

BLM3061 Mikroışl. Sist. ve Assmb. D

ÖDEV 2

8086 Bellek Haritalama & Adres Çözümleme

Soru: Aşağıdaki devre bileşenlerini kullanarak (ihtiyaca göre mantık kapıları eklenebilir) 8086 mikroışlemcisi için istenen bellek haritasını oluşturacak adres çözümleme devresini tasarlayıp Proteus benzetim ortamında çalıştırınız. (100p)

Bileşenler:

1. 8086 Mikroışlemci	x	1 tane
2. 74273 Sekizli D Tipi Flip-Flop	x	3 tane
3. 74LS138 Dekoder	x	2 tane
4. 27C128 EPROM (16KB)	x	4 tane
5. 62256 SRAM (32KB)	x	2 tane

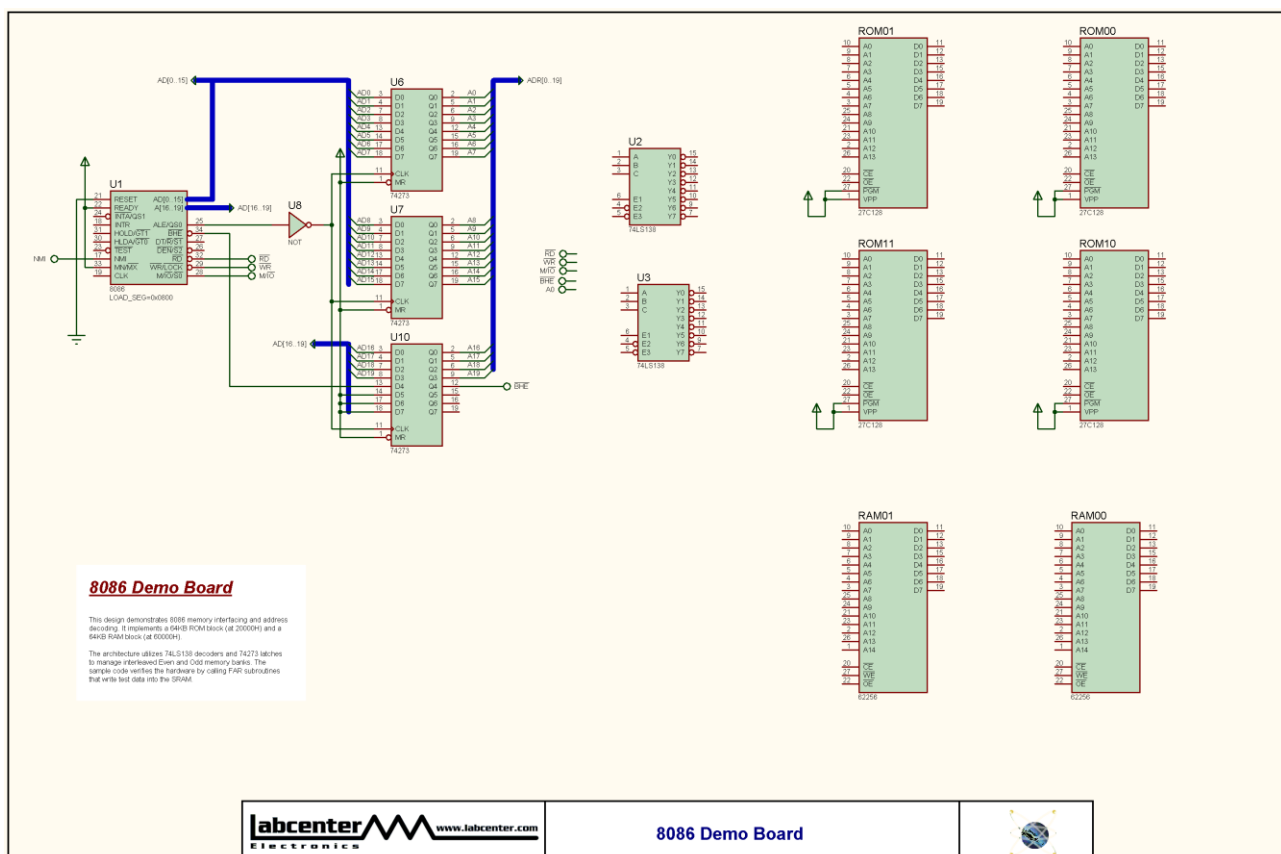
İstenenler:

