



MÜHENDİSLİK VE MİMARLIK FAKÜLTESİ

SINAV KAĞIDI

2019/2020 EĞİTİM – ÖĞRETİM YILI/2020/2021 ACADEMIC YEAR
BAHAR DÖNEMİ/SEMESTER

Öğrencinin/Student's

Adı Soyadı/Name, Surname :

Numarası/Number :

Bölüm-Program /Department-Programme :

İmzası/Signature :

Kullanılan Kağıt Sayısı/Number of Papers Used:

Toplam Not – Paraf/Total Credit - Initials

DERS Course	Adı/Name : Mantıksal Sistem Tasarımı	SINAV Exam	Tarih/Date : 04.12.2020								
	Kodu/Code : BLM201		Süresi/Duration : 120 Dk /Min								
	Sorumlusu/Lecturer : Dr. Öğr. Üyesi. Vecdi Emre Levent		Türü/Type : Vize / Midterm								
Soru Numarası/Numbers of the Questions	1	2	3	4	5	6	7	8	9	10	
Alınan Not/Scored Points											
SINAV KURALLARI/Exam Rules : 1. İstedığınız sorudan başlayabilirsiniz. 2. İlk 30 dakika soru sormak yasaktır.											

*Her sorunun puan değeri rakamsal olarak yanına belirtilmelidir./The points for each question must be stated next to the question.

SORULAR/Questions



Soru 1 (40 Puan):

myModule isimindeki bir modülü test edecek bir testbench kodu yazınız.

myModule A, B, C, D bir bitlik girişleri ve 1 bitlik Z isminde çıkışı vardır. Modülde tüm girişler AND işlemi yapılarak dışarı çıkartılmaktadır.

Testbench modülünü tb_myModule olarak isimlendiriniz.

Testbench modülü, myModule modülüne aşağıda verilen sayıları A, B, C, D girişlerine 10 ns aralıklarla göndermelidir.

- 0. nanosaniye; A = 1, B = 1, C = 1, D = 0
- 10. nanosaniye; A = 0, B = 0, C = 1, D = 0
- 20. nanosaniye; A = 0, B = 0, C = 0, D = 0
- 30. nanosaniye; A = 1, B = 1, C = 1, D = 1

Soru 2 (40 Puan):

Aşağıda özellikleri verilen ALU'nun (Arithmetic Logic Unit) tasarımı Verilog dilinde yapılacaktır.

Geliştirilecek olan modülün ismi myALU olacaktır.

Giriş olarak 8'e bitlik i_a, i_b ve 3 bitlik i_op olacaktır.
Çıkışta ise 16 bitlik o_c sinyali üretilmektedir.

i_op sinyalinin değerine göre ALU tasarımı farklı operasyonlar yapacaktır. i_op'un değerine göre yapılacak operasyonlar aşağıdaki tabloda özetlenmiştir.

i_op	Operasyon
0	Çıkarma ($i_a - i_b$)
1	AND ($i_a \& i_b$)
2	Toplama ($i_a + i_b$)
3	OR ($i_a i_b$)
4	XOR ($i_a \wedge i_b$)
5	MOV ($o_c = i_b$, Yani sonuç işlem yapılmadan doğrudan i_b olur)
Diğer	Diğer durumlarda $o_c = 16' b10101010101010$

Soru 3 (20 Puan):

Neden tüm devreler, Kombinasyonel (Combinational) devreler ile yapılamamaktadır?

Bazı devrelerin sadece kombinasyonel devreler ile yapılamamasını açıklarken, Ardışık (Sequential) devrelerin ne olduğu, nasıl çalıştığı ve sorunun çözümündeki rolü ile birlikte ifade ediniz.