

Bu bölüm artık öğrencinin:

- Elektriğin temelini (Aşama 1),
- Dijital düşünmeyi ve kapı sistemlerini (Aşama 2) bildiği varsayımıyla hazırlanmıştır.

Bu aşamada amaç, **dijital devrelerin “davranışsal” hale gelmesi**, yani öğrencinin artık sistem tasarlarlarken **koşullar, zamanlama, sayaç, kontrol akışı** gibi kavramları anlamasıdır.

Artık “mantık kapısı kurmak” yerine, “küçük bir dijital sistem tasarlamak” aşamasına geçiyoruz.

◆ AŞAMA 3: DİJİTAL MANTIKTAN MİKRODENETLEYİCİ MANTIĞINA

Oyun / Simülasyon: Digital Logic Sim

Süre: 2–3 Hafta (7 Ders)

Amaç: Dijital sistemlerin ardışık mantığını, sayaçları, zamanlayıcıları ve mikrodenetleyici mantığını öğretmek.

Seviye: Orta (Temel dijital bilgisi olan öğrenci)



Genel Akış

Ders	Konu	Ana Odak	Uygulama
1	Kombinasyonel vs Ardışık Mantık	Hafıza ve zaman farkı	Karşılaştırmalı devre
2	Clock (Saat) Sinyali	Zamanlama kavramı	LED yanıp sönme devresi
3	Sayaçlar (Counter)	Bit artırma, binary sayma	4-bit counter yapımı
4	Encoder & Decoder	Bilgiyi sıkıştırma ve çözme	7-segment sayı göstergesi
5	Multiplexer & Demultiplexer	Seçici veri hattı	LED seçici kontrol sistemi
6	Register & Hafıza Yapıları	Geçici veri tutma	8-bit veri kaydırma (shift register)
7	Mini Proje: Dijital Trafik Lambası	Tüm sistemleri birleştirme	Timer, latch, LED kombinasyonu

DERS 1 — Kombinasyonel vs Ardışık Mantık

Kazanım:

Öğrenci, bir sistemin “zamanla değişen” veya “anlık karar veren” türlerini ayırt etmeyi öğrenir.

Konu Özeti:

- **Kombinasyonel mantık:** Girişler neyse çıkış anında ona göre belirlenir (örnek: AND, OR).
- **Ardışık mantık:** Çıkış, sadece girişe değil, geçmiş duruma da bağlıdır (örnek: flip-flop, counter).
- Bu fark, dijital sistem tasarımında temel kavramdır.

Uygulama (Digital Logic Sim):

1. Basit bir AND kapısı (kombinasyonel) kur.
2. Bir SR latch (ardışık) kur.
3. Girişleri değiştir ve çıkışların davranış farkını gözlemle.

Değerlendirme:

- Hangi sistemde “önceki durum” önemliydi?
- Neden ardışık sistemler zamanla değişiyor?

DERS 2 — Clock (Saat) Sinyali

Kazanım:

Clock sinyaliyle ardışık sistemlerin zamanla çalıştığını kavrar.

Konu Özeti:

- Clock sinyali (osilatör), sistemin “ritmini” belirler.
- Flip-flop’lar ve sayaçlar clock darbesiyle çalışır.
- Dijital sistemlerde “senkronizasyon” için kullanılır.

Uygulama:

1. Clock sinyali (pulse generator) ekle.
2. LED’i clock sinyaliyle yanıp söner hale getir.
3. Farklı frekans değerleriyle (1 Hz, 5 Hz, 10 Hz) davranışı gözlemle.

Değerlendirme:

- Frekans arttıkça LED davranışı nasıl değişti?
- Clock olmadan sistem nasıl davranıyor?

DERS 3 — Sayaçlar (Counter)

Kazanım:

Binary sayma, clock darbeleriyle artış mantığını kavrar.

Konu Özeti:

- Counter'lar clock sinyalini kullanarak adım adım sayar.
- 4-bit counter 0000 → 1111 arası sayar.
- Her bit bir flip-flop tarafından tutulur.

