

# Computer Architecture

## Week 9: RISC-V CPU Design



Fenerbahçe University



## Professor & TAs

Prof: Dr. Vecdi Emre Levent

Office: 311

Email: [emre.levent@fbu.edu.tr](mailto:emre.levent@fbu.edu.tr)

TA: Arş. Gör. Uğur Özbalkan

Office: 311

Email: [ugur.ozbalkan@fbu.edu.tr](mailto:ugur.ozbalkan@fbu.edu.tr)

# RISC-V Tabanlı İşlemci Tasarımı

Bu proje kapsamında başlangıç tasarım verilen bir RISC-V işlemcisinin ALU ve instruction decoder blokları temel SystemVerilog dili özellikleri kullanılarak tasarım ve doğrulama çalışmaları yapılacaktır.

# RISC-V Tabanlı İşlemci Tasarımı

## Proje Ekibi

- Proje 4 kişilik ekiplerden oluşacaktır.
- Her bir proje ekibinin bir sorumlusu olacaktır.
- Öğrenciler 4 kişilik kendi proje ekiplerini ve proje sorumlusunu belirlemelidirler.

# RISC-V Tabanlı İşlemci Tasarımı

## Proje Ekibi

- Ekiplerin kurulması ve proje sorumlusunun belirlenmesi en geç 23.4.2021 tarihine kadar tamamlanmalıdır.
- Ekip sorumluları, Blackboard'da açılmış olan "Proje Ekip Sorumluların Takımlarını Bildirmesi" başlığının altına, ekip üyelerinin isimlerini göndermelidirler.

# RISC-V Tabanlı İşlemci Tasarımı

## Kullanılacak Araçlar

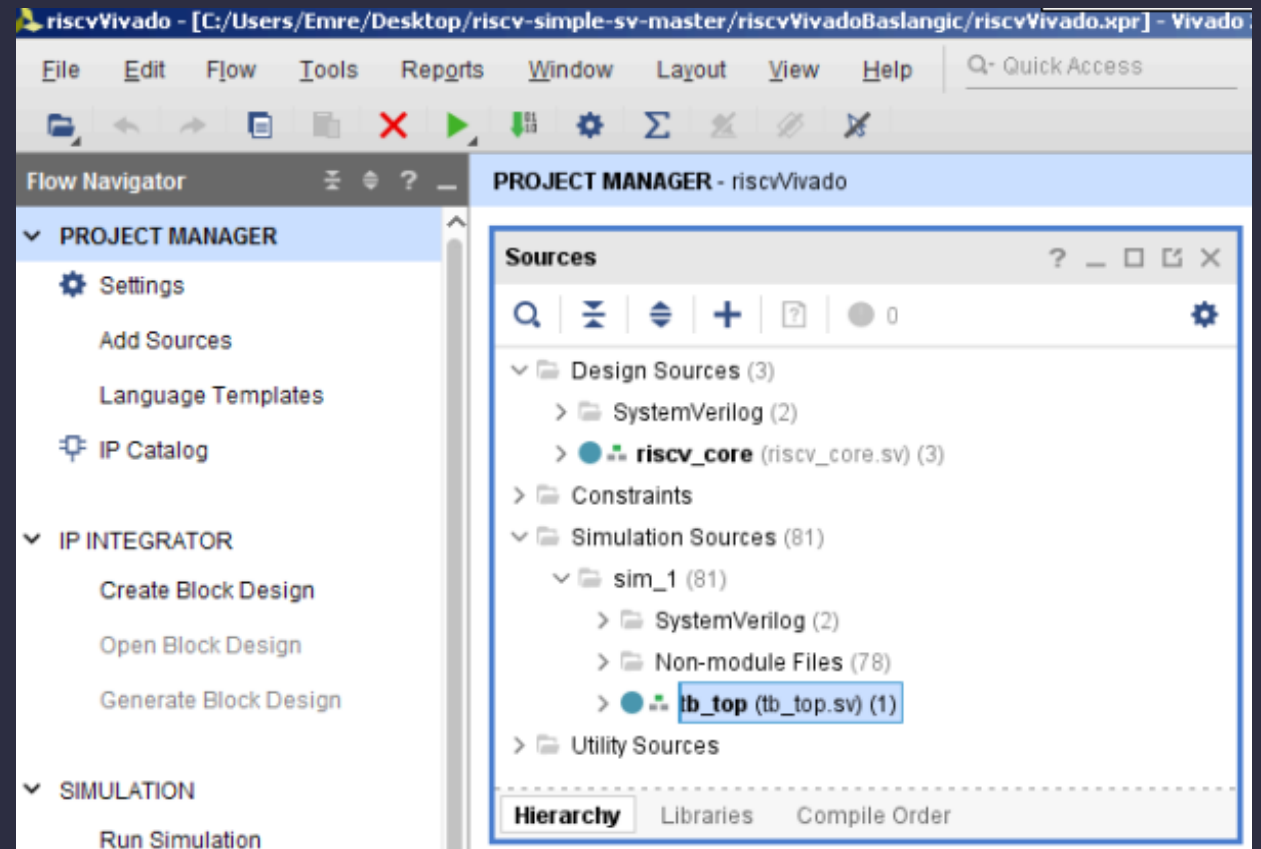
- Xilinx Vivado Design Suite

Xilinx Vivado Design Suite, FPGA geliştirme kartları üzerinde çalışmalar yapmak için gerekli olan tasarımı oluşturmak için kullanılmaktadır. Verilog, VHDL vb.. donanım tasarım dillerini alarak, FPGA'ye konfigüre edilebilecek (Xilinx firması FPGA'leri için .bit uzantılı dosyalar) tasarım dosyasını oluşturur.

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri

Proje açıldığında Şekil'de gösterilen başlangıç dosyaları görülmelidir. Design source altında başlangıç tasarımının tepe modülü olan riscv\_core system verilog dosyası, simulation source bölümünde ise tb\_top systemverilog dosyası bulunmaktadır.



# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri

Tasarımda tanımı yapılmış ancak gerçekleştirilmesi yapılmamış modüller bulunmaktadır.

Bunlar;

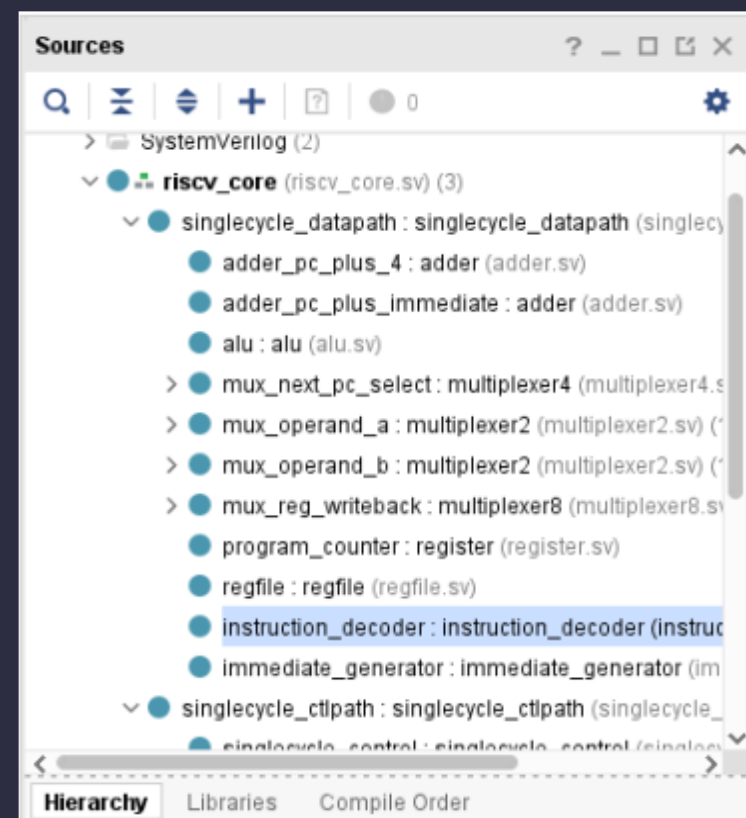
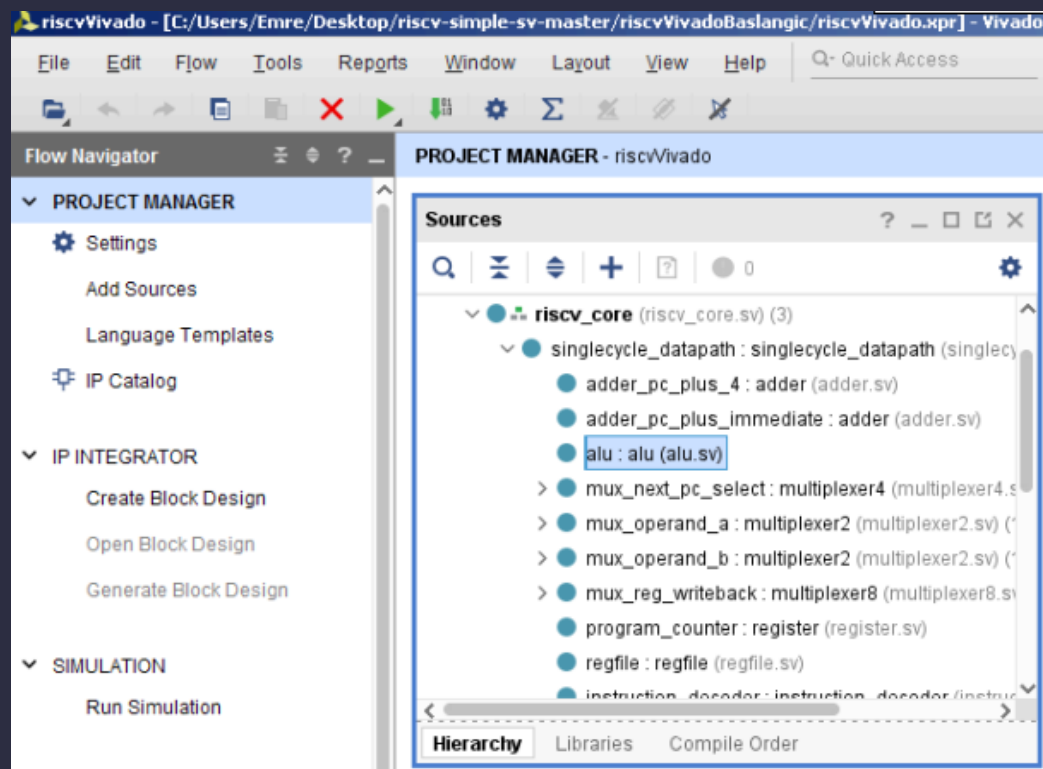
- ALU
- Instruction Decoder

modülleridir.



# RISC-V Tabanlı İşlemci Tasarımı

Aşağıdaki şekillerde bu modüllerin, proje hiyerarşisindeki konumları verilmiştir.



# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri - ALU

Aşağıda ALU ünitesinin giriş ve çıkış sinyalleri gösterilmektedir.

```
module alu (  
    input          [4:0]  alu_function,  
    input signed   [31:0] operand_a,  
    input signed   [31:0] operand_b,  
    output logic   [31:0] result,  
    output                    result_equal_zero  
);
```

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri - ALU

İşlemcinin ALU'sunun destekleyeceği 11 adet işlem vardır.

Bu işlemlerden hangisinin yapılacağı `alu_function` girişinden gelmektedir.

İşlemlere göre `a` ve `b` sayıları, `result` isminde sonuç çıkışı ve sonuç eğer sıfır ise, ayrı bir çıkış olarak sonucun sıfır olması durumunda 1 olan bir çıktı vardır.

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri - ALU

Aşağıdaki tabloda ALU'nun desteklediği işlemler ve operasyon kodları verilmektedir.

ALU_ADD	5'b00001
ALU_SUB	5'b00010
ALU_SLL	5'b00011
ALU_SRL	5'b00100
ALU_SRA	5'b00101
ALU_SEQ	5'b00110
ALU_SLT	5'b00111
ALU_SLTU	5'b01000
ALU_XOR	5'b01001
ALU_OR	5'b01010
ALU_AND	5'b01011

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri - ALU

Operasyonların açıklamaları aşağıda listelenmektedir.

- ADD:  $A + B$
- SUB:  $A - B$
- SLL:  $A \ll B$
- SLR:  $A \gg B$
- SRA:  $A \ggg B$
- SEQ:  $A == B$
- SLT:  $A < B$
- SLTU:  $\$unsigned(A) < \$unsigned(B)$
- XOR:  $A \wedge B$
- OR:  $A | B$
- AND:  $A \& B$

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri – Instruction Decoder

Aşağıda Instruction Decoder ünitesinin giriş ve çıkış sinyalleri gösterilmektedir.

```
module instruction_decoder(  
    input  [31:0] inst,  
    output [6:0]  inst_opcode,  
    output [2:0]  inst_func3,  
    output [6:0]  inst_func7,  
    output [4:0]  inst_rd,  
    output [4:0]  inst_rs1,  
    output [4:0]  inst_rs2  
);
```

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri – Instruction Decoder

Bu modülde giriş olarak 32 bitlik instruction word'u alınmaktadır.

Çıkışta ise instruction'un parse edilmiş hali, yani decode edilmiş hali çıkış olarak verilmektedir.

- Opcode, instruction'un ilk 7 bitini yani [6:0]'ı temsil etmekte
- Func3, instruction'un 14-12 bitleri arasını [14:12];
- Func7, instruction'un 31-25 bitleri arasını [31:25];
- Rd, instruction'un 11-7 bitleri arasını [11:7];
- RS1, instruction'un 19-15 bitleri arasını [19:15];
- RS2, instruction'un 24-20 bitleri arasını [24:20];

Buna göre instruction sinyalini parçalayarak ilgili sinyallerin üzerlerine atama yapılmalıdır.

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri – Proje Testi

Projenin doğru çalıştığıının test edilmesi için başlangıç test kodları verilmiştir.

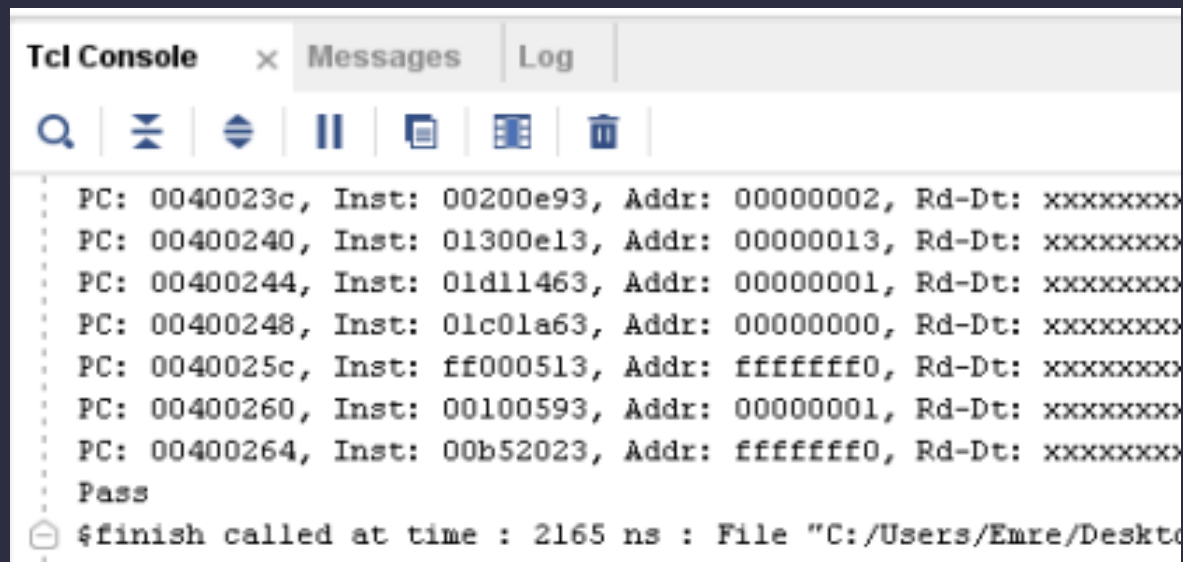
Test kodunda daha önceden hazırlanmış olan bir CPU test uygulamasının makine diline döndürülmüş halini, işlemciye besleyip sonucunu kontrol eden bir uygulama bulunmaktadır.

Tasarım simülasyonu başlatılıp play tuşuna basıldığında, en fazla 10000 cycle bekleyip, sonuç hesaplanmış ise aşağıdaki şekilde görülebilen pass çıktısını vermektedir. Aksi takdirde fail çıktısı verecektir.



# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri – Proje Testi



```
Tcl Console x Messages Log
[Icons: Search, Zoom In, Zoom Out, Stop, Copy, Paste, Delete]
PC: 0040023c, Inst: 00200e93, Addr: 00000002, Rd-Dt: xxxxxxxx
PC: 00400240, Inst: 01300e13, Addr: 00000013, Rd-Dt: xxxxxxxx
PC: 00400244, Inst: 01d11463, Addr: 00000001, Rd-Dt: xxxxxxxx
PC: 00400248, Inst: 01c01a63, Addr: 00000000, Rd-Dt: xxxxxxxx
PC: 0040025c, Inst: ff000513, Addr: ffffffff0, Rd-Dt: xxxxxxxx
PC: 00400260, Inst: 00100593, Addr: 00000001, Rd-Dt: xxxxxxxx
PC: 00400264, Inst: 00b52023, Addr: ffffffff0, Rd-Dt: xxxxxxxx
Pass
$finish called at time : 2165 ns : File "C:/Users/Emre/Desktop
```

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri – Notlandırma ve Proje Teslimi

- Projenin iki ana değerlendirme kriteri vardır. Her iki kriter 50 şer puandır.
- İlk kriter RISC-V işlemcisi komutlarının (instructions) doğru çalıştırılmasıdır.
- İkinci kriter ise Proje Teslim Dokümanı ve Sunumdur.

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri – Notlandırma ve Proje Teslimi

- Projenin teslimi için aşağıdaki adımların gerçekleştirilmesi gerekmektedir. İstenen dosyaları sadece proje ekip sorumlusunun getirmesi, Blackboard ve Github (Çok yaygın bir açık kaynak kod paylaşım platformudur)'a yüklemelidir.
- Proje ekip sorumlusunun Blackboard üzerinden açılmış olan “Proje Teslim” sayfasına aşağıdaki dosyaların yüklenmesi gerekmektedir.

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri – Notlandırma ve Proje Teslimi

- SystemVerilog RTL Tasarım (Tamamlanmış RTL dosyaları .v uzantılı)
- Hazırlanan powerpoint sunum dosyası (.ppt uzantılı dosya)
- Proje Teslim Dokümanı (Word formatında yüklenmelidir)
- Dokümanın alt başlıkları doldurulmalıdır

# RISC-V Tabanlı İşlemci Tasarımı

## Tasarım Gereksinimleri – Notlandırma ve Proje Teslimi

- Kaydedilen powerpoint sunum video'su youtube'a yüklenip, adresi, dokümanın sonuçlar bölümündeki açılmış yere link'i yazılmalıdır (Video'nun herkes'e görünür olmamasını istiyorsanız, youtube'a yükledikten sonra liste dışı seçeneğini seçerek, sadece link'e sahip olan kişilerin görmesini sağlayabilirsiniz).
- Blackboard'a yüklenen tüm dosyalar (Tamamlanmış RTL tasarım, ppt uzantılı sunum dosyası ve Proje Teslim Dokümanını (PDF formatında)), github.com sitesine üye olup, yüklenip, dokümanın sonuçlar bölümündeki yere link'i yazılmalıdır.