

İTÜ

Bilgisayar ve Bilişim Fakültesi
BLG222 – Bilgisayar Organizasyonu



Dönem Sonu Sınavı

26.05.2011

Süre: 120 dk.

Ad Soyad: _____ Numara: _____ İmza: _____

Açıklamalar:

1. Bu sınav sonucu dönem notunuzun %50'sini belirleyecektir.
2. Kitap ve notlar açıktır, kitap ve notların paylaşımı mümkün değildir.
4. **Her soruyu bulunduğu sayfa üzerine çözün.**
5. Sınav 100 puan üzerindedir.

SORU 1: [30 puan]

Kitabın 5. bölümünde gösterilen temel bilgisayar ve komut seti için:

- a) DR saklayıcısının denetim sinyallerine (LD(DR), INC(DR) ve CLR(DR)) ait lojik ifadeleri yazınız.
- b) PC saklayıcısının denetim sinyallerine (LD(PC), INC(PC) ve CLR(PC)) ait lojik ifadeleri yazınız.
- c) Belleğin yaz (write) denetim sinyaline ait lojik ifadeyi yazınız.

İTÜ

Bilgisayar ve Bilişim Fakültesi
BLG222 – Bilgisayar Organizasyonu



Dönem Sonu Sınavı

26.05.2011

Süre: 120 dk.

Ad Soyad: _____ Numara: _____ İmza: _____

SORU 2: [30 puan]

Kitabın 5. Bölümünde gösterilen temel bilgisayara yığın registerinin (SP) eklendiğini varsayın. Komut setine ise aşağıda verilen 2 komut eklenmiştir:

PUSH: $AR \leftarrow SP$
 $M[AR] \leftarrow DR, SP \leftarrow SP + 1$
POP: $SP \leftarrow SP - 1$
 $AR \leftarrow SP$
 $DR \leftarrow M[AR]$

- BSA komutunu, alt programa dallanıldığında dönüş adresinin yığında saklanmasını sağlayacak şekilde değiştiriniz. Yeni BSA komutunu RTL ile yazınız.
- Alt programın sonunda kullanılmak üzere RET adında yeni bir komut eklenecektir. Bu komut, yığındaki dönüş adresini alıp, programın kaldığı yerden devam etmesini sağlayacaktır. RET komutunu RTL ile yazınız.
- Yeni tasarımın kitaptaki tasarıma göre olumlu ve olumsuz yönlerini yazınız.

İTÜ

Bilgisayar ve Bilişim Fakültesi
BLG222 – Bilgisayar Organizasyonu



Dönem Sonu Sınavı

26.05.2011

Süre: 120 dk.

Ad Soyad: _____ Numara: _____ İmza: _____

SORU 3: [40 puan]

Derste anlatılan mikroprogramlı bilgisayar mimarisine, sadece bellek üzerinde işlem yapan ve 4 bitlik komut kodu $(A)_{16}$ olan **MOD** komutu eklenecektir. **A** ve **B** 16 bitlik işaretli sayı olmak üzere **MOD** komutu **A MOD B** işlemini yapmaktadır.

A	B	A MOD B
$(15)_{10}$	$(2)_{10}$	$(1)_{10}$
$(15)_{10}$	$(8)_{10}$	$(7)_{10}$
$(108)_{10}$	$(32)_{10}$	$(44)_{10}$

B sayısı, sadece $2^n = \{1, 2, 4, 8, \dots, 2^{15}\}$ değerlerinden bir tanesi olabilmektedir. **A** ve **B** sayıları bellekte önce **B** sonra **A** sayısı olacak biçimde arka arkaya bulunmak zorundadır. Komut çalıştıktan sonra hesaplanan mod değeri, **A** sayısının bulunduğu bellek gözüne yazılacaktır.

Komut : MOD 700

Önce

	...
B → $(700)_{16}$	$(0008)_{16} = (8)_{10}$
A → $(701)_{16}$	$(000E)_{16} = (15)_{10}$
	...

Sonra

	...
$(700)_{16}$	$(0008)_{16} = (8)_{10}$
$(701)_{16}$	$(0007)_{16} = (7)_{10}$
	...

Yukarıda ayrıntıları verilen **MOD** komutu için gerekli mikroprogramı, mikroprogram belleğindeki adres yerleşimlerini de göstererek yazınız..

NOT : F3 için **AYRILMIŞ (RESERVED)** olarak belirtilen $(111)_2$ mikrokodunun **INCAR** ($AR \leftarrow AR + 1$) işlemini gerçekleştirecek şekilde tasarım yaptığınızı varsayınız.