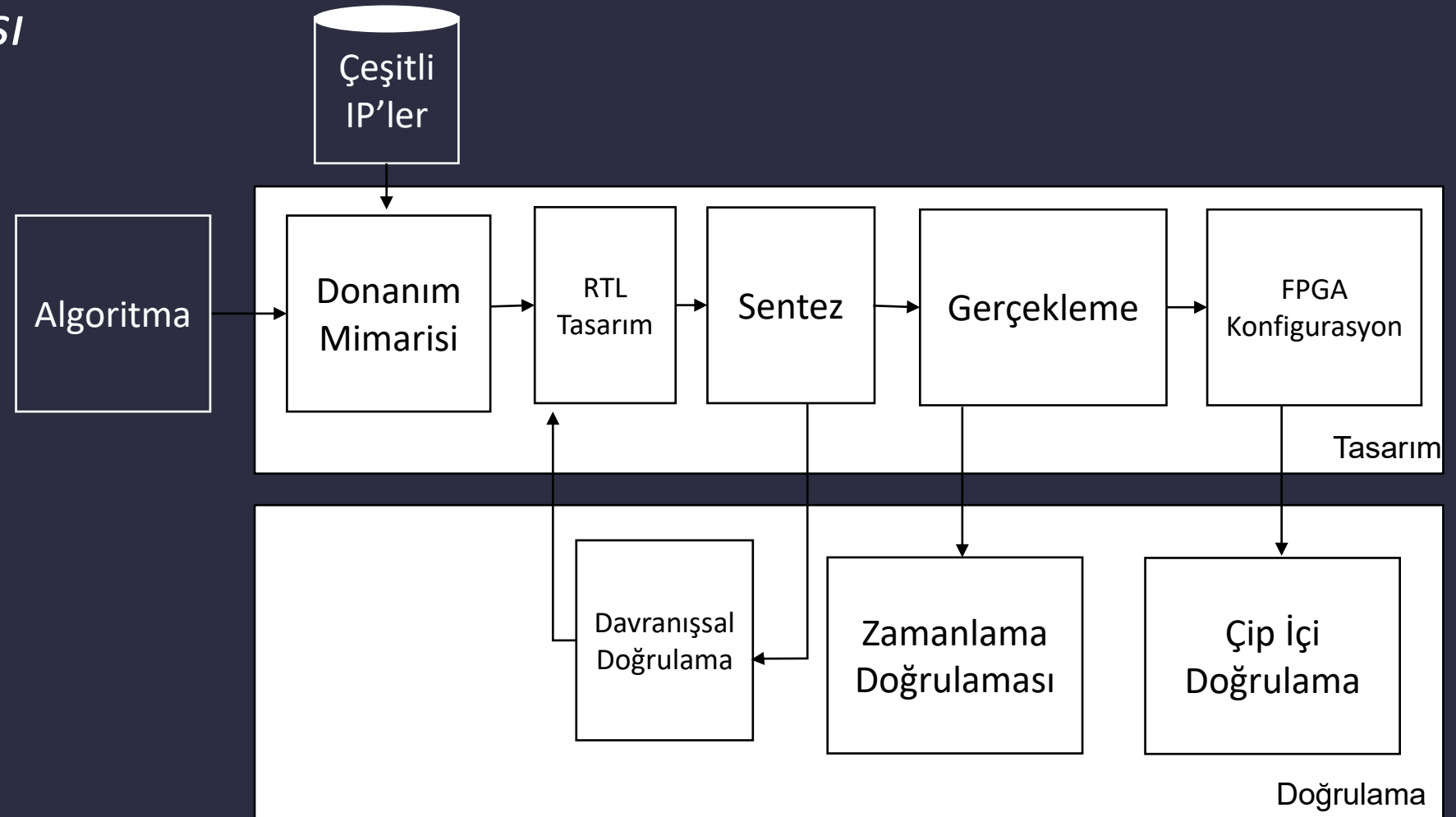
Fenerbahçe Üniversitesi

3. Hafta İçeriği

- Kombinasyonel Devreler (Combinational Circuits)
 - FPGA Nedir?
 - Vivado Tasarım Aracı

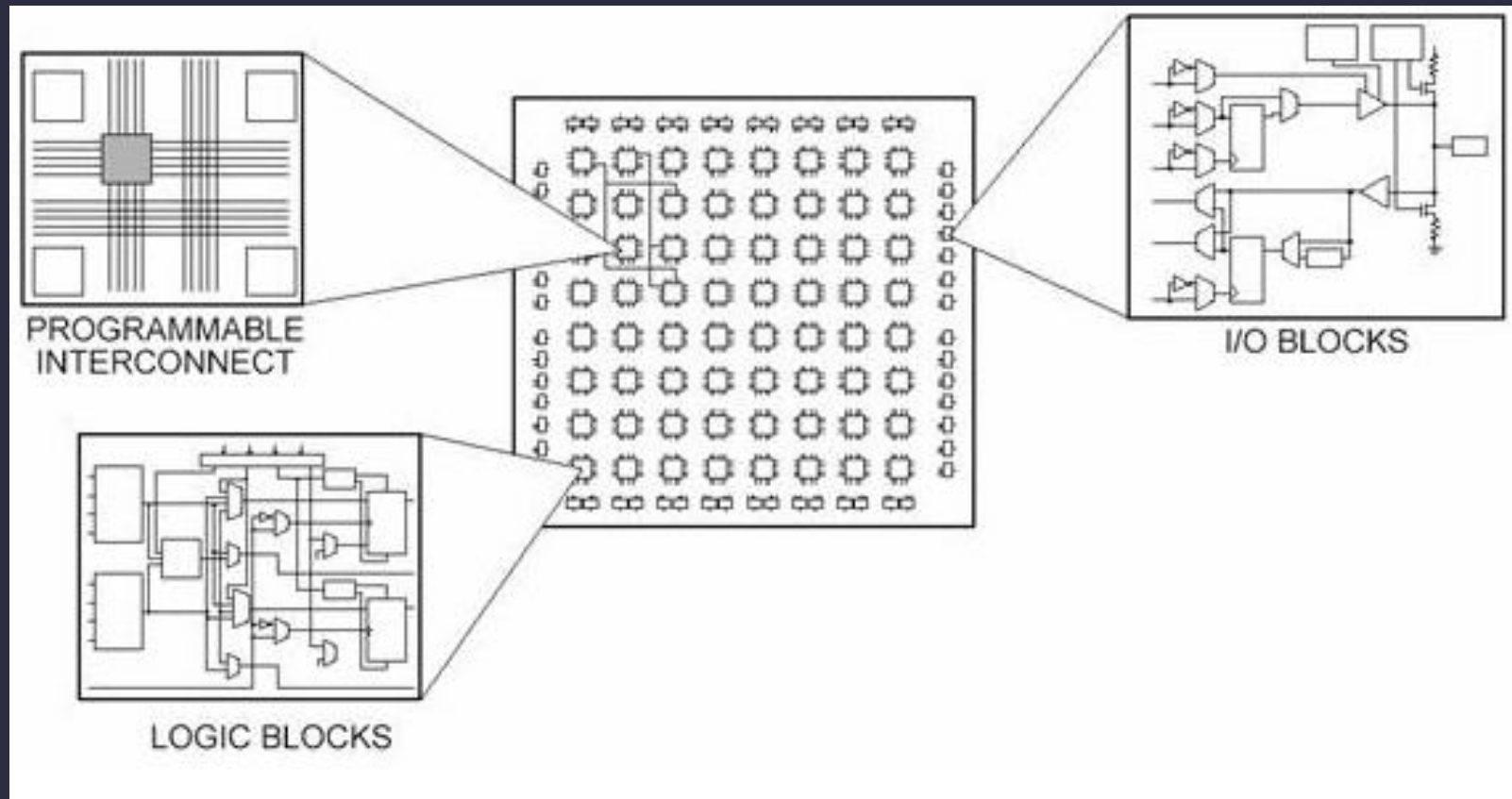
Vivado Tasarım Aracı

- Tasarım Akışı*



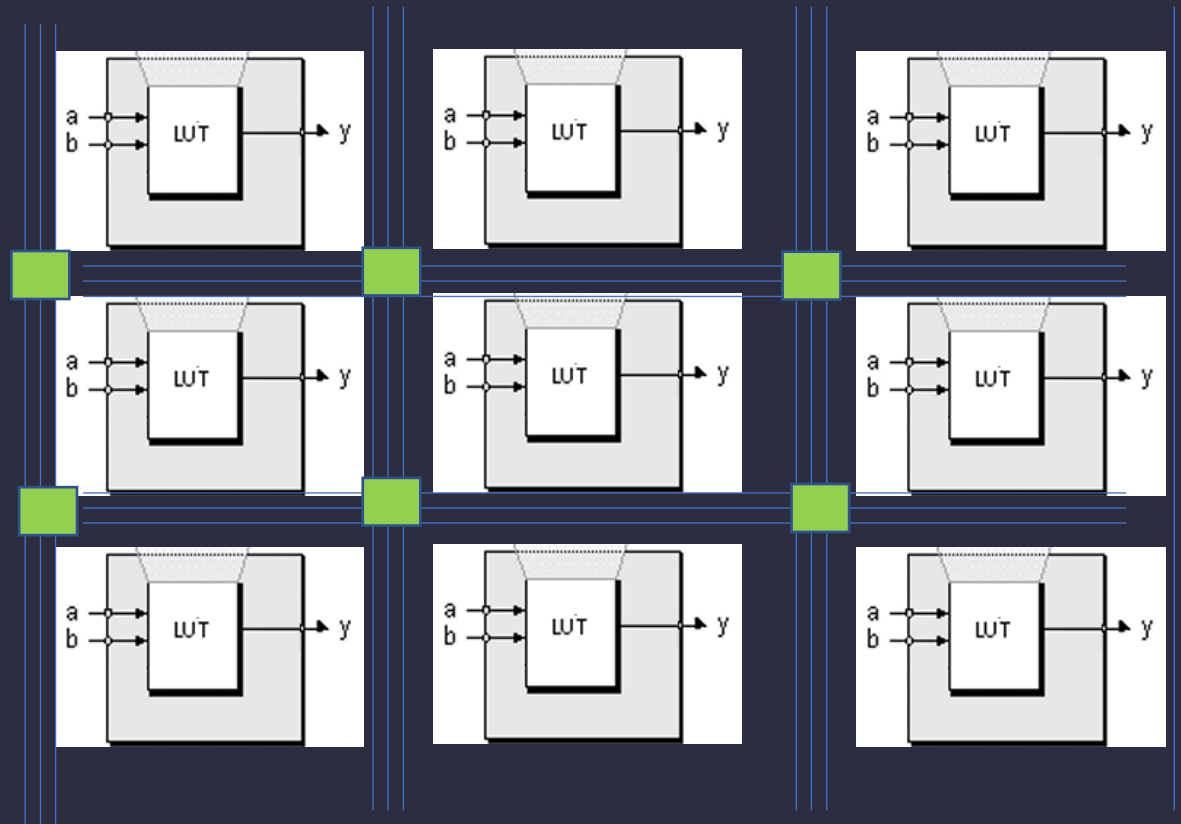
Verilog – Kombinasyonel Devreler

FPGA Nedir?



Verilog – Kombinyasyonel Devreler

FPGA Nedir?



Çip Tasarımı Eğitimi

FPGA Nedir?

- FPGA (Field Programmable Gate Arrays) entegre bir çiptir.
- İçerisinde programlanabilir bloklar ve bu bloklar arasında konfigüre edilebilir bağlantılar bulunmaktadır.
- Bu bloklar ve bağlantılar programlanarak, istenilen devre FPGA'in içerisinde gerçekleştirilebilmektedir.



Örnek FPGA Çip'i

Çip Tasarımı Eğitimi

FPGA Nedir?

- Bazı FPGA'ler sadece tek seferlik programlanabilmektedir.
- Bu FPGA'lere one-time programmable (OTP) denmektedir.

Çip Tasarımı Eğitimi

FPGA Nedir?

- FPGA'in açılımındaki "Field Programmable" kısmı, alanda programlanabilir, yani hangi iş için kullanılacak ise, o alan ile alakalı mantıksal tasarımın yüklenebileceğini ifade etmektedir.
- Bunun anlamı, FPGA çipleri fabrikada üretildikten sonra, sonradan istenilen tasarımın içerisine atılabilmesidir.

Çip Tasarımı Eğitimi

Neden FPGA önemlidir?

Çok çeşitli entegre devreler (IC – Integrated Circuits) bulunmaktadır.

- Bellek
- Mikroprosesörler
- Programlanabilir mantık cihazları (Programmable logic devices)
 - SPLD (Simple)
 - CPLD (Complex)
- Uygulama spesifik entegre devreler (ASIC – Application specific integrated circuits)
- Ve FPGA'ler ...

Çip Tasarımı Eğitimi

PLD'ler

- İç mimarisi PLD üreticisi tarafından önceden belirlenmiştir.
- FPGA'ler gibi alanda programlanabilir yapılardır
- Ancak FPGA'lere kıyasla çok daha basit ve küçüktürler.
- Küçük olması sebebiyle FPGA'lerde kurulabilen büyük devreler, PLD'lerde kurulması daha zordur.

Çip Tasarımı Eğitimi

ASIC'ler

- Bu yapılar yüz milyonlarca mantık kapısını içerebilmektedirler.
- Devasa ve karmaşıklığı yüksek devrelerin gerçekleştirilmesinde faydalıdırlar.
- ASIC'ler sadece belirli bir işlemi yapmak üzere üretilmiş entegre devrelerdir.
- ASIC'ler çok yüksek performans (yüksek frekanslarda çalışma) getirsede, ASIC tasarımı karmaşıklığı ve tasarım zamanı ve maliyetleri oldukça yüksektir.
- Ve üretilen çipler modifiye edilemez. Çipte bir hata olması durumunda üretilmiş bütün çipler çöpe atılacaktır.

