

MİKROİŞLEMCİLER

Dr. Meltem KURT PEHLİVANOĞLU

W-2

MİKROİŞLEMCİLER

Digital Logic +

Digital Design +

Computer Architecture +

Microprocessors +

Microcontrollers +

Assembly Language Programming

Digital Logic (Sayısal Mantık)

- Sayısal Sistem (digital system) iki gerilim seviyesine göre çalışır. Her Sayısal Sistemin bu iki gerilim seviyesine karşılık gelen bir biçimi olmalıdır. Bu nedenle Sayısal Devreler Binary (İkilik) Sayı sisteminde kullanılan **1 ve 0** ile tanımlanmak zorundadır.
- Bu Sayısal Sistemin girdilerinin ikilik koda dönüşmesini sağlar

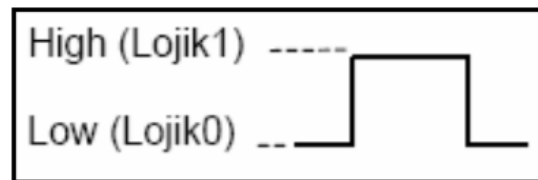
Digital Logic (Sayısal Mantık)

Pozitif Mantık

Pozitif Mantık

Yüksek	Alçak
1	0
Doğru	Yanlış
+5V	0V
Kapalı	Açık

- Yukarıdaki Pozitif Mantık ifadelerini kullanarak sayısal kavramları tanımlanabilir. Örneğin bir anahtarın kapalı olması sayısal sistemde '1' veya '5V'a eşit olacaktır.
- Bir kare dalga'nın yükselme ve düşmesinin çok küçük zaman diliminde olduğu düşünülürse kare dalga sayısal sinyallere güzel bir örnek olabilir. Aşağıda bir kare dalga üzerindeki Lojik seviyeler gösterilmiştir.



Pozitif mantık sayısal sinyal

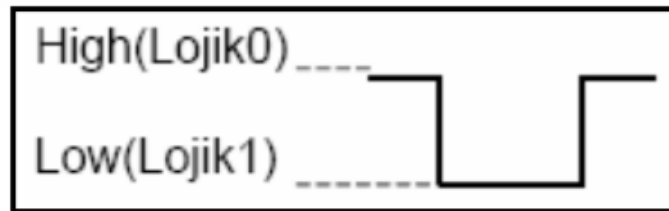
Digital Logic (Sayısal Mantık)

Negatif Mantık

Negatif Mantık

Yüksek	Alçak
0	1
Doğru	Yanlış
0V	+5V
Açık	Kapalı

- Sayısal devrelerde negatif mantık kullanımı bazı uygulamalarda tasarımcıya büyük kolaylıklar sağlamaktadır. Örneğin elektriksel gürültü problemi yaşanan sistemlerin tasarımında Negatif mantık kullanımı gürültü probleminin ortadan kalkmasını sağlayabilir.

