



TOBB Ekonomi ve Teknoloji Üniversitesi
Bilgisayar Mühendisliği Bölümü
Elektrik ve Elektronik Mühendisliği Bölümü

2 Mart 2025
BİL 265/264L - Mantıksal Devre
Tasarımı ve Laboratuvarı
2024 - 2025 Öğretim Yılı
Bahar Dönemi
Lab Sınavı

AÇIKLAMALAR:

1. Sınavı çözmeye başlamadan önce tüm açıklamaları ve soruları okuyun. Sınavda toplam 3 sayfa, 2 soru var ve soruların toplam değeri **100** puandır. Bütün soruların değeri köşeli ayraç ile belirtilmiştir. Sınav süresi **150** dakikadır.
2. Sınav sırasında kodlarınızla alakalı soru kabul edilmeyecektir.
3. Sınav esnasında internet ve tarayıcı kullanımı yasaktır. Bilgisayarda Xilinx Vivado programı dışında hiçbir program **KESİNLİKLE** açık olamaz.
4. İnternete bağlı olduğu veya herhangi bir tarayıcısı açık olduğu görülen kişilerin sınavları geçersiz sayılacak ve kopya olarak değerlendirilip gerekli işlemler yapılacaktır.
5. Sınavda **kapı seviyesinde modelleme istenen sorularda** 'for', 'assign', 'if' ya da '&' gibi Verilog operatörleri kullanmanız **KESİNLİKLE** yasaktır. Kullanıldığı takdirde sorunuz **geçersiz** sayılacaktır.
6. Sınav boyunca her türlü araç/gereç ve kaynak kullanımı yasaktır. (hesap makinesi, akıllı saat, telefon, pdf dosyaları vb.)
7. Sonucu yanlış olan yanıtlar puan alamayabilir. Açıklamasız kod yazmamaya özen göstermeniz alacağınız puanı artıracaktır.
8. <Projenizin bulunduğu dizin>\<Proje ismi>\<Proje ismi>.srcs\sources_1\new → dizininde yazdığınız ".v" uzantılı dosyaları bulabilirsiniz. Simülasyon dosyalarını ise aynı uzantıda .srcs'den sonra \sim_1 klasöründe bulabilirsiniz.
9. **10. talimata uyulmaması ve dosya isimlerinin yanlış yazılması durumlarında toplam puanınız üzerinden 20 puan kırılabacaktır.**
10. Dosya gönderimi için sorularda belirtilen ".v" uzantılı dosyalarınızı "isim_soyisim_numara_labsinavi" isimli bir klasöre attıktan sonra klasörü sıkıştırınız ve sınav sırasında gözetmenin getireceği USB veya Uzak'a yüklemeye hazır olacak şekilde bekleyiniz.

Sınavda Göndermeniz Gereken .v dosyaları: (Gönderim yapmak istemediğiniz soruları eklemek zorunda değilsiniz.)

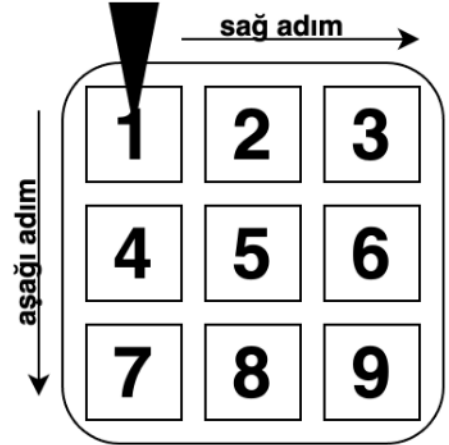
- tahmin.v
- kapisma.v
- durum.v

1. [70 Puan] Sayı Tahmin Oyunu

a) [40 Puan] Oluşturacağınız modüle "tahmin" (oluşacak dosya "tahmin.v") ismini verin. İstenen modülü Verilog dilinde **kapı seviyesinde** tasarım kullanarak gerçekleştirin.

