

ELM 234 - Ödev #1



- Cevaplarınızı uzatmadan, **kısa, öz, açık** ve düzenli bir şekilde belirterek, ödevinizin okunmasına ve doğru şekilde değerlendirilmesine yardımcı olacaktır.
- **Problemlerde gidiş yolunuzu** ve yönteminizi gösterin.
- **Soruları arkadaşlarınızla tartışın fakat kendiniz çözün ve cevaplarınızı başka öğrencilerle paylaşmayın.**
- Cevaplarınızı **tek bir PDF dosyası** olarak teslim edin. Başka formatlar **kabul edilmeyecektir.**
- HDL yazmanız istenilen sorularda oluşturduğunuz kodları **düzgün formatlı şekilde** (Consolas veya Courier gibi monospaced fontlar kullanarak 12 pt. şekilde tek satır) ödevinize ekleyin. Örnek format ödevin sonuna mevcuttur. Simülasyon sonuçlarınızın ekran görüntüsünü ekleyin.

Problem 1. Alıştırmalar [4 + 4 + 4 = 12 puan]

- A. Aşağıda 10'luk sistemde verilen sayıların **8-bitlik** ve **11-bitlik** 2's complement şekliyle gösteriniz.
- a. 49 b. -104 c. 127 d. -17
- B. Aşağıda 2'lik sistemde verilen işlemleri hem **işaretsiz (unsigned)** varsayarak, hem de **2's complement** varsayarak yapınız. Sonuçları ondalık sayı olarak ifade ediniz.
- a. 11011 - 11110 b. 01100 + 10111 c. 10110 - 00111 d. 01011 + 01111
- C. Aşağıda verilen boolean denklemini sadece NAND kapıları kullanarak yazın.

$$X = \overline{AB} + AC + B(\overline{A} + C)$$

Problem 2. Devre tasarımı [2 + 4 + 4 + 10 = 20 puan]

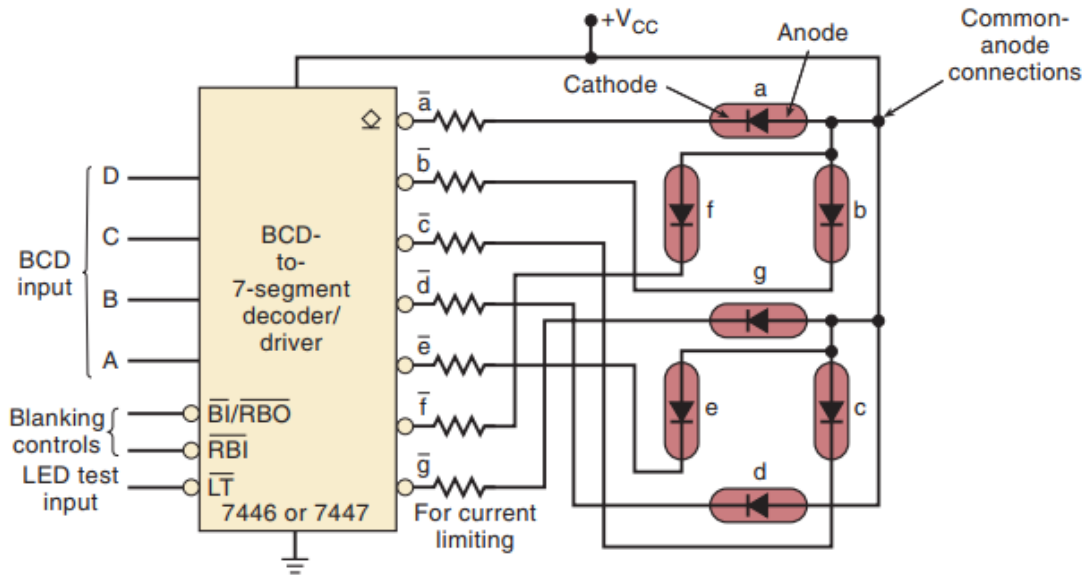
Yeni yapılan GTÜ konferans salonunun 3 adet girişi ve bu girişlerin yanında salondaki tavan lambalarını kontrol eden 3 adet anahtar bulunmaktadır. Her bir anahtarın lambaları kapatıp açabildiği bir lojik devre tasarlayın. (Her bir anahtar, lambalar açık ise kapatacak, kapalı ise açacak)

- A. Doğruluk tablosunu oluşturun. (Truth table)
- B. Boolean teoremleri kullanarak sadeleştirin.
- C. 74xx serisi IC'ler kullanarak devre şemasını çizin. Bütün bağlantıları (V_{dd} , gnd dahil) net bir şekilde gösterin. Ders kitabının Appendix A3'deki IC'leri referans alın.
- D. Devrenizi HDL kullanarak tasarlayın, bir testbench devresi oluşturun ve Modelsim de simüle edin. (Bunun için paylaşılan Modelsim youtube videosunu izlemeniz gerekir.)

Problem 3. Decoder tasarımı [4 + 4 + 4 + 18 = 30 puan]

Elimizdeki bir 7-Segment Şekil 1 de verildiği gibi BCD-to-7SD decoder devresine bağlanmıştır. Sizden 7-Segment de sadece 3-8 arası sayıları arasını gösterecek şekilde BCD girişlerini {3,4,5,6,7,8} rakamları ile eşleştiren bir decoder devresi tasarlamanız istenmektedir. Örnek olarak:

- BCD girişlerinden 4 sayısı verildiği zaman (DCBA → 0100), 7-Segment 4 sayısını gösterecek (bcfg segmentleri).
- BCD girişlerinden 8 sayısı verildiği zaman (DCBA → 1000), 7-Segment 8 sayısını gösterecek (abcdefg segmentleri).
- BCD girişlerinden 0-2 arası sayıları veya 9 ve üzeri sayılar verildiği zaman, 7-Segmentin ne gösterdiğinin önemi yok (don't care)
- Decoder çıkışlarının **active-low** olduğunu dikkate almayı unutmayın.



Şekil 1

- Her 6 farklı durum için (3,4,5,6,7,8 çıkışları) doğruluk tablosu oluşturun.
- İstediğiniz herhangi iki durumu **boolean teoremleri kullanarak** en sade haline getirin.
- Geri kalan durumları **K-map kullanarak** sadeleştirin (don't care lardan yararlanmayı unutmayın)
- Devrenizi HDL kullanarak tasarlayın, bir testbench devresi oluşturun ve Modelsim de simüle edin.

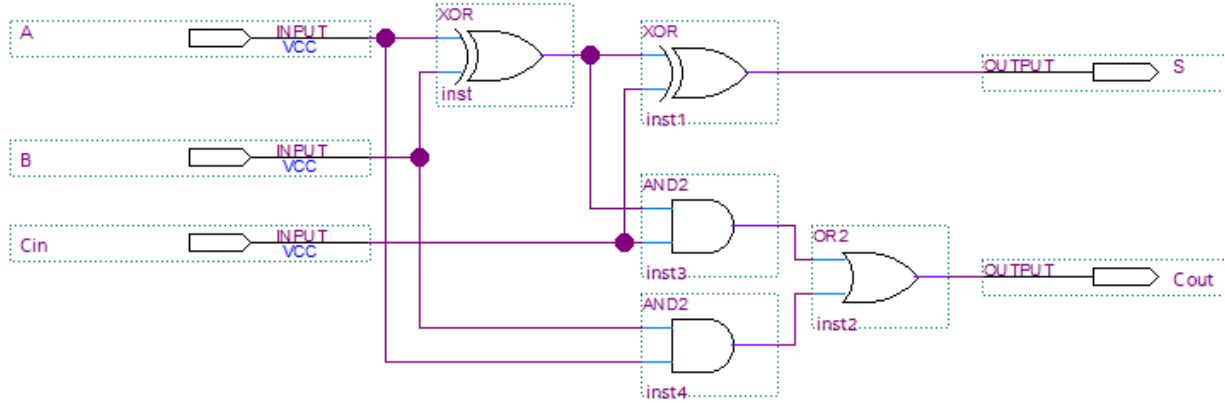
Problem 4. Kodlar [4 + 6 = 10 puan]

Dijital Sistemlerde hata tespiti için kullanılan Parity biti hakkında bilgi veriniz.

- İki türünü açıklayınız
- ASCII ile kodlanan **COVID-19** kelimesini msb'ye parity biti eklenmiş şekilde ve 8-bitlik gruplar halinde iki tür parity için de ayrı ayrı gösteriniz.

Problem 5. Gecikme [8 puan]

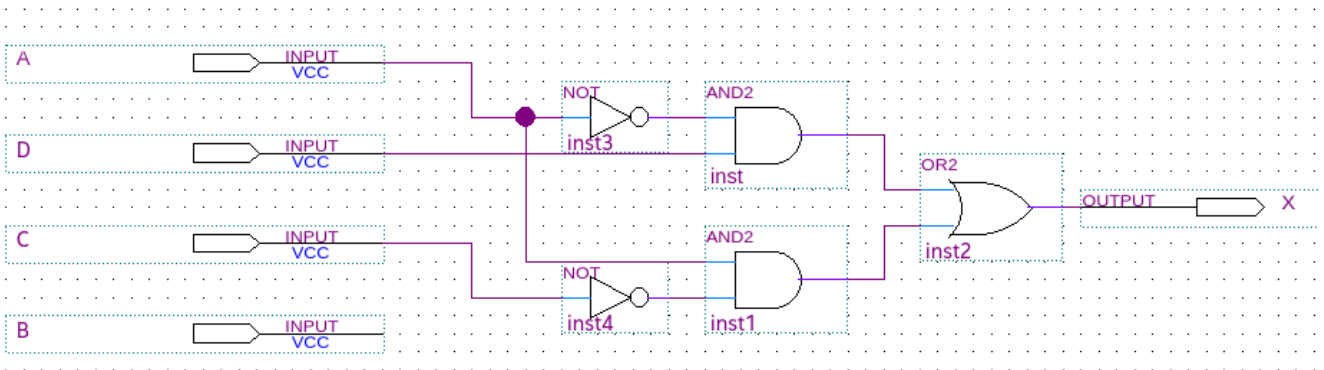
Şekil 2 de verilen devrenin **critical ve short path** lerini Tablo 1 deki gecikme değerlerini kullanarak hesaplayın.



Şekil 2

Problem 6. Glitch [8 + 12 = 20 puan]

- Şekil 3 de en sade şekli verilen 4 girişli devrenin hangi koşulda glitch oluşabileceği veya hiçbir şekilde oluşmayacağı hakkında yorum yapın. Varsa bu glitchi önleyebilecek bir devre ekleyin. Tablo 1 deki değerleri baz alın.
- Devrenizi HDL kullanarak tasarlayın, bir testbench devresi oluşturun ve Modelsim de verilen gecikme değerlerini ekleyerek simüle edin.



Şekil 3

Kapı	t_{pd} (ps)	t_{cd} (ps)
NOT	25	20
2-giriş AND	55	30
2-giriş OR	65	25
2-giriş XOR	80	65

Tablo 1

Örnek kod formatı:

Tablo içerisinde monospace font kullanarak (12pt, Consolas)

```
/* Dosya adı burada yorum olarak bulunsun
 * basit_devre.sv
 */

module basit_devre (
    input  logic a, b, c,
    output logic y
);

assign y = a & b & c;

endmodule
```