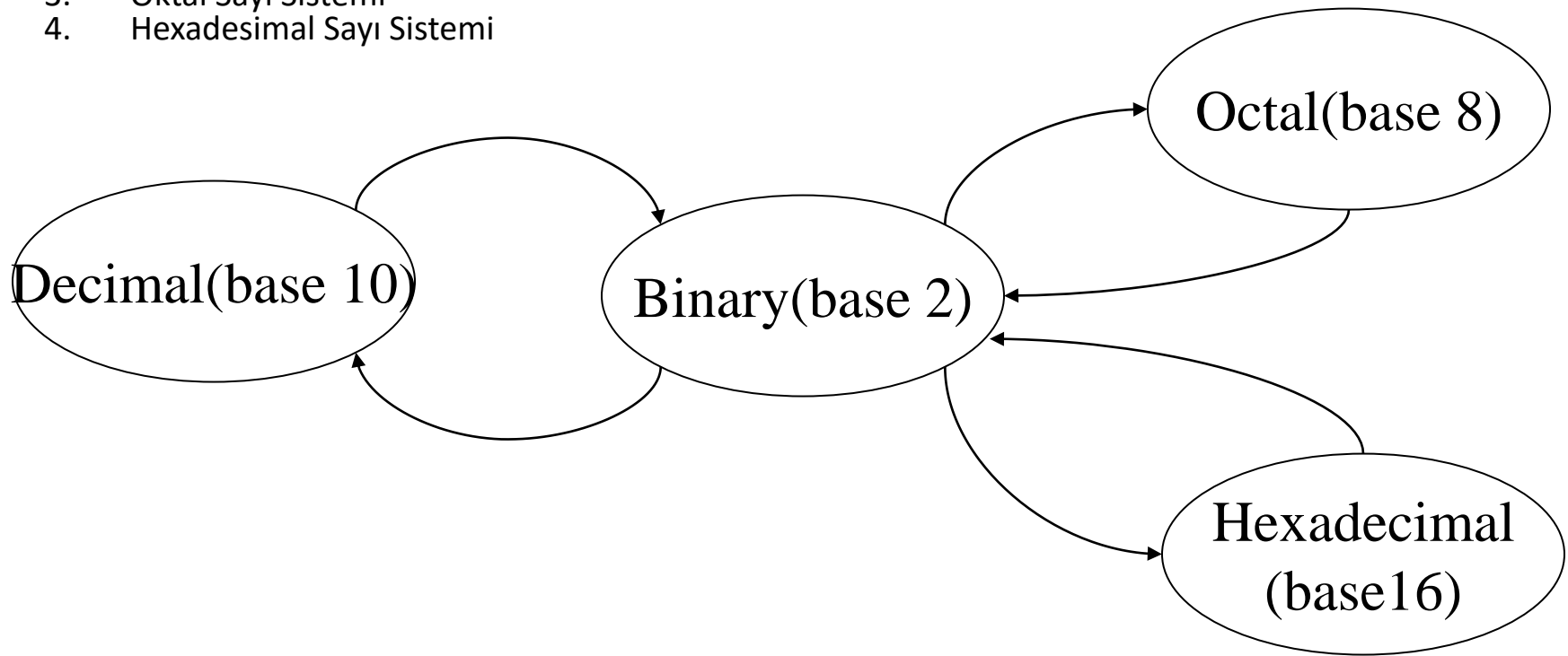


Negatif mantık sayısal sinyal

Sayı Sistemleri

- Dijital elektronikte dört çeşit sayı sistemi kullanılmaktadır.

1. Desimal Sayı Sistemi
2. Binary Sayı Sistemi
3. Oktal Sayı Sistemi
4. Hexadesimal Sayı Sistemi



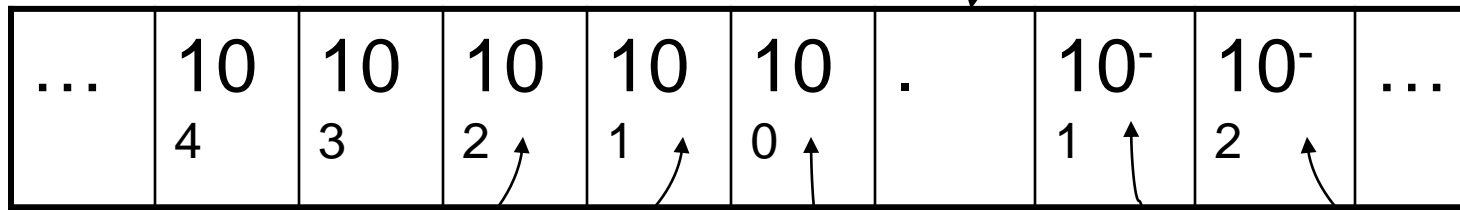
Sayı Sistemleri

Decimal

Taban : 10

Semboller : 0,1,2,3,4,5,6,7,8,9

decimal point



hundreds position

tens position

ones position

tenths position

hundredth position

Basamak Değeri Basamak Ağırlığı

↓ ↙

$$234.56_{10} = 2 \times 10^{+2} + 3 \times 10^{+1} + 4 \times 10^{+0} + 5 \times 10^{-1} + 6 \times 10^{-2}$$

↑

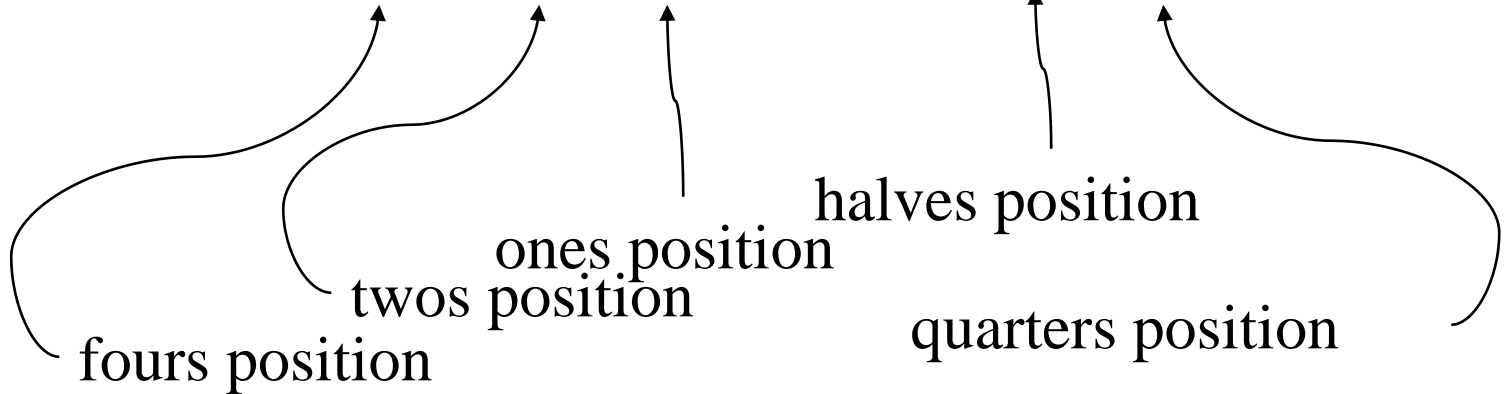
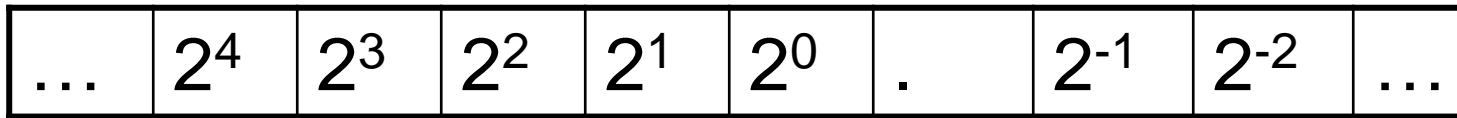
Taban Değeri

Sayı Sistemleri

Binary

Taban : 2
Semboller : 0,1

binary point



1 0 1 1 . 0 1
↑ ↑ ↑
En büyük Değerli Bit (Most Significant Bit, MSB) İkili Nokta (Binary Point, BP) En Küçük Değerli Bit (Least Significant Bit, LSB)

Basamak Değeri Basamak Ağırlığı
↓ ↙
 $1101.01_2 = 1 \times 2^{+3} + 1 \times 2^{+2} + 0 \times 2^{+1} + 1 \times 2^{+0} + 0 \times 2^{-1} + 1 \times 2^{-2}$
↑
Taban Değeri

Sayı Sistemleri

Örnek (Decimal Sayılar-Binary Sayılar)

Onlu sayılara örnek

$$\begin{aligned} 346.17_{10} &= (3 \times 10^2) + (4 \times 10^1) + (6 \times 10^0) + (1 \times 10^{-1}) + (7 \times 10^{-2}) \\ &= 300 \quad + \quad 40 \quad + \quad 6 \quad + \quad 0.1 \quad + 0.07 \end{aligned}$$

İkili sayılara örnek

$$\begin{aligned} 1101.01_2 &= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} \\ &= 8 \quad + \quad 4 \quad + 0 \quad + 1 \quad + 0 \quad + .25 \\ &= 13.25_{10} \end{aligned}$$

Sayı Sistemleri

Binary to Decimal

$$(110101)_2 = (?)_{10}$$

$$\begin{aligned} 110101_2 &= 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ &= 32 + 16 + 0 + 4 + 0 + 1 \\ &= 53_{10} \end{aligned}$$

Sayı Sistemleri

Decimal to Binary

$$(50)_{10} = (?)_2$$

$$50_{10} = 32 + 18$$

$$= 32 + 16 + 2$$

$$= 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^1$$

$$50_{10} = 110010_2$$

Bölüm	Kalan	
$50/2 = 25$	0	LSB
$25/2 = 12$	1	
$12/2 = 6$	0	
$6/2 = 3$	0	
$3/2 = 1$	1	
$1/2 = 0$	1	MSB

$$50_{10} = 110010_2$$

MSB.....LSB

Sayı Sistemleri

Decimal to Binary

$$(50.6875)_{10} = (?)_2$$

Tam kısım=50

Ondalık kısım=0.6875

Bölüm	Kalan
$50/2 = 25$	0
$25/2 = 12$	1
$12/2 = 6$	0
$6/2 = 3$	0
$3/2 = 1$	1
$1/2 = 0$	1

$0.6875 * 2 = 1.3750$	1
$1.3750 * 2 = 0.7500 (2.7500 - 2)$	0
$0.7500 * 2 = 1.5000$	1
$1.5000 * 2 = 1.0000$	1

$$50.6875_{10} = 110010.1011_2$$

$$50_{10} = 110010_2$$

Sayı Sistemleri

Octal

- Taban : 8
- Semboller : 0,1,2,3,4,5,6,7

7 0 3 . 1 5

↑ ↑ ↑

En büyük Değerli Basamak (Most Significant Digit, MSD) Sekizli Nokta (Octal Point) En Küçük Değerli Basamak (Least Significant Digit, LSD)

Basamak Değeri Basamak Ağırlığı

↓ ↙

$$703.15_8 = 7 \times 8^{+2} + 0 \times 8^{+1} + 3 \times 8^{+0} + 1 \times 8^{-1} + 5 \times 8^{-2}$$

↑

Taban Değeri

Sayı Sistemleri

Hexadecimal (Hex)

- Taban : 16
- Semboller : 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

1 A 3 . 1 F

↑ ↑ ↑

En büyük Değerli On altılı Nokta En Küçük Değerli
Basamak (Hexadecimal Basamak
(Most Significant Point) (Least Significant
Digit, MSD) Digit, LSD)

Basamak Basamak
Değeri Ağırlığı

↓ ↙

$$1A3.1F_{16} = 1 \times 16^{+2} + 10 \times 16^{+1} + 3 \times 16^{+0} + 1 \times 16^{-1} + 15 \times 16^{-2}$$

↑

Taban
Değeri

Sayı Sistemleri

Hex to Octal

$$\text{B2F}_{16} = (?)_8$$

$$\begin{aligned}\text{B2F}_{16} &= 1011 \ 0010 \ 1111 \ \{\text{convert to binary}\} \\ &= 101\textcolor{red}{|}1 \ 00\textcolor{red}{|}10 \ 1\textcolor{red}{|}111 \ \{\text{group into three-bit groupings}\} \\ &= 5457_8 \ \{\text{Convert to octal}\}\end{aligned}$$

Sayı Sistemleri

Decimal	Binary	Octal	Hexadecimal
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

**0-15 Decimal Sayıların
Binary Octal Hex Karşılıkları**

Sayı Sistemleri

Negatif Sayılar

- **2'ye göre tümleyen alma** bilgisayar mimarilerinin negatif sayıları kodlama biçimidir.
- 1. yöntem
 - 1'e göre tümleyen al
 - 1 ekle
- 2. yöntem
 - LSB den başlayarak '1' bit değerini görene kadar '0' değerlerini yaz
 - İlk gördüğün '1' bit değerini de yaz
 - Kalan bitlerin 1'e göre tümleyenlerini yaz

Sayı Sistemleri

Negatif Sayılar

- **1. Yöntem**

(-50)?

$$(50)_{10} = (110010)_2$$

- 001101 (1'e göre tümleyen)
- 001101 + 1 = 001110 (1 ekleme)

Sayı Sistemleri

Negatif Sayılar

- **2. Yöntem**
(-50)?

$$(50)_{10} = (110010)_2$$

$$- \quad 0011 \big|_{10}$$

Sayı Sistemleri

Taşma