Şekilde görülen kareler üzerinde bir sayı tahmin oyunu oynanmaktadır. Oyuncu verilen adım sayılarıyla imleci sağ ve aşağı yönde hareket ettirip sisteme giriş olarak verilen sayıyı tahmin etmeye çalışmaktadır. Bu tahmin oyunu için kurallar aşağıdaki gibidir:

- İmleç başlangıçta "1" sayısını göstermektedir, gelen sag_adim ve asagi_adim girişlerine göre hareket ettirilmektedir ve bu hareket sonunda imlecin denk geldiği sayı oyuncunun tahmin ettiği sayıdır.
- İmleç her bir adım sayısı için birer birim ilgili yönde hareket etmektedir. Örneğin eğer sag_adim=1 ve asagi_adim=2 şeklinde geldiyse imleç 8 sayısında durur, yani oyuncunun tahmin ettiği sayı 8 olur fakat örneğin eğer iki yönde de adım sayısı 0 ise imleç 1 sayısında kalır ve tahmin edilen sayı 1 olur.
- Adım sayısı şekilden de görüleceği üzere maksimum 2 olabilir fakat eğer adım sayısı herhangi bir yönde 3 gelirse yine ilgili yönde (sağ veya aşağı) 2 birim ilerlenir. Örneğin sag_adim=3 ve asagi_adim=1 geldiyse imleç 6 sayısında durur. (Bu durumda sag_adim=2 olsaydı da 6 sayısında duracaktı.)
- Tahmin edilen sayı (1-9 arasında) sayi_tahmin çıkışından verilir.
- Tahmin edilen sayı, giriş olarak verilen sayı ile karşılaştırılır ve eğer tahmin edilen sayı doğruysa tahmin_dogru çıkışından mantık-1 verilir.



Yazacağınız modülün giriş ve çıkışları aşağıdaki gibidir:

Devrenin girişleri:

sag_adim: 2 bitlik sağa kaç birim gidileceğini gösteren giriş sinyali

asagi_adim: 2 bitlik aşağı kaç birim gidileceğini gösteren giriş sinyali

sayi: 4 bitlik tahmin edilmesi gereken sayıyı belirten giriş sinyali

Devrenin çıkışları:

sayi_tahmin: 4 bitlik oyuncu tarafından tahmin edilen sayı çıkış sinyali

tahmin_dogru: 1 bitlik tahmin edilen sayının giriş olarak verilen sayıyla eşleşip eşleşmediğini (eşleşiyorsa mantık-1, eşleşmiyorsa mantık-0) belirten çıkış sinyali

Örnek: sag_adim=0, asagi_adim=1, sayi=5 → sayi_tahmin=4, tahmin_dogru=0

Örnek: sag_adim=2, asagi_adim=3, sayi=9 → sayi_tahmin=9, tahmin_dogru=1

b) [30 Puan] Oluşturacağınız modüle "kapisma" (oluşacak dosya "kapisma.v") ismini verin. İstenen modülü Verilog dilinde **davranışsal modelleme** kullanarak gerçekleştirin. Bu modülde a şıkında gerçekleştirdiğiniz "tahmin" modülünü alt modül-modüller olarak kullanmanız **zorunludur**.

Üç arkadaş a şıkında anlatılan tahmin oyununda kapismaktadır. Doğru sayı tahmininde bulunan oyuncular oyunu kazanmaktadır.

- Eğer 1. oyuncu tahmin oyununu kazandıysa kazanan cikisinden 1, eğer 2. oyuncu kazandıysa 2, eğer 3. oyuncu kazandıysa 3, eğer hiçbir oyuncu kazanmadıysa 0 çıkışı verilir.
- Eğer birden fazla oyuncu oyunu kazandıysa; bu durumda kazanan çıkışı için 1. oyuncu > 3. oyuncu > 2. oyuncu önceliği vardır. Örneğin 2. oyuncu ve 3. oyuncu oyunu kazanmış, 1. oyuncu kazanamamış ise kazanan çıkışından önceliği olduğu için 3 verilir.
- Oyuncuların tahmin ettikleri sayılara göre toplam puan hesaplanmaktadır. Kazanan oyuncuların tahmin ettikleri sayılar toplama pozitif olarak etki ederken kaybeden oyuncuların tahmin ettikleri sayılar negatif olarak etki etmektedir. Toplam puan negatif olamaz, **minimum 0** olabilir. Örneğin 1. oyuncunun 5, 2. oyuncunun 7, 3. oyuncunun 5 tahmin ettiği ve 5 sayısının doğru tahmin olduğu durumda toplam puan 5 - 7 + 5 şeklinde hesaplanır ve 3 olur. Fakat bu örnekte doğru tahmin 5 olmasaydı toplam puan negatif olacağından 0 olarak kabul edilirdi.