Çip Tasarımı Eğitimi

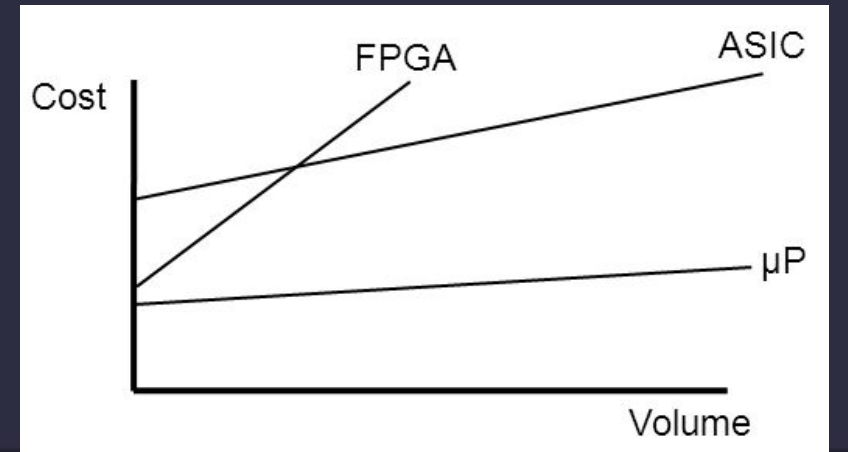
FPGA'ler

- Bu nedenle FPGA'ler PLD ve ASIC'ler arasındaki boşluğu doldurmaktadır.
- FPGA'Ler PLD'ler gibi programlanabilir yapıdadır.
- Ancak PLD'lere göre çok daha fazla programlanabilir alanı vardır.
- ASIC'lerin aksine; ortaya çıkan tasarımda bir hata olması durumunda, tasarım tekrar tekrar modifiye edilerek test edilebilir.

Çip Tasarımı Eğitimi

FPGA'ler

- FPGA'ler ASIC'lere göre çok daha ucuzdurlar. (ASIC'ler ancak milyonlarca adet üretildiğinde birim maliyeti ucuz olmaktadır)
- FPGA'ler üzerinde tasarım geliştirmek ASIC bir tasarım yaratmaya göre çok daha kolaydır.
- Tasarım süresinin az olması nedeniyle, bir ürün üretileceğinde, piyasaya çıkma süresi daha kısadır.
- Tasarım ekipleri ASIC'te yatırılması gereken NRE (Non recurring engineering) yatırımını yapmadan, FPGA üzerinde tasarımlarını geliştirmeleri mümkün olmaktadır.



Çip Tasarımı Eğitimi

FPGA'in ağırlıklı olarak kullanıldığı sektörler ve uygulama alanları

- Telekomünikasyon
- Ağ haberleşmesi
- Otomotiv
- Sağlık
- Çeşitli endüstriyel uygulamalar
- ASIC tasarımlarının prototipleri

- DSP (Digital signal processor) uygulamaları
- SoC (System on Chip), tüm gerekli elektronik sistemin bir araya toplandığı tek bir IC

Çip Tasarımı Eğitimi

FPGA'lerin girdiği pazarlar

- ASIC: Bir çok ASIC uygulaması artık FPGA'ler ile yapılmaktadır.
- DSP: Günümüz FPGA'leri ile sinyal işleme uygulamaları, DSP cihazlarına göre çok daha yüksek hızlarda yapılabilmektedir.
- Mikrokontrolörler: Basit kontrol fonksiyonları için kullanılan mekanizmalardır. FPGA fiyatlarının düşmesi ile bu basit görevleri görmesi için FPGA'in içerisine soft işlemciler gömülerek aynı mekanizmayı görmesi sağlanmaktadır.

Çip Tasarımı Eğitimi

FPGA Kullanım Nedenleri

- Hesaplama Gücü
- Nanosaniye mertebesinde giriş alma ve çıktı verebilme

Çip Tasarımı Eğitimi

FPGA Kullanım Nedenleri

İşlemciler

`c[0]=a[0]+b[0];`

`c[1]=a[1]+b[1];`

`c[2]=a[2]+b[2];`

`c[3]=a[3]+b[3];`

...

`c[1000]=a[1000]+b[1000];`

FPGA

`c[0]<=a[0]+b[0];`

`c[1]<=a[1]+b[1];`

`c[2]<=a[2]+b[2];`

`c[3]<=a[3]+b[3];`

...

`c[1000]<=a[1000]+b[1000];`

Çip Tasarımı Eğitimi

FPGA Kullanım Nedenleri

İşlemciler

```
c[0]=a[0]+b[0];  
c[1]=a[1]+b[1];  
c[2]=a[2]+b[2];  
c[3]=a[3]+b[3];  
...  
c[1000]=a[1000]+b[1000];
```



FPGA

```
c[0]<=a[0]+b[0];  
c[1]<=a[1]+b[1];  
c[2]<=a[2]+b[2];  
c[3]<=a[3]+b[3];  
...  
c[1000]<=a[1000]+b[1000];
```

Çip Tasarımı Eğitimi

FPGA Kullanım Nedenleri

İşlemciler

```
c[0]=a[0]+b[0];  
c[1]=a[1]+b[1]; ←  
c[2]=a[2]+b[2];  
c[3]=a[3]+b[3];  
...  
c[1000]=a[1000]+b[1000];
```

FPGA

```
c[0]<=a[0]+b[0];  
c[1]<=a[1]+b[1];  
c[2]<=a[2]+b[2];  
c[3]<=a[3]+b[3];  
...  
c[1000]<=a[1000]+b[1000];
```

Çip Tasarımı Eğitimi

FPGA Kullanım Nedenleri

İşlemciler

```
c[0]=a[0]+b[0];  
c[1]=a[1]+b[1];  
c[2]=a[2]+b[2]; ←  
c[3]=a[3]+b[3];  
...  
c[1000]=a[1000]+b[1000];
```

FPGA

```
c[0]<=a[0]+b[0];  
c[1]<=a[1]+b[1];  
c[2]<=a[2]+b[2];  
c[3]<=a[3]+b[3];  
...  
c[1000]<=a[1000]+b[1000];
```

Çip Tasarımı Eğitimi

FPGA Kullanım Nedenleri

İşlemciler

`c[0]=a[0]+b[0];`

`c[1]=a[1]+b[1];`

`c[2]=a[2]+b[2];`

`c[3]=a[3]+b[3];` 

...

`c[1000]=a[1000]+b[1000];`

FPGA

`c[0]<=a[0]+b[0];`

`c[1]<=a[1]+b[1];`

`c[2]<=a[2]+b[2];`

`c[3]<=a[3]+b[3];`

...

`c[1000]<=a[1000]+b[1000];`

Çip Tasarımı Eğitimi

FPGA Kullanım Nedenleri

İşlemciler

`c[0]=a[0]+b[0];`

`c[1]=a[1]+b[1];`

`c[2]=a[2]+b[2];`

`c[3]=a[3]+b[3];`

...

`c[1000]=a[1000]+b[1000];` 

FPGA

`c[0]<=a[0]+b[0];`

`c[1]<=a[1]+b[1];`

`c[2]<=a[2]+b[2];`

`c[3]<=a[3]+b[3];`

...

`c[1000]<=a[1000]+b[1000];`

Çip Tasarımı Eğitimi

FPGA Kullanım Nedenleri

İşlemciler

`c[0]=a[0]+b[0];`

`c[1]=a[1]+b[1];`

`c[2]=a[2]+b[2];`

`c[3]=a[3]+b[3];`

...

`c[1000]=a[1000]+b[1000];`

FPGA

`c[0]<=a[0]+b[0];`



`c[1]<=a[1]+b[1];`



`c[2]<=a[2]+b[2];`



`c[3]<=a[3]+b[3];`



...

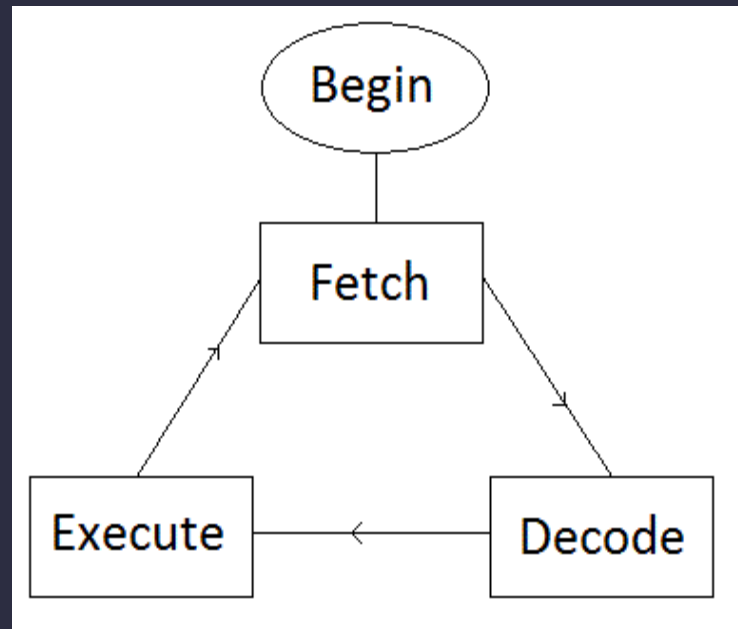
`c[1000]<=a[1000]+b[1000];`



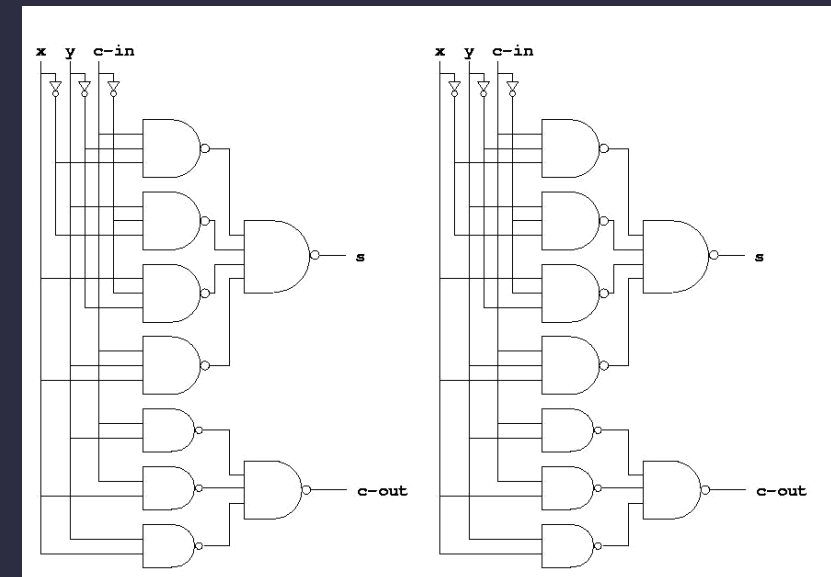
Çip Tasarımı Eğitimi

FPGA Kullanım Nedenleri

İşlemciler



FPGA



Çip Tasarımı Eğitimi

FPGA Kullanım Nedenleri

CPU

1000 adet işlem için (her işlemin 1 clock cycle sürdüğü varsayılmıştır),

Ortalama Mikro işlemci hızı: 3 ghz için,
İşletim sistemi yok düşünüldüğünde

Gerekli zaman = İşlem sayısı X (1/Frekans)

$$\begin{aligned} &= 1000 \times 1/3 \text{ milyar} \\ &= 333 \text{ nano saniye} \end{aligned}$$

FPGA

1000 adet işlem için (her işlemin 1 clock cycle sürdüğü varsayılmıştır),

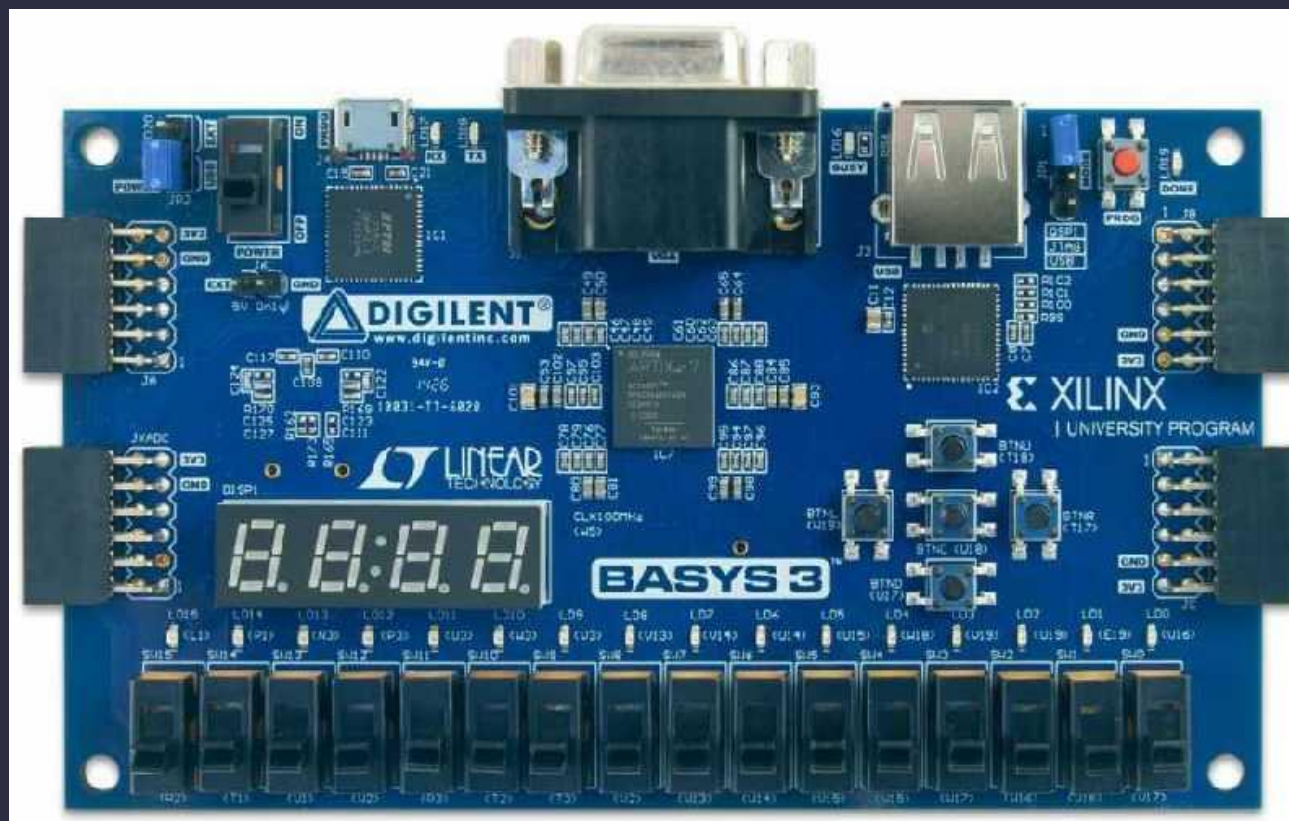
Ortalama fpga frekansı: 100 mhz için

Gerekli zaman = 1 x (1/Frekans)

$$\begin{aligned} &= 1 \times 1/100 \text{ m} \\ &= 10 \text{ nano saniye} \end{aligned}$$

Verilog – Kombinasyonel Devreler

LAB'larda Xilinx Artix 7 FPGA Barındıran Basys3 FPGA'leri kullanılacaktır.



Verilog – Kombinasyonel Devreler

Xilinx'in FPGA'lerini programlamak için Vivado isimli bir geliştirme ortamı bulunmaktadır.

Vivado yazılımı ile FPGA tasarımları yapılmaktadır.

Verilog – Kombinasyonel Devreler

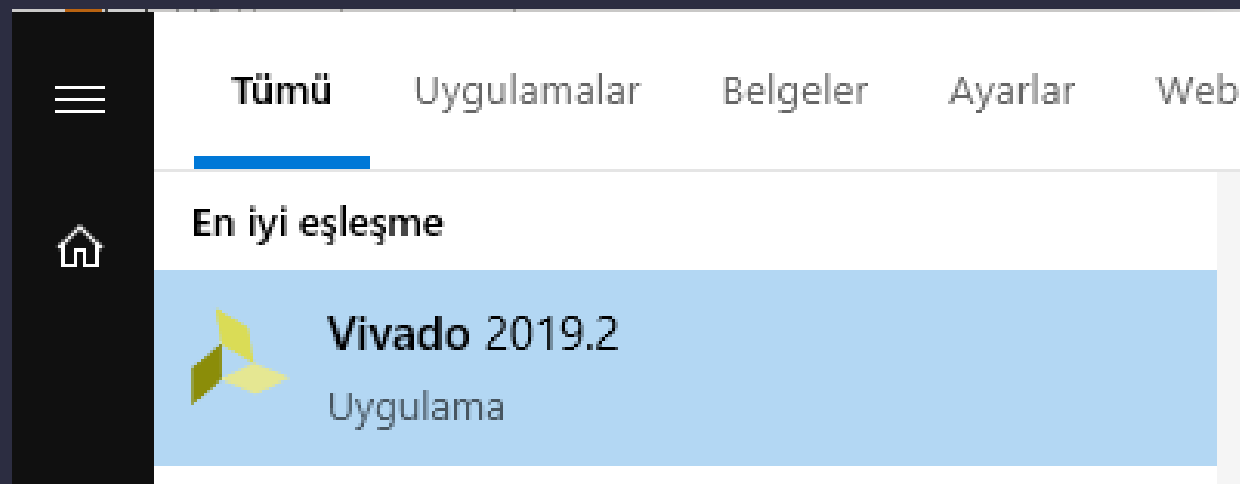
Vivado Tasarım Aracı

- İndirme Adresi: <https://www.xilinx.com/support/download.html>
- Kurulum ve Lisanslama Video'su: <https://www.youtube.com/watch?v=yW1bJbXnbRU>

Verilog – Kombinasyonel Devreler

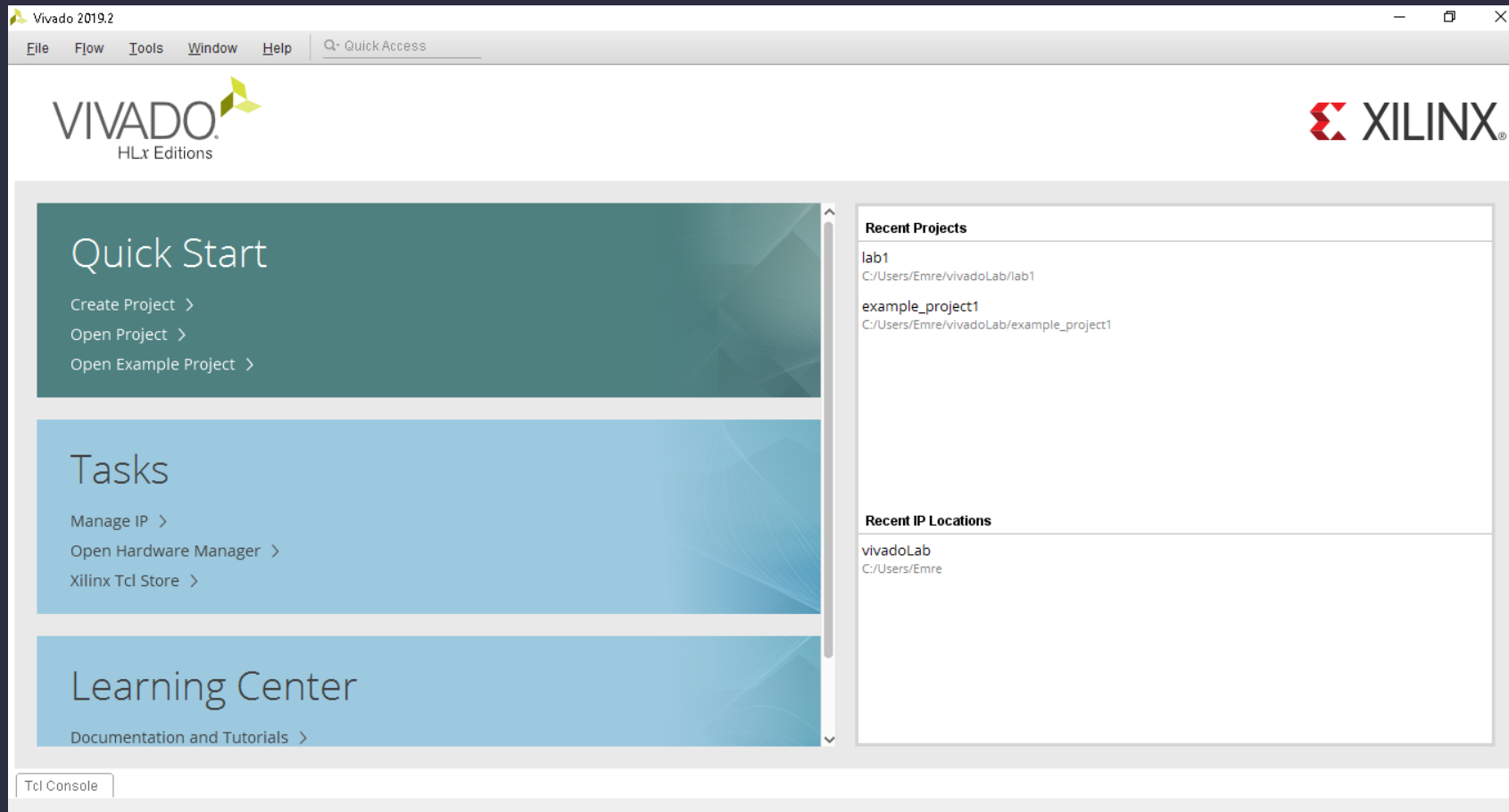
Vivado Tasarım Aracı

Vivado Tasarım Aracı, Xilinx'in 7 ve daha yeni jenerasyon FPGA'leri için kullanılabilen bir geliştirme ortamıdır. Bu ortam Xilinx'in sunduğu çeşitli geliştirme ve doğrulama araçlarını barındırır. Başlat'ın içerisinde 'Vivado' kelimesi ile arama yaparak uygulamanın kısa yoluna erişilebilir.



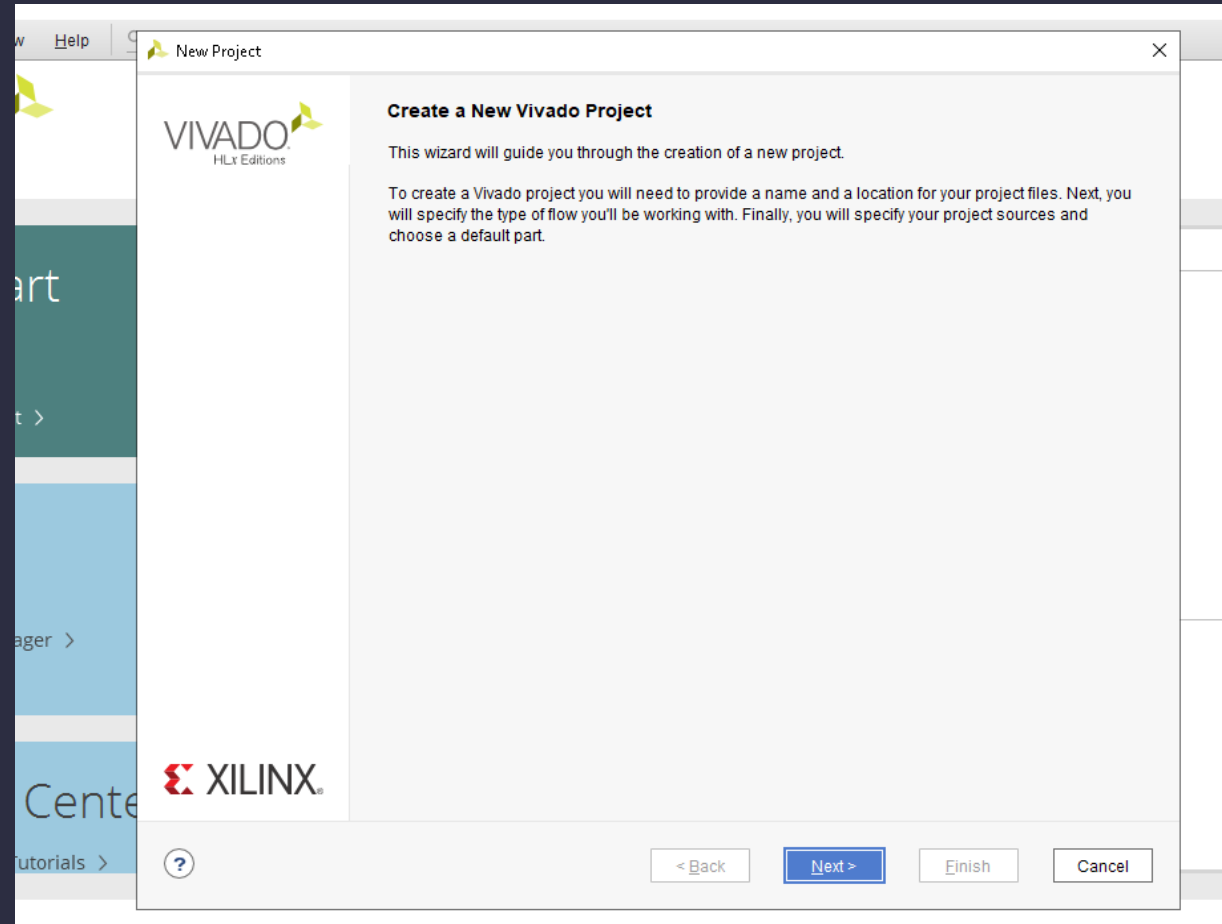
Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı



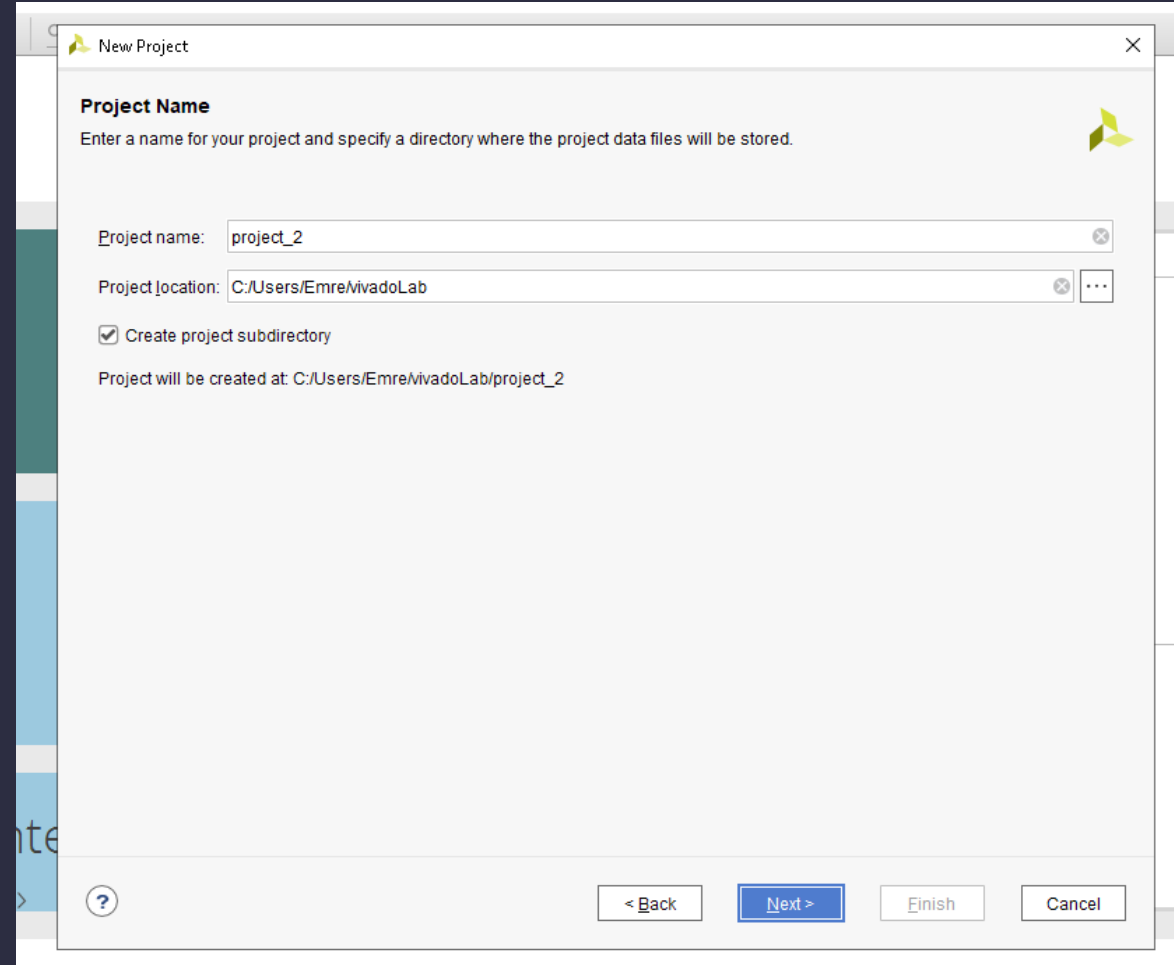
Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı



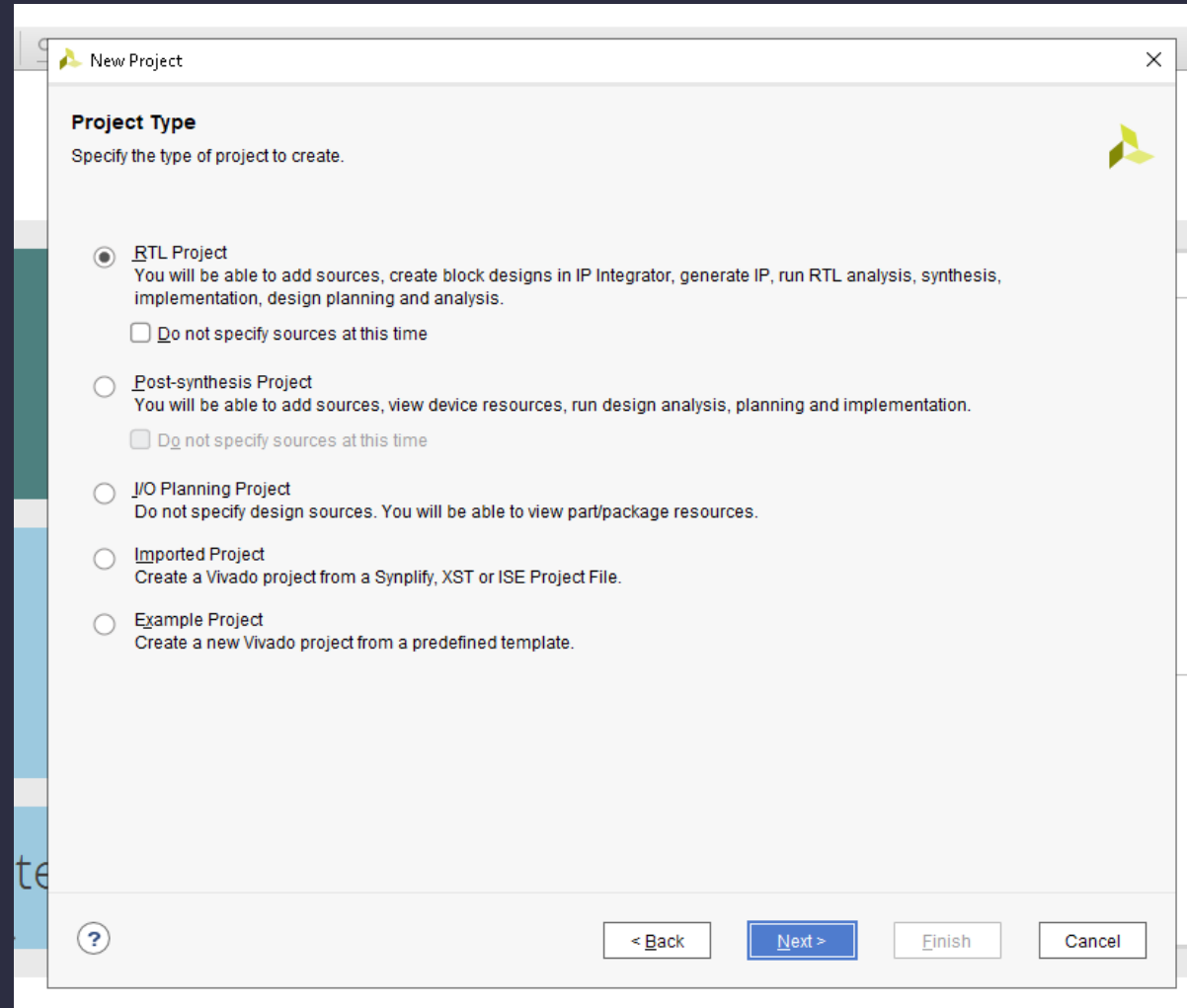
Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı



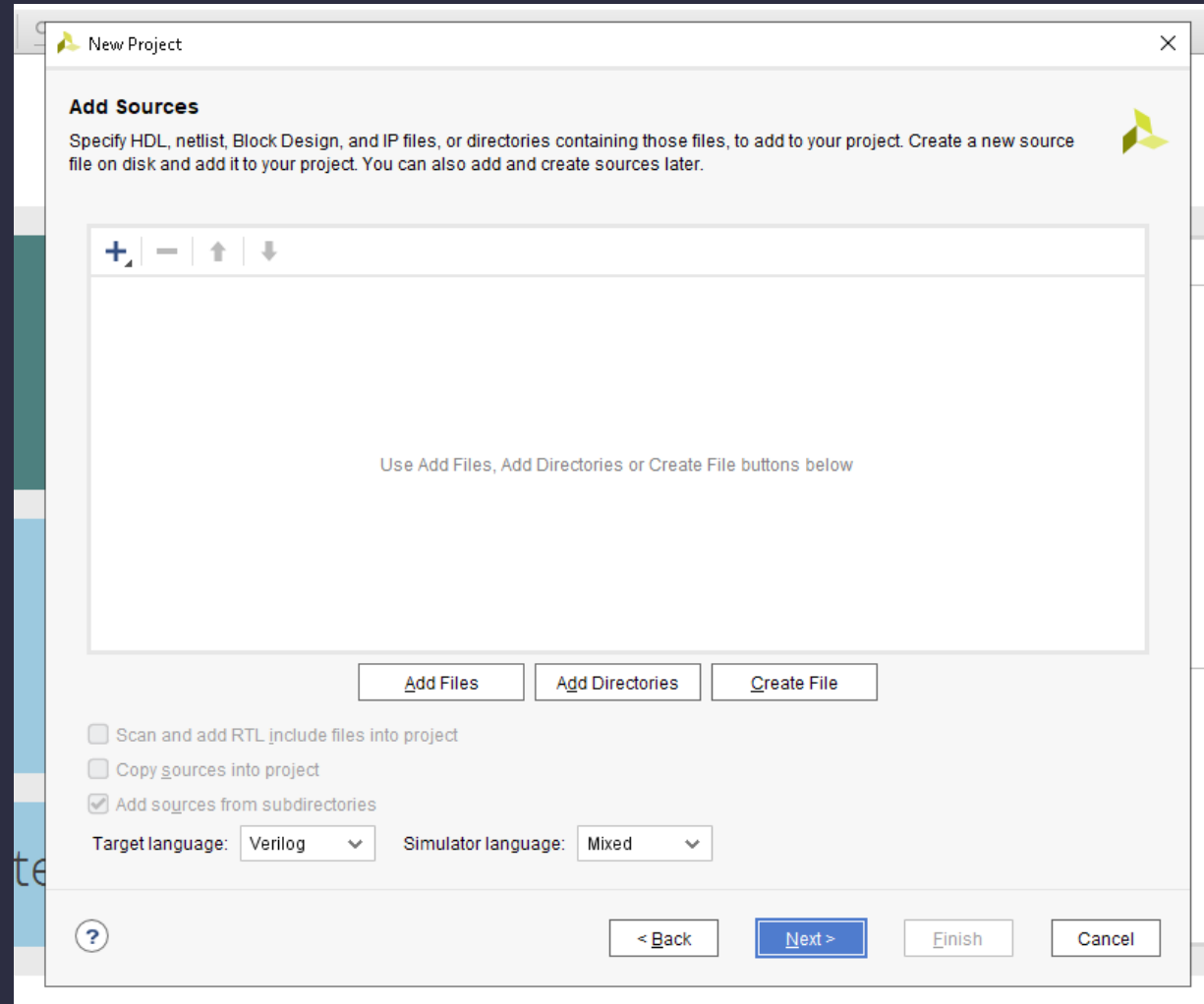
Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı



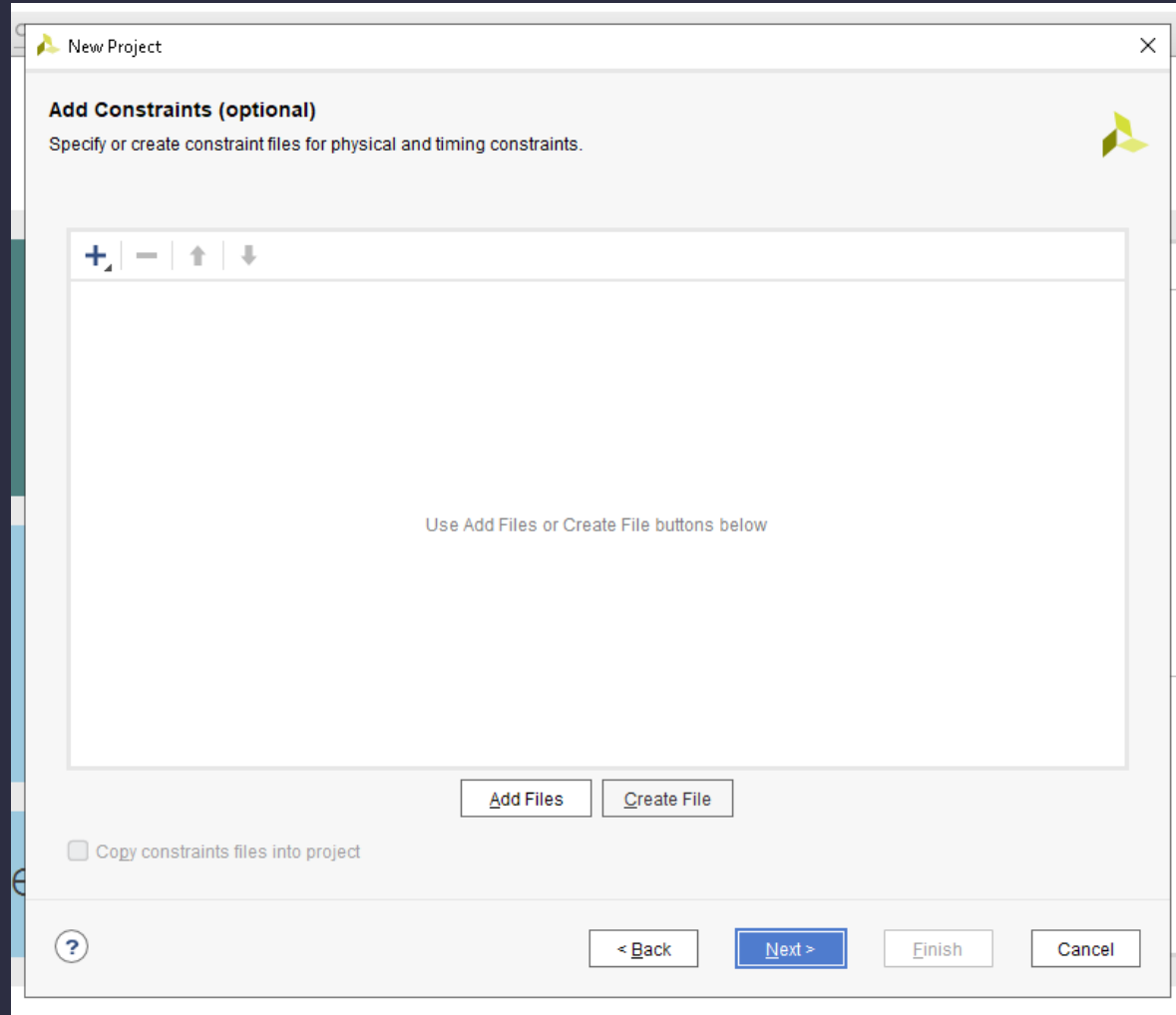
Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı



Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı



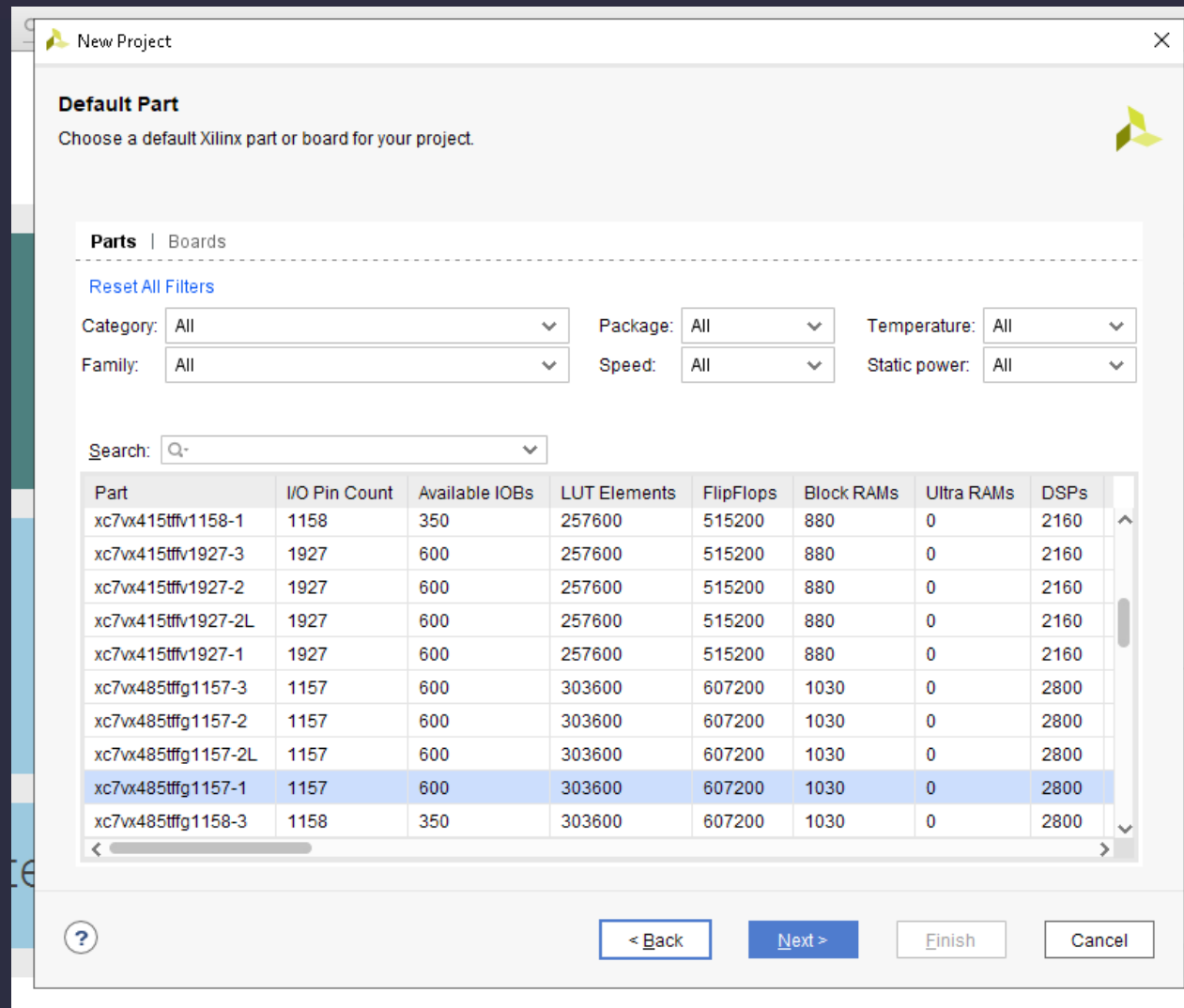
Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı

LAB'larda kullanacağımız FPGA olan

XC7A35Tcpg236-1

Model seçilmelidir.



The image shows the 'New Project' dialog box in Vivado. The 'Default Part' section is active, showing a table of available parts. The table has columns for Part, I/O Pin Count, Available IOBs, LUT Elements, FlipFlops, Block RAMs, Ultra RAMs, and DSPs. The part 'xc7vx485tffg1157-1' is selected, highlighted in blue. Below the table are navigation buttons: '< Back', 'Next >', 'Finish', and 'Cancel'.

Default Part
Choose a default Xilinx part or board for your project.

Parts | Boards

[Reset All Filters](#)

Category: All Package: All Temperature: All
Family: All Speed: All Static power: All

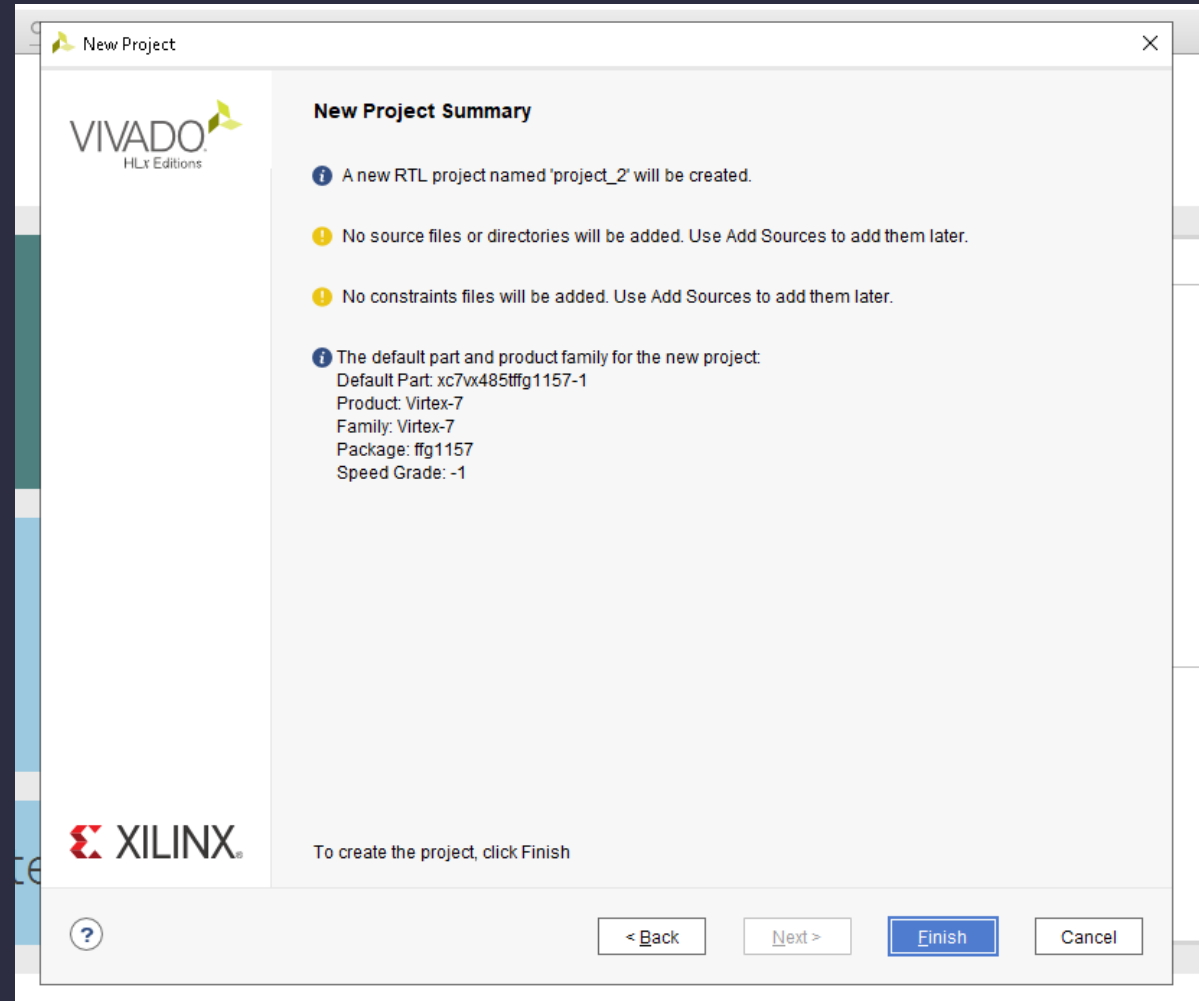
Search: Q-

Part	I/O Pin Count	Available IOBs	LUT Elements	FlipFlops	Block RAMs	Ultra RAMs	DSPs
xc7vx415tffv1158-1	1158	350	257600	515200	880	0	2160
xc7vx415tffv1927-3	1927	600	257600	515200	880	0	2160
xc7vx415tffv1927-2	1927	600	257600	515200	880	0	2160
xc7vx415tffv1927-2L	1927	600	257600	515200	880	0	2160
xc7vx415tffv1927-1	1927	600	257600	515200	880	0	2160
xc7vx485tffg1157-3	1157	600	303600	607200	1030	0	2800
xc7vx485tffg1157-2	1157	600	303600	607200	1030	0	2800
xc7vx485tffg1157-2L	1157	600	303600	607200	1030	0	2800
xc7vx485tffg1157-1	1157	600	303600	607200	1030	0	2800
xc7vx485tffg1158-3	1158	350	303600	607200	1030	0	2800