Uygulama:

1. 4 adet D flip-flop kullanarak binary counter kur.
2. Clock sinyalini ilk flip-flopa bağla.
3. LED'lerle 4-bit sayımı gözlemle (0000 → 1111).

Değerlendirme:

- Her clock darbesinde ne değişiyor?
 - En yüksek bit (MSB) hangi adımda değişiyor?
-

DERS 4 — Encoder & Decoder

Kazanım:

Veri sıkıştırma (encoder) ve çözme (decoder) mantığını kavrar.

Konu Özeti:

- **Encoder:** Çok giriş → az bit (örnek: 8 tuş → 3-bit kod).
- **Decoder:** Az bit → çok çıkış (örnek: 3-bit sayı → 8 LED).
- 7-segment ekranlarda decoder sık kullanılır.

Uygulama:

1. 3-bit decoder (binary to 7-segment) kur.
2. Counter çıkışını decoder'a bağla.
3. LED'lerle 0–7 arası sayıları göster.

Değerlendirme:

- Decoder ne yapıyor?
- Encoder–decoder farkı nedir?

DERS 5 — Multiplexer (MUX) ve Demultiplexer (DEMUX)

Kazanım:

Birden çok veriyi tek hatta yönlendirmeyi veya ayırmayı öğrenir.

Konu Özeti:

- **MUX:** Çok giriş → bir çıkış (seçici veri hattı).
- **DEMUX:** Bir giriş → çok çıkış (veri dağıtıcı).
- Bilgisayarlarda veri yolu (bus) kontrolü bu sistemle yapılır.

Uygulama:

1. 4 girişli MUX kur.
2. Seçim hatlarını değiştirerek hangi girişin LED'e aktarıldığını gözlemle.
3. Aynısını DEMUX için ters yönde kur.

Değerlendirme:

- Hangi seçim hattında hangi veri geçiyor?
 - Bu sistem nerelerde kullanılır?
-

DERS 6 — Register ve Hafıza Yapıları

Kazanım:

Verilerin ardışık şekilde tutulduğu register yapısını kavrar.

Konu Özeti:

- Register, birden fazla flip-flop'tan oluşur.
- Veriler clock darbeleriyle kayar (shift register).
- Bu yapı RAM ve işlemci içi veri taşıyıcısıdır.

Uygulama:

1. 8-bit shift register kur.
2. Clock ile veriyi sağa/sola kaydır.
3. LED'lerde bitlerin kayışını gözlemle.

Değerlendirme:

- Clock arttıkça bit akışı nasıl ilerliyor?
- Register ile counter farkı nedir?

DERS 7 — Mini Proje: Dijital Trafik Lambası

Kazanım:

Zamanlayıcılar, latch'ler ve sayaçları birleştirerek kompleks dijital sistem oluşturur.

Konu Özeti:

- Trafik lambası: belirli sırada renk değiştiren zamanlama devresidir.
- Sayaç + decoder + timer kombinasyonu kullanılır.
- Bu proje, mikrodenetleyici mantığının temel modelidir.

Uygulama:

1. 3 LED (kırmızı–sarı–yeşil) bağla.
2. Clock sinyalini bir counter'a bağla.
3. Decoder kullanarak LED'lerin sırayla yanmasını sağla.
4. Süreleri ayarlamak için RC zamanlama ekle.

Değerlendirme:

- Sıralama doğru mu ilerliyor?
- Clock frekansı değişirse sistem davranışı nasıl değişir?

Aşama 3 Sonu – Değerlendirme

Öğrenci artık:

- Zaman tabanlı sistemleri anlar,
- Bit manipülasyonu, hafıza, MUX–DEMUX gibi dijital yapı taşlarını kavrar,
- Basit mikrodenetleyici (Arduino, PIC, ESP32 vb.) programlamasına geçmeye hazırdır.

Bir sonraki aşamada (Aşama 4), artık dijital devrelerin “akıllı” hale gelmesini ele alacağız:

Aşama 4 — Mikrodenetleyiciler ve Gerçek Sistem Simülasyonu (Tinkercad Circuits)

Burada artık kod yazarak LED, sensör, servo gibi bileşenleri kontrol edeceğiz.