- 8086 mikroışlemcili bir sistemde, 64KB ROM ve 64KB RAM bloklarını belirli başlangıç adreslerine yerleştirecek adres çözümleme devresinin tasarlanması beklenmektedir.
- Tasarımda 74LS138 dekodeerleri ve mantık kapıları kullanılarak çift/tek bank yapısına uygun Chip Select (CS) sinyalleri üretilmeli ve sağlanan binary dosyaları ROM'lara yüklenmelidir.
- Hazırlanan donanımın doğruluğu, ROM üzerindeki uzak (FAR) alt yordamları çağırın bir Assembly kodu ile RAM üzerine veri yazdırılarak test edilmelidir.
- 8086 mikroışlemcili sistemde aşağıdaki bellek bloklarını belirtilen fiziksel adreslere yerleştiriniz:
 - ROM Bloğu (64KB):** 20000H fiziksel adresinden başlamalıdır. Bu blok, her biri 16KB kapasitesinde 4 adet **27C128 EPROM** kullanılarak oluşturulacaktır.
 - RAM Bloğu (64KB):** 60000H fiziksel adresinden başlamalıdır. Bu blok, her biri 32KB kapasitesinde 2 adet **62256 SRAM** kullanılarak oluşturulacaktır.Verilen devrede mikroışlemcinin, adres (ADR[0-19] yolunda), veri (AD[0-15] yolunda) ve kontrol (\overline{BHE} , \overline{WR} , \overline{RD} , M/\overline{IO} terminallerinde) uçlarının uygun şekilde ayrıştırılmış ve tutulmuş olduğunu varsayın. Çözüm için izole I/O haritalama (**isolated I/O map**) ve çift-tek banklar için ayrı adres çözümleme (**seperate bank decoder**) kullanmalısınız. (25p)
- Verilen ROM ve RAM bloklarını yukarıda bahsedilen koşullara verilen adres aralıklarına yerleştiren **adres çözümleme devresini tasarlayın**. Hafıza birimleri ile adres çözümleme devresi ve mikroışlemci arasında **gerekli tüm bağlantıları yapın**. (25p)
- Verilen **00.bin**, **01.bin**, **10.bin** ve **11.bin** dosyalarını ilgili ROM entegrelerine (ROM00, ROM01, ROM10, ROM11) yükleyiniz.
- Bu dosyalar yardımıyla biri 2000H:0000H adresinde diğeri ise 2800H:0002H adresinde iki adet FAR tipinde alt yordam oluşturulmuştur. Kodunuz ENDLESS: JMP ENDLESS döngüsü içerisinde bu yordamları sürekli çağırılmalıdır. Bu yordamlar RAM'in düşük adreslerine belirli değerler yazacaktır. Simülasyonu duraklatıp RAM içeriğini kontrol ederek başarılı bir şekilde tutulduğunu doğrulayınız. (25p)
- Tasarlanan adres çözümleme mantığını, hangi adres bitlerinin ($A_{19} - A_1$) hangi bellek birimini aktif ettiğini gösteren bir **Adres Çözümleme Tablosu** ile raporda (PDF) ve videoda açıklayınız. (25p)
- Ayrıca devre şemasında yapılan bağlantılar, yazılan kod ve sonuçlar kısa ve öz biçimde (üstünkörü değil ama gereksiz ayrıntıları da içermeyecek biçimde) videoda açıklanmalıdır.

Ödevin bazı bağlantıları eksik devre yapısı aşağıdaki görselde görülebilir:

[illegible]

Ödevin bazı bağlantıları eksik devre yapısı aşağıdaki görselde görülebilir:



Proteus şemasındaki isimlendirmeler aşağıdaki kurallara (şablonlara) göre yapılmalıdır.

- **RxMx0 (Even Bank Şablonu):** İsmi **0** ile biten tüm entegreler (ROM00, ROM10, RAM00) **Çift** adreslere yerleştirilir.
- **RxMx1 (Odd Bank Şablonu):** İsmi **1** ile biten tüm entegreler (ROM01, ROM11, RAM01) **Tek** adreslere yerleştirilir.
- **ROM0x (Düşük Blok):** ROM bloğunun düşük adreslerini (20000H - 27FFFH) oluşturan entegreler (ROM00 ve ROM01).
- **ROM1x (Yüksek Blok):** ROM bloğunun yüksek adreslerini (28000H - 2FFFFH) oluşturan entegreler (ROM10 ve ROM11).