< Back Next > Finish Cancel

Verilog – Kombinasyonel Devreler

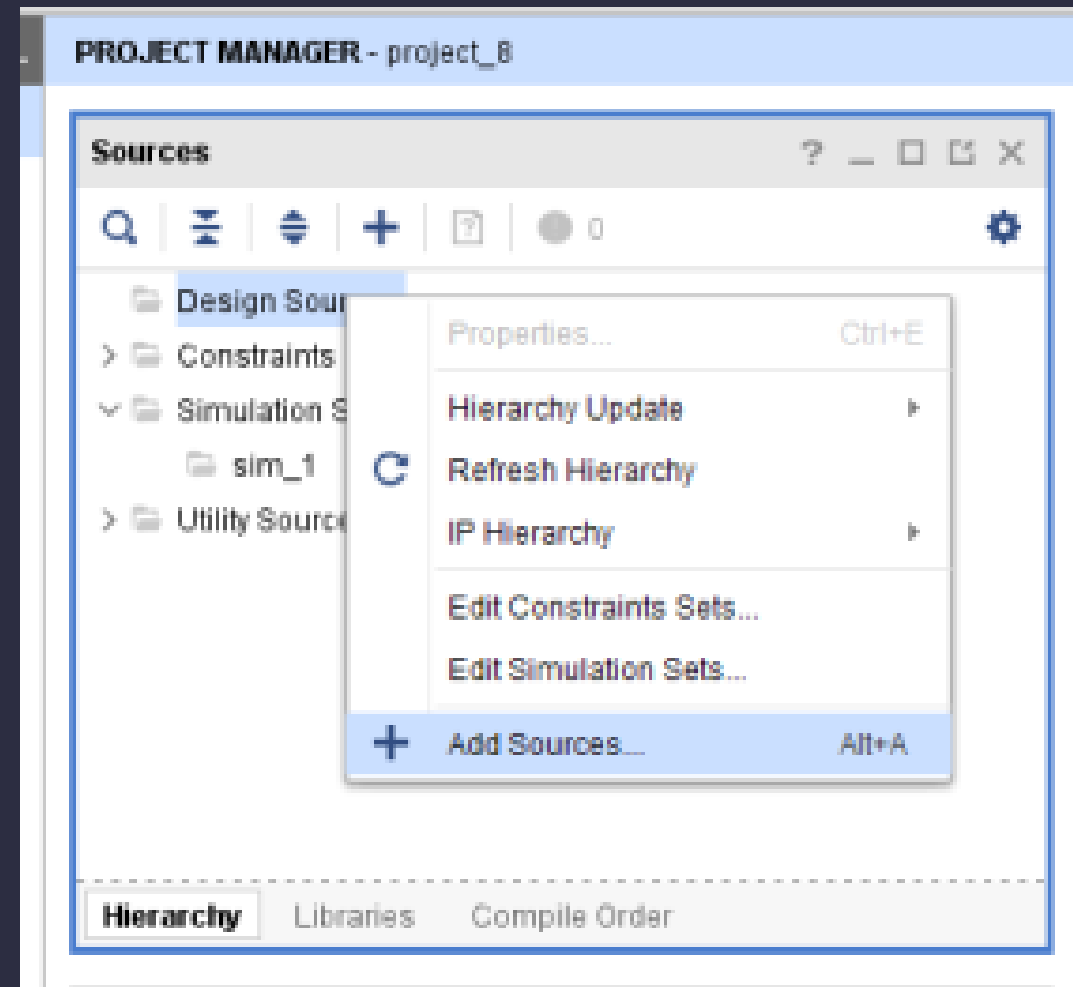
Vivado Tasarım Aracı



Verilog – Kombinasyonel Devreler

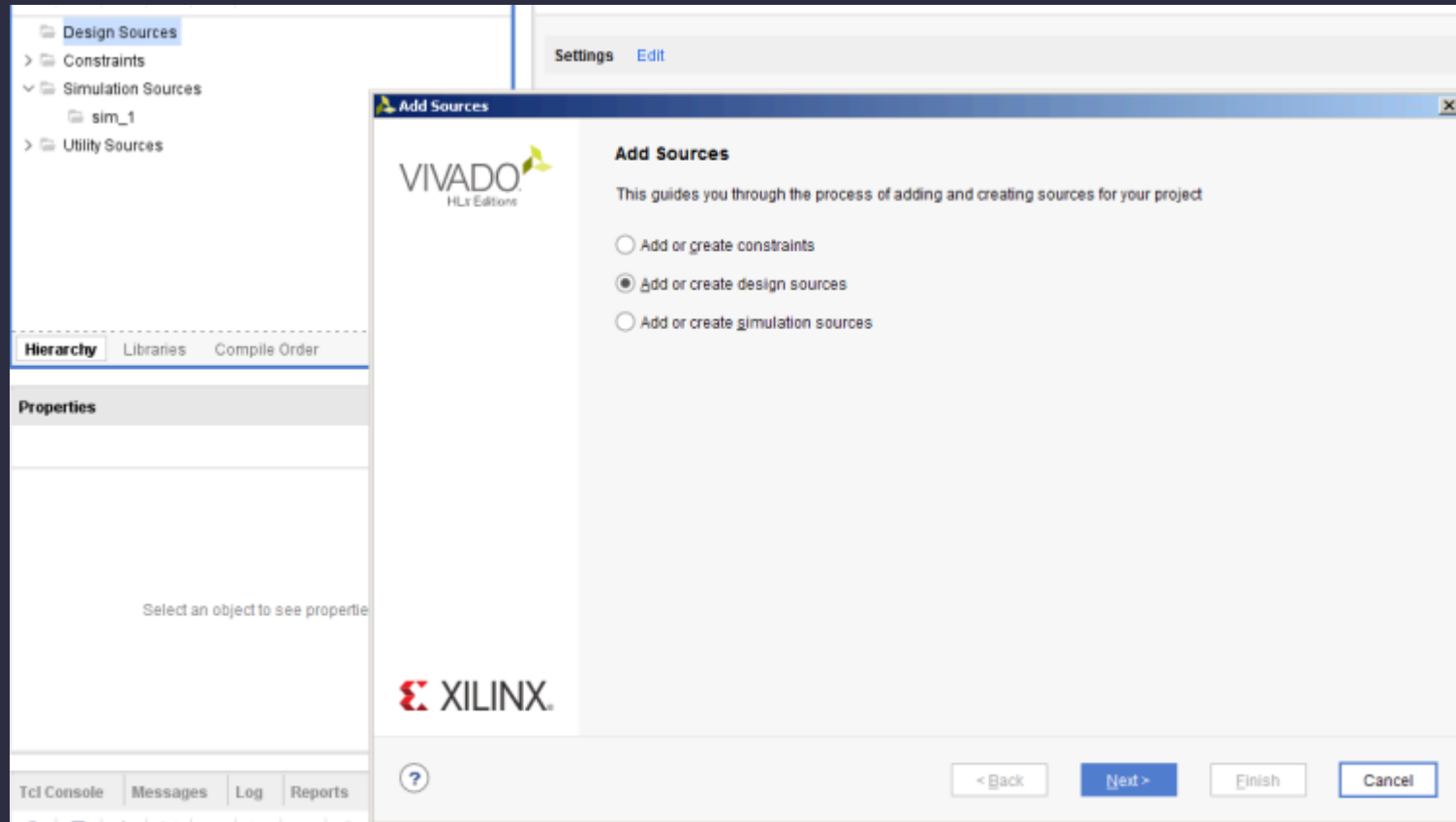
Vivado Tasarım Aracı

Yeni tasarım kaynağı eklemek



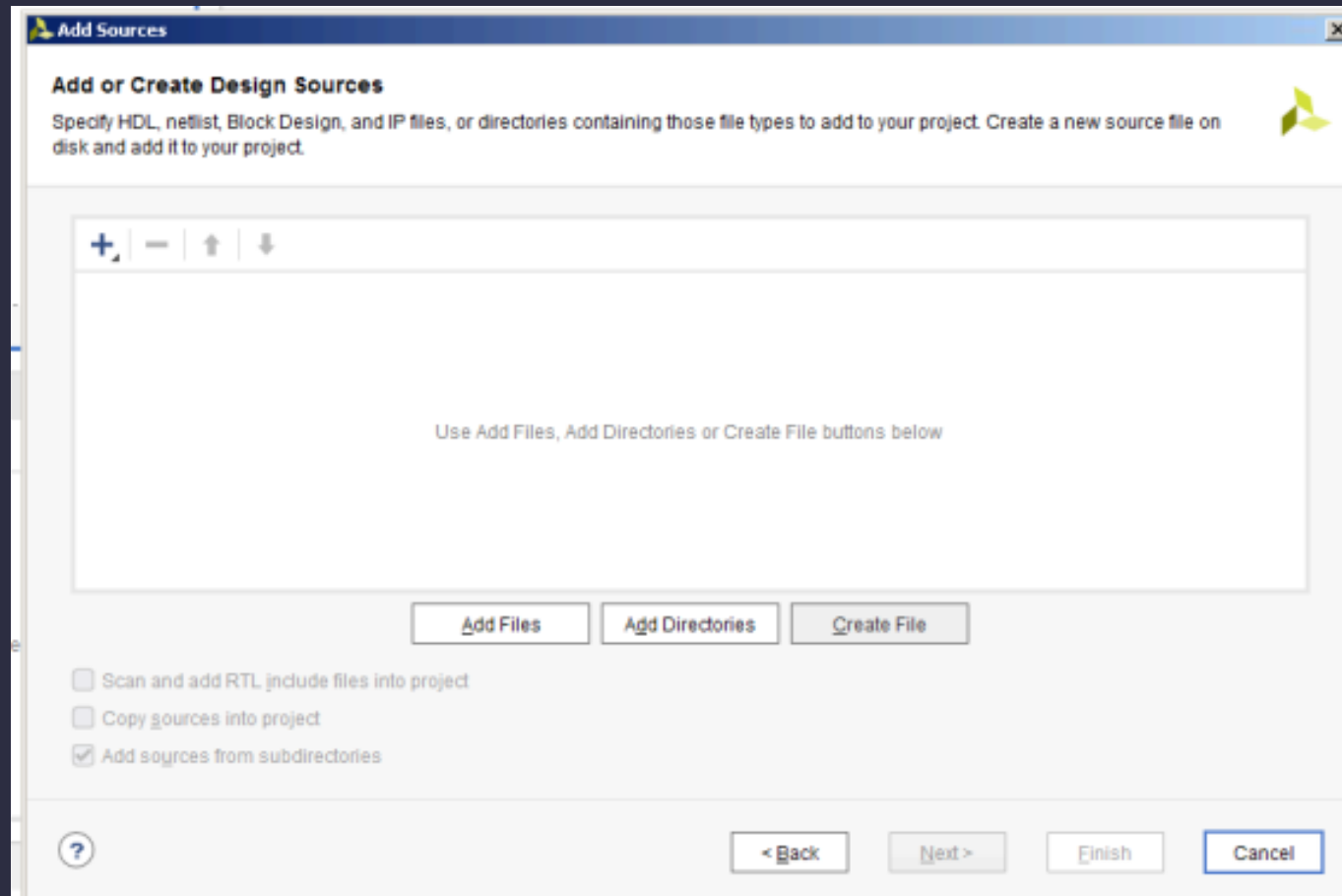
Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı



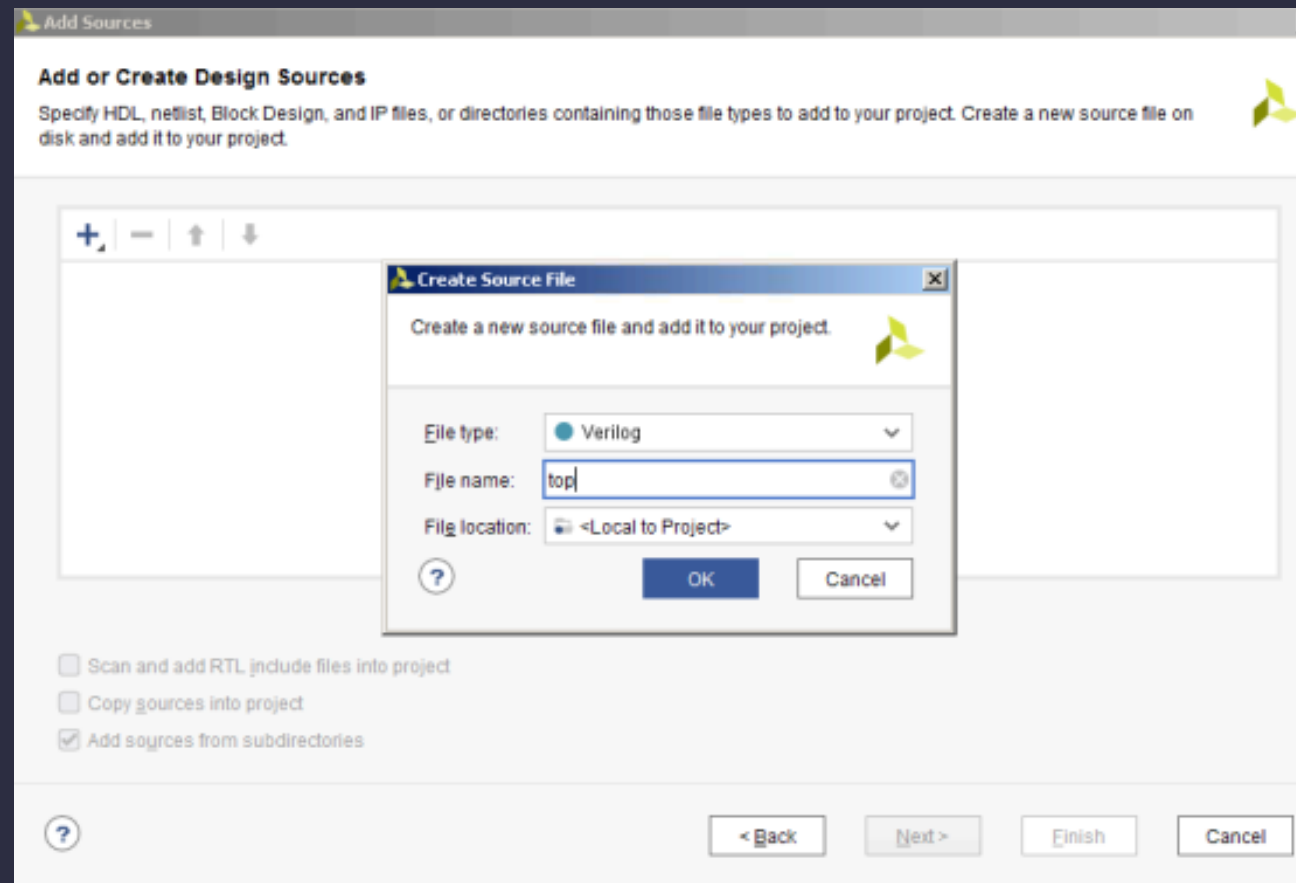
Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı



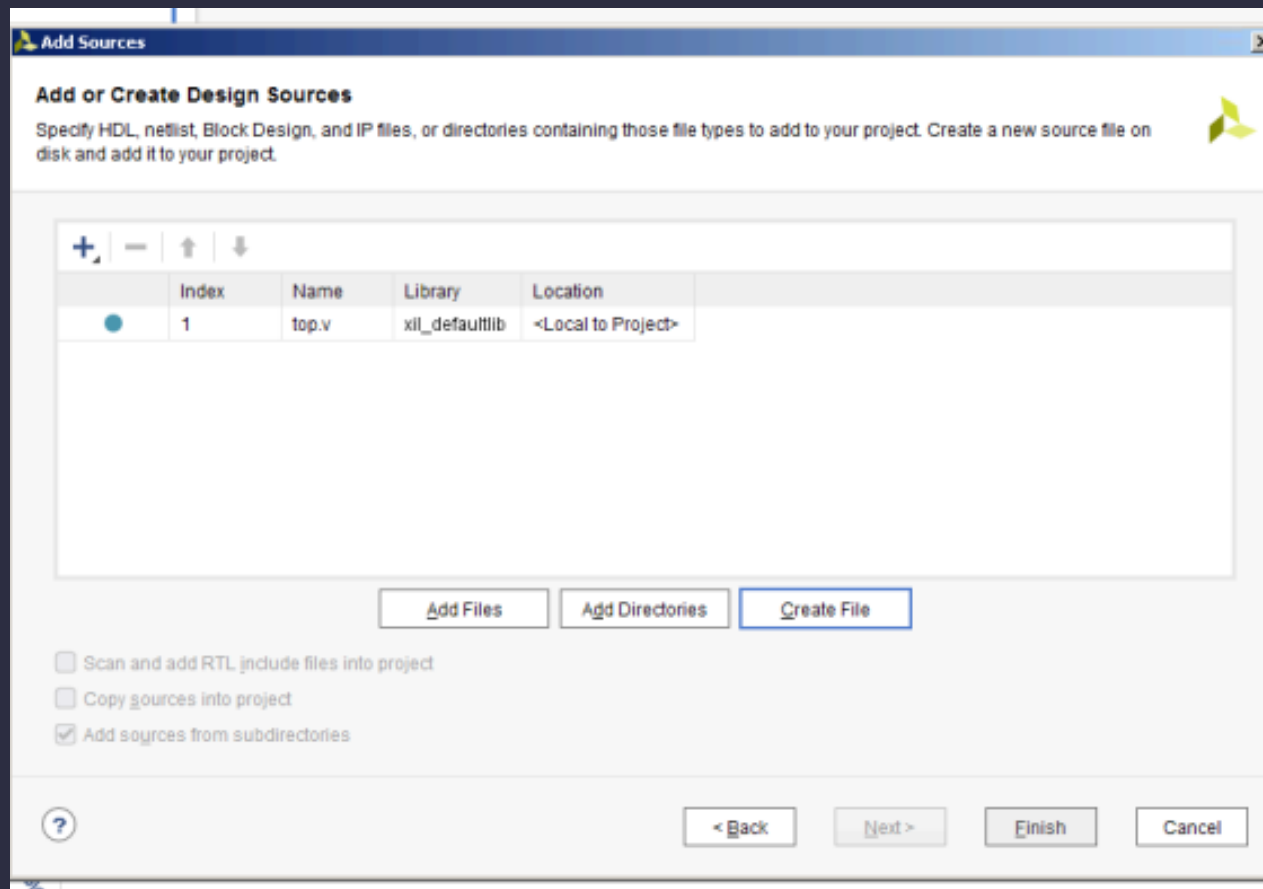
Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı



Verilog – Kombinasyonel Devreler

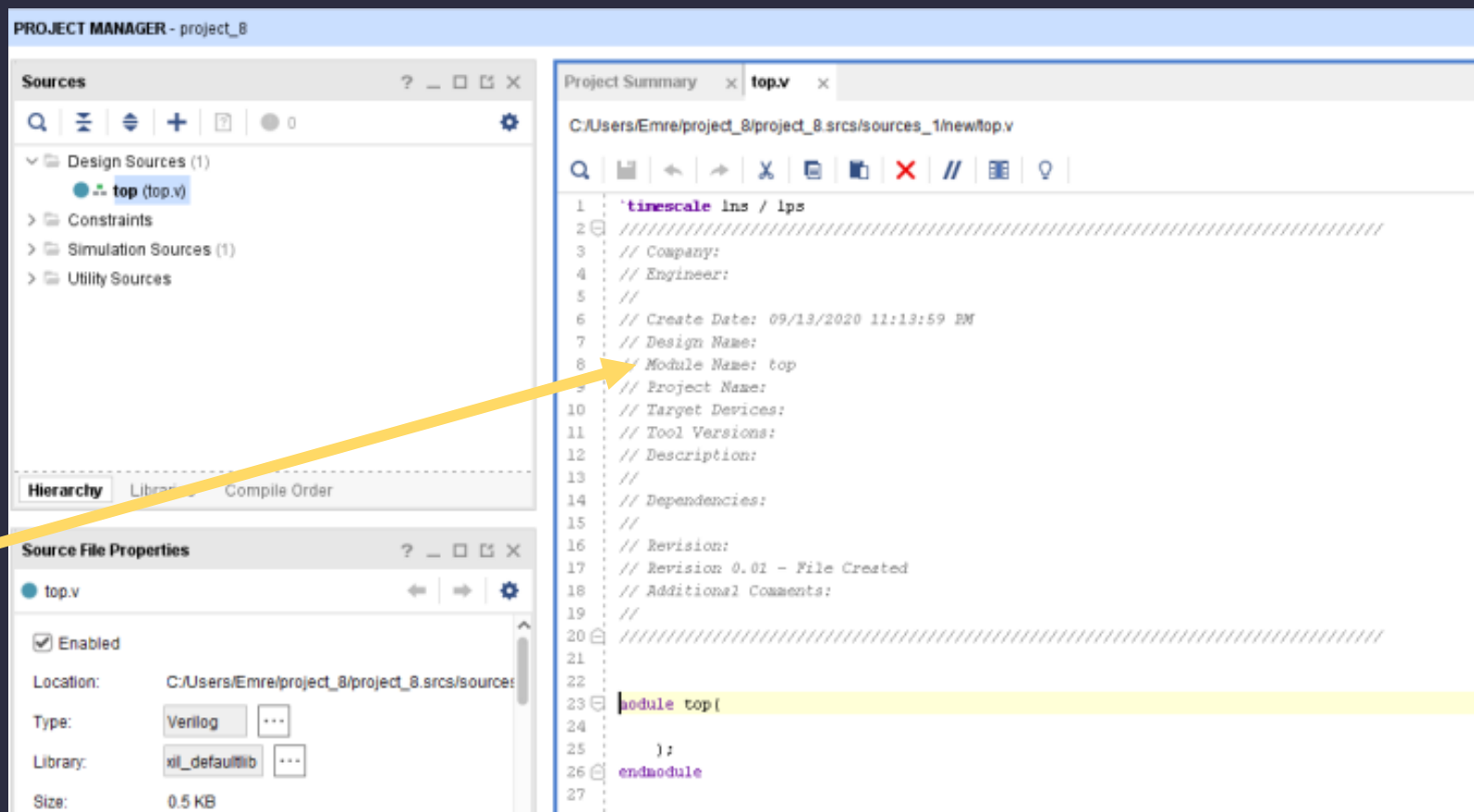
Vivado Tasarım Aracı



Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı

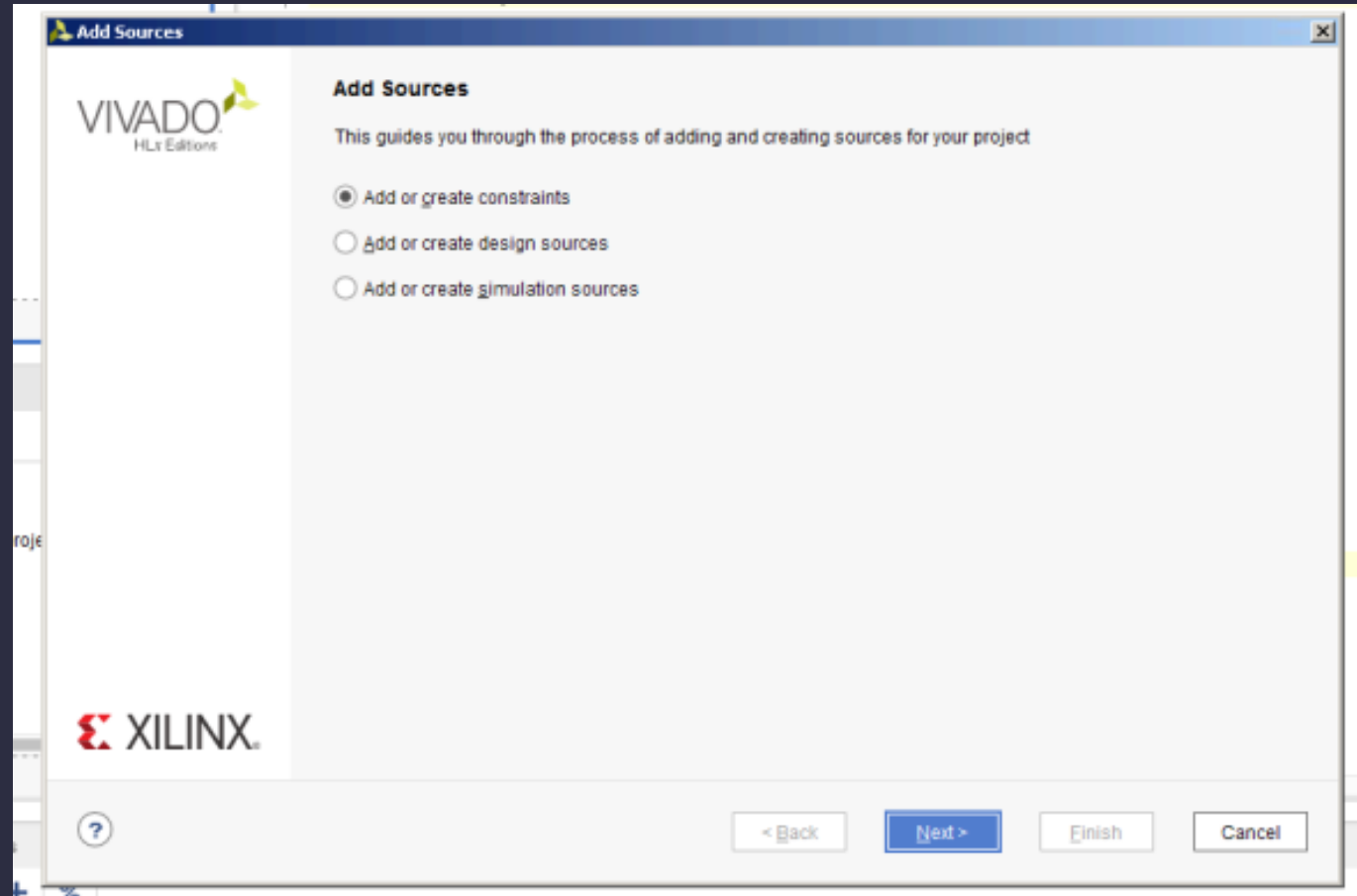
RTL Tasarımı
Yapılacak Dosya



Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı

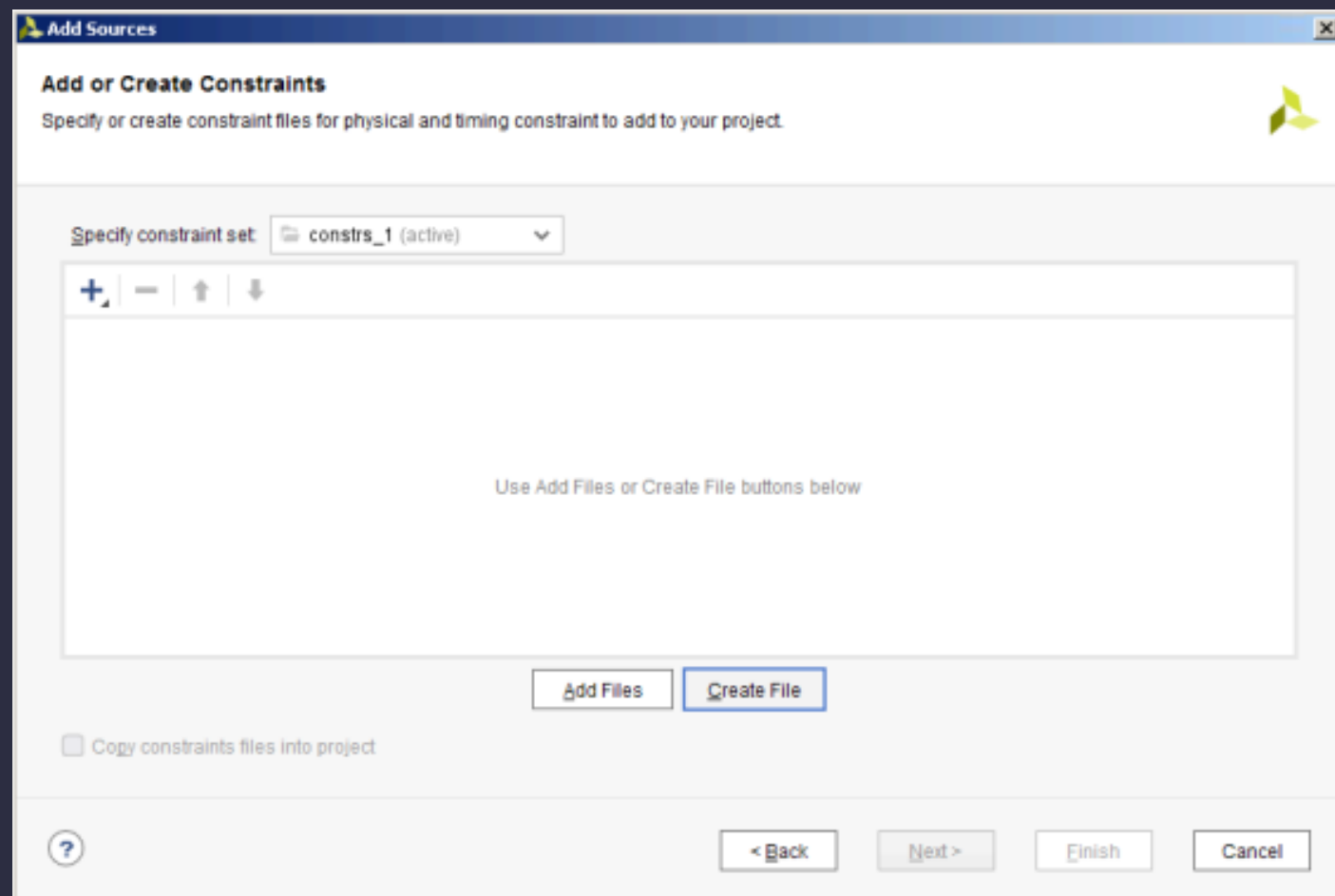
Kısıt (constraint) eklemek



Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı

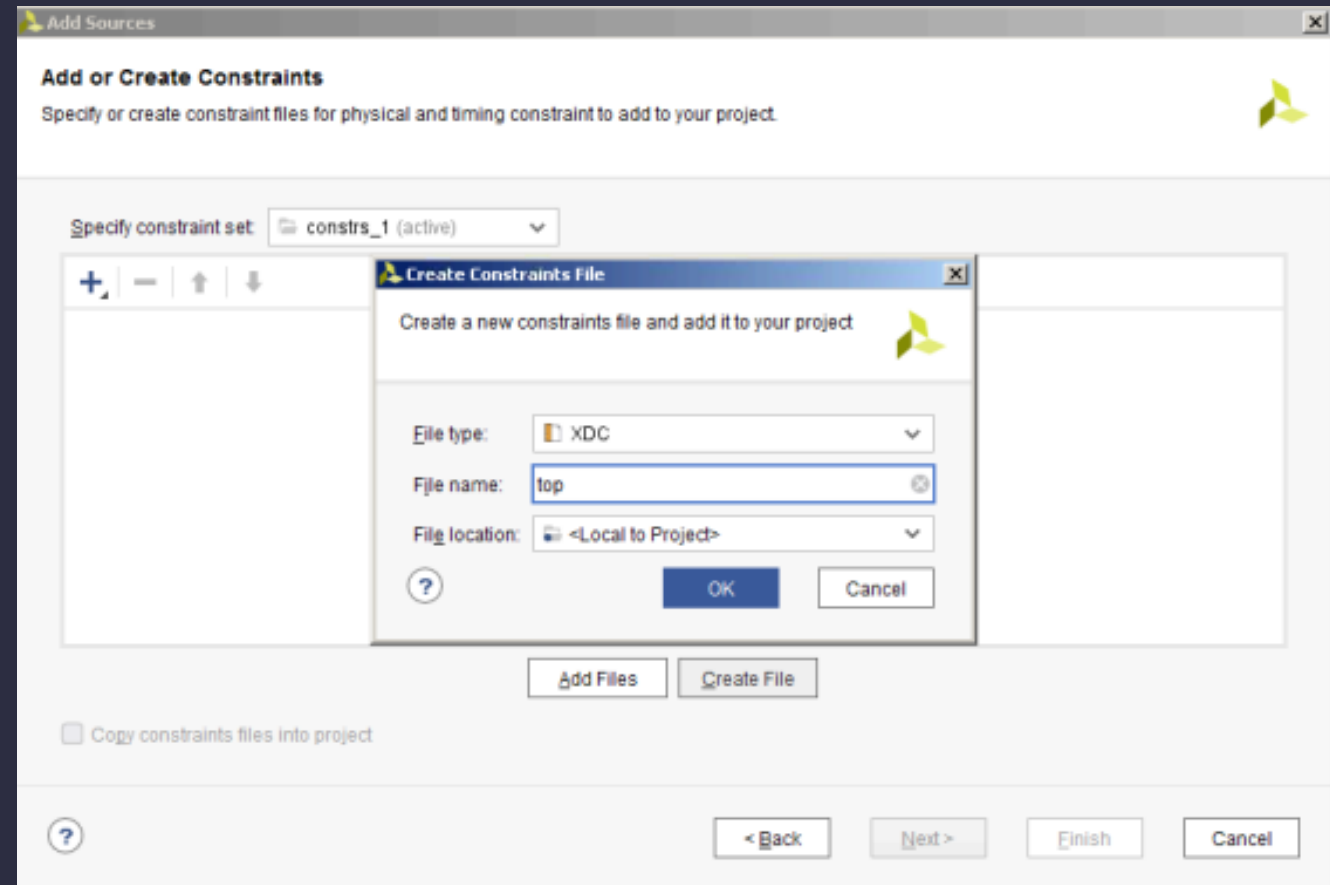
Kısıt (constraint) eklemek



Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı

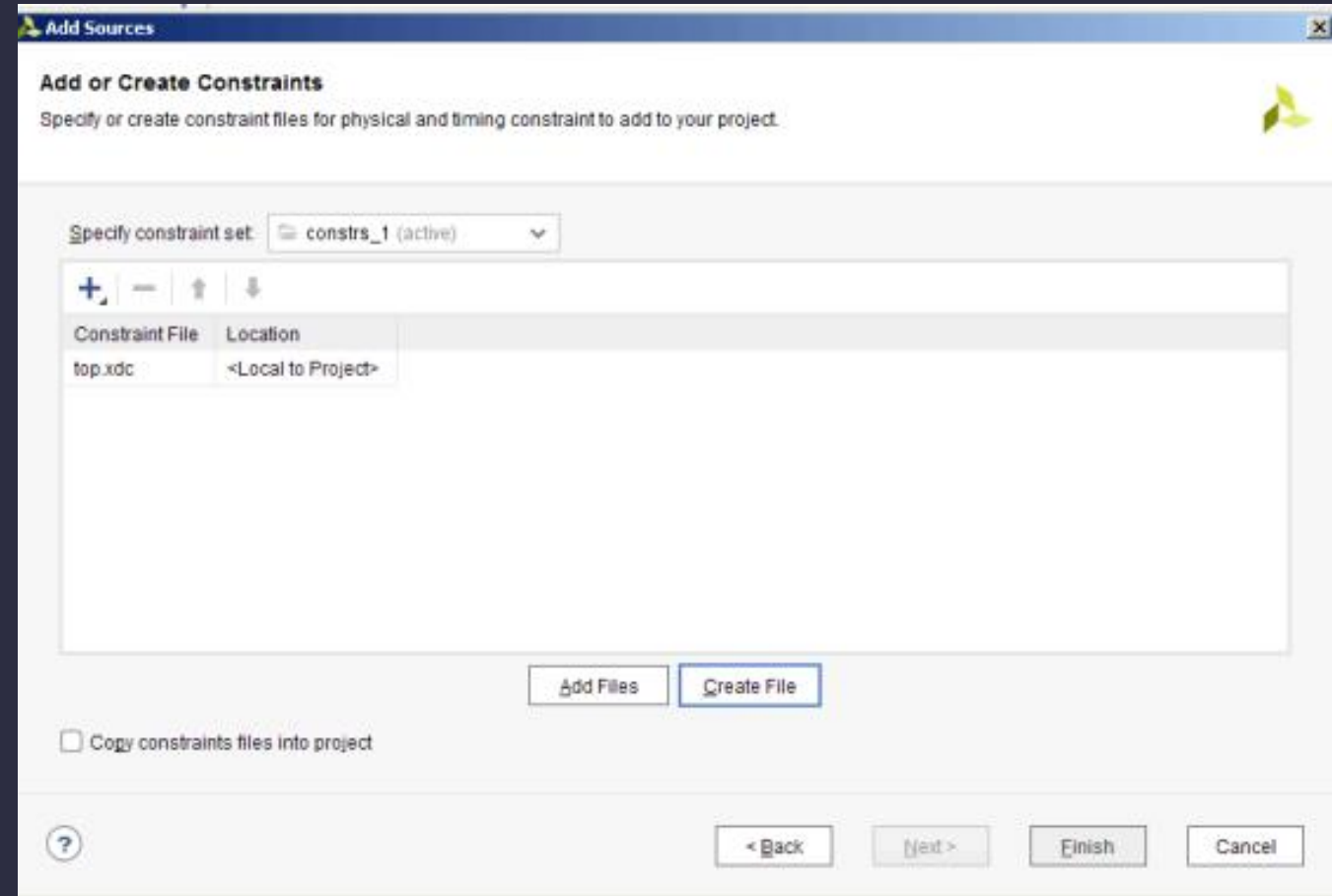
Kısıt (constraint) eklemek



Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı

Kısıt (constraint) eklemek

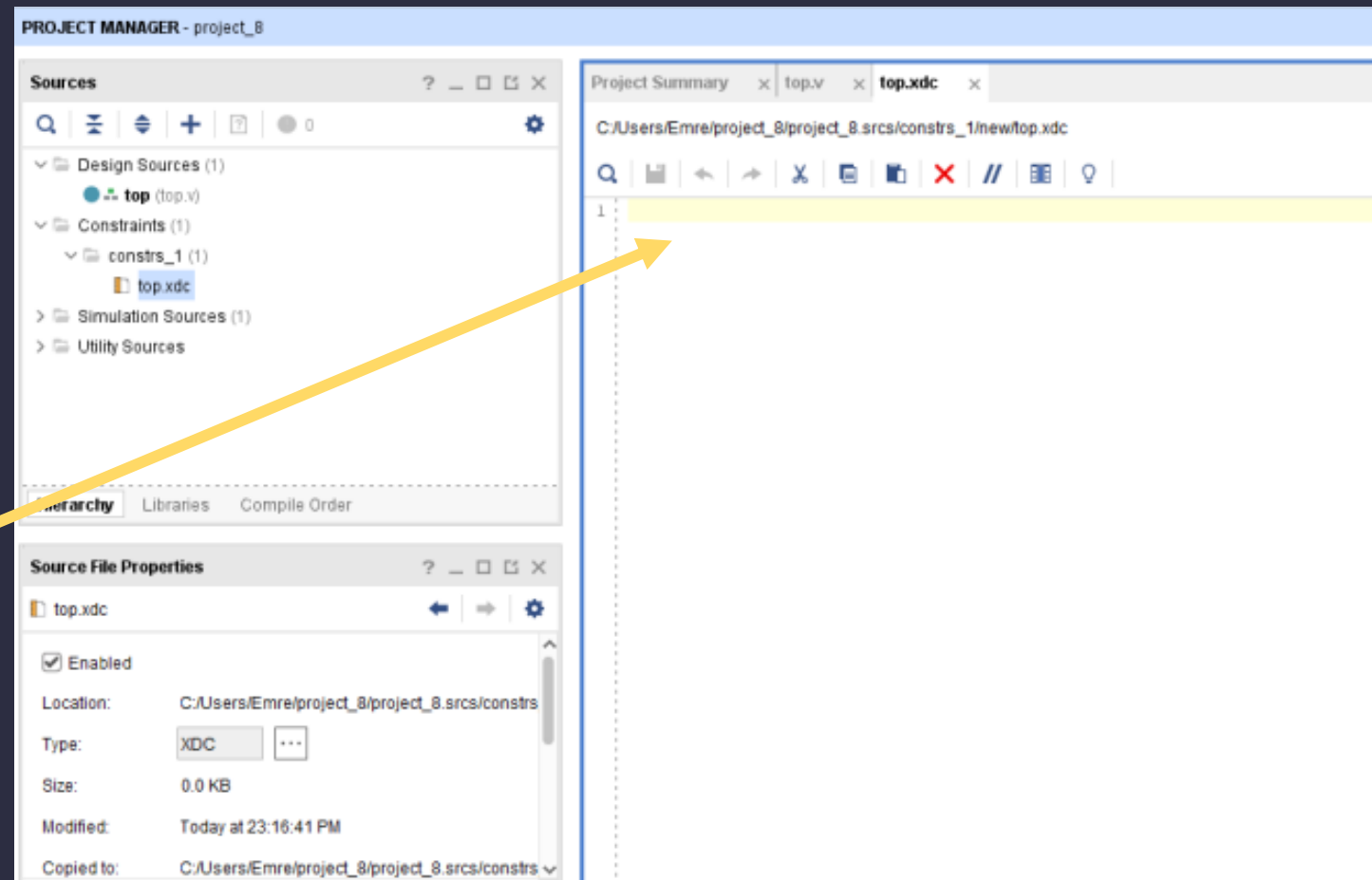


Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı

Kısıt (constraint) eklemek

Kısıtların yazılacağı dosya



Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı

Bitstream üretimi

▼ PROGRAM AND DEBUG

 [Generate Bitstream](#)

▼ Open Hardware Manager

Open Target

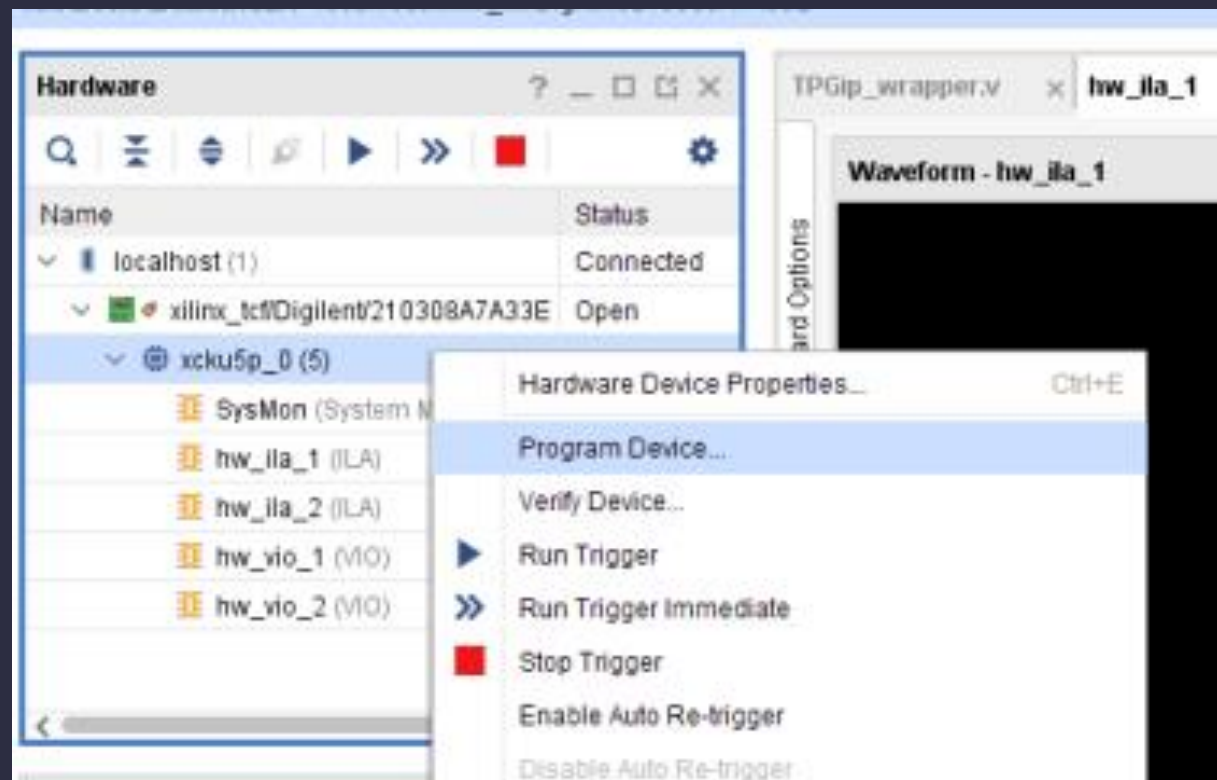
Program Device

Add Configuration Memory Devi

Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı

Bitstream üretimi



Verilog – Kombinasyonel Devreler

Vivado Tasarım Aracı

Bitstream üretimi

