

# BLM3061 Mikroişlemci Sistemleri ve Assembly Dili

## LAB 5

### *8255 7 Segment Gösterge Tasarımı*

#### **Deney:**

**Özet:** Bu deneyde 8086 mikroişlemcisi, 74HC373 latch ve 8255 programlanabilir çevre birimi kullanılarak 8 adet tuştan oluşan bir giriş ünitesi ile ortak anot 7 segment gösterge sürülecektir. Her tuşa basıldığında göstergede farklı bir rakam (0 7 arası veya isteğe göre seçilen 8 farklı sembol) belirecektir. Tuşlar Port B den okunacak, 7 segment kodları Port A üzerinden göstergeye gönderilecektir.

#### **Bileşenler:**

1. Ekte verilen Proteus taslak dosyasında tüm bileşenler vardır.
2. 8086 mikroişlemci, 74HC373 latch, 8255 programlanabilir çevre birimi, Ortak anot 7 segment gösterge, 8 adet buton (tuş), Gerekli sayıda direnç (7 segment için akım sınırlama dirençleri, Tuşlar için pull down ya da pull up dirençleri) Saat kaynağı, reset devresi ve güç bağlantıları (taslak devrede hazır)
3. İhtiyaca göre devreye ekleme çıkarma yapılabilir.

#### **İstenenler:**

- a. 8255 adresleme ve port yönlendirme ayarı:  
8255 temel adresinin belirtilmesi, Port A nın çıkış, Port B nin giriş, Port C nin ihtiyaç durumuna göre giriş veya çıkış olarak ayarlanması. Kontrol kelimesinin doğru biçimde hesaplanıp programa yazılması.
- b. Tuşların Port B ye doğru bağlanması ve okunması:  
8 adet tuşun Port B nin 8 bitine bağlanması. Tuş basılmadığında girişlerin kararlı lojik seviyede olması için dirençlerin kullanılması. Program içinde Port B nin okunup, okunan değerin kaydedilmesi.
- c. Tuş kodu ile gösterilecek rakamların eşlenmesi:  
Her bir tuş için hangi rakamın gösterileceğinin tablolaştırılması, ortak anot 7 segment için her rakamın segment kodlarının doğru hesaplanması. Assembly kodunda bu eşlemenin tablo veya karar yapıları ile uygulanması (örneğin lookup table, çoklu karşılaştırma)
- d. 7 segment göstergenin doğru sürülmesi:  
Seçilen rakamın segment kodunun Port A ya gönderilmesi. Tuşa basılı tutulduğunda rakamın göstergede sabit kalması. Farklı tuşa geçildiğinde göstergenin yeni rakama güncellenmesi.
- e. Program akışının döngü halinde çalışması ve yorum satırları:  
Programın sonsuz döngü içinde tuş okumaya ve gösterge güncellemeye devam etmesi. Kodda her ana bölüm için açıklayıcı yorum satırlarının bulunması (başlatma, port ayarı, tuş okuma, karar, çıkış yazma gibi)
- f. Doğru ve istenen düzen, dosya adlandırma ve çalışır simülasyon

- g. Her şıkkın doğru/olması gereken çıktısının nasıl olduğunu gösteren bir görsel ekleyiniz.

**Teslim Edilecek Dosyalar:**

1. Video Kaydı:

4 dakikayı geçmeyen bir video hazırlanmalıdır. Bu videoda, istenilen soruların cevapları ve yazılan kodun açıklaması detaylı bir şekilde anlatılmalıdır.

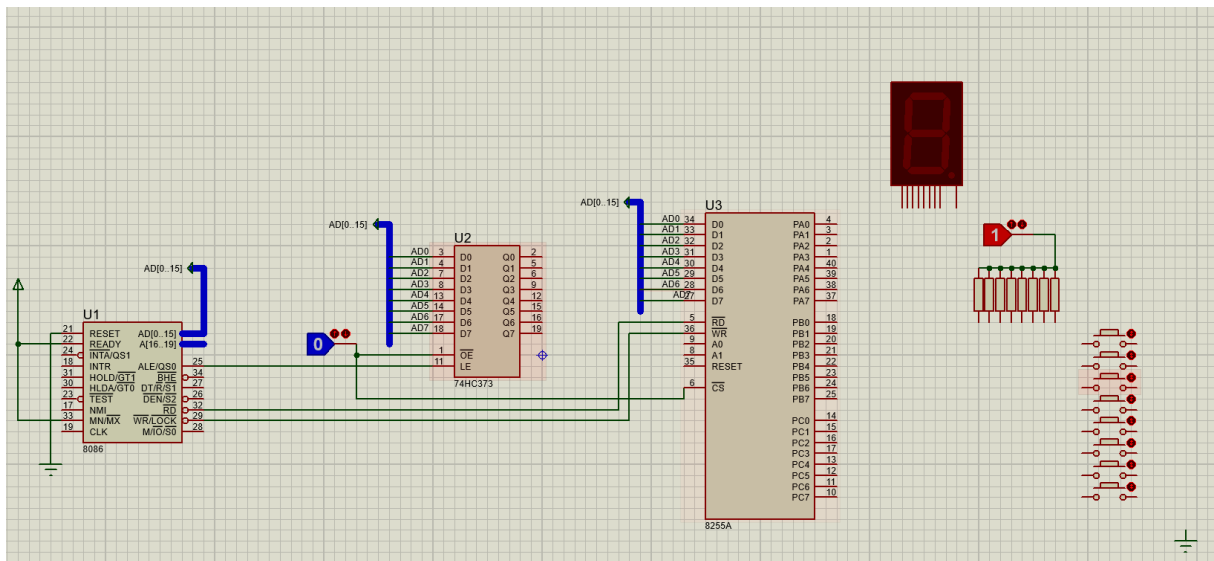
- ## 2. Proteus Projesi:

Proteus simülasyon ortamında çalışan devre ve kodun yer aldığı bir proje dosyası hazırlanmalıdır. Dosya ismi şu formatta olmalıdır: OğrenciNo\_IsimSoyisim.pdsprj Dosyada, Devrenin Proteus'ta çalışan hali, Mikroişlemciye bağlı kod ve devrenin tam bağlantıları bulunmalıdır.

- ### 3. Zip Dosyası:

Tüm dosyalar zip formatında bir arşiv haline getirilmelidir. Zip dosyasının ismi şu formatta olmalıdır: ÖğrenciNo\_İsimSoyisim.zip. Zip dosyası şu dosyaları içermelidir, Video kaydı, Proteus projesi dosyası (.pdsprj)

**Deneyin bazı bağlantıları eksik devre yapısı aşağıdaki görseldeki gibidir:**



Başarılar dilerim :)

**Firat F. Olcay**