Yazacağınız modülün giriş ve çıkışları aşağıdaki gibidir:

Devrenin girişleri:

sag_adimlar: 6 bitlik oyuncuların sağa kaç birim gideceğini gösteren giriş sinyali (En anlamlı 2 biti 1. oyuncu, ortadaki 2 biti 2. oyuncu, en anlamsız 2 biti ise 3. oyuncu için)

asagi_adimlar: 6 bitlik oyuncuların aşağı kaç birim gideceğini gösteren giriş sinyali (En anlamlı 2 biti 1. oyuncu, ortadaki 2 biti 2. oyuncu, en anlamsız 2 biti ise 3. oyuncu için)

sayi: 4 bitlik tahmin edilmesi gereken sayıyı (tüm oyuncular için) belirten giriş sinyali

Devrenin çıkışları:

kazanan: 2 bitlik kazanan oyuncuyu belirten çıkış sinyali

toplam_puan: 5 bitlik oyuncuların toplam puanını gösteren çıkış sinyali

Örnek: sag_adimlar=6'b00_00_00, asagi_adimlar=6'b00_00_00, sayi=4'b0001

→ kazanan=1, toplam_puan=3

Örnek: sag_adimlar=6'b00_00_00, asagi_adimlar=6'b00_00_00, sayi=4'b0010

→ kazanan=0, toplam_puan=0

Örnek: sag_adimlar=6'b01_10_11, asagi_adimlar=6'b01_10_11, sayi=4'b1001

→ kazanan=3, toplam_puan=13

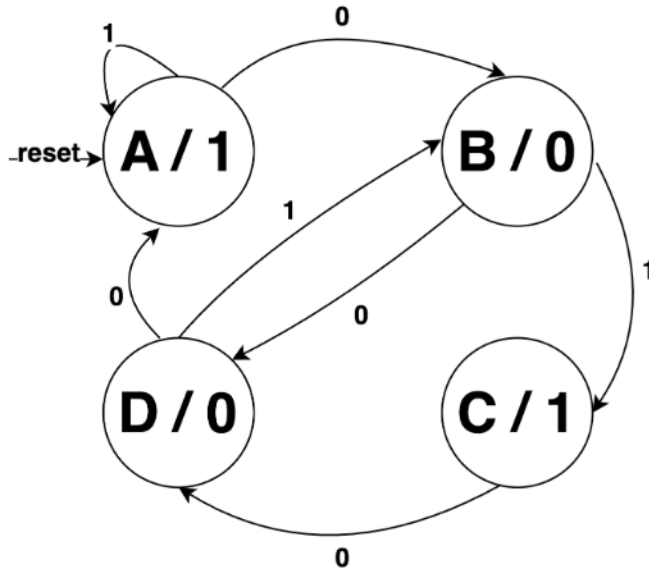
Örnek: sag_adimlar=6'b00_10_11, asagi_adimlar=6'b10_10_11, sayi=4'b0111

→ kazanan=1, toplam_puan=0

2. [30 Puan] Durum Makinesi

Oluşturacağınız modüle "durum" (oluşacak dosya "durum.v") ismini verin. İstenen modülü Verilog dilinde **davranışsal modelleme** kullanarak gerçekleştirin.

Şekilde verilen durum diyagramını Verilog donanım tanımlama dilinde gerçekleştirmeniz beklenmektedir. A, B, C, D harfleri durumları, durumların içindeki sayılar çıkışları, oklar durum geçişlerini ve okların üstündeki sayılar ise girişleri temsil etmektedir. Varsayılan durum olarak başlangıç durumu A durumudur, reset geldiğinde yine A durumuna geçilir. Her saat vuruşunda girişlere uygun durum geçişleri yapılır ve ilgili çıkışlar dışarı verilir.



Yazacağınız modülün giriş ve çıkışları aşağıdaki gibidir:

Devrenin girişleri:

saat: 1 bitlik saat sinyali

reset: 1 bitlik reset sinyali (senkron)

giris: 1 bitlik durum makinesindeki girişi belirten sinyal

Devrenin çıkışları:

cikis: 1 bitlik durum makinesindeki çıkışı belirten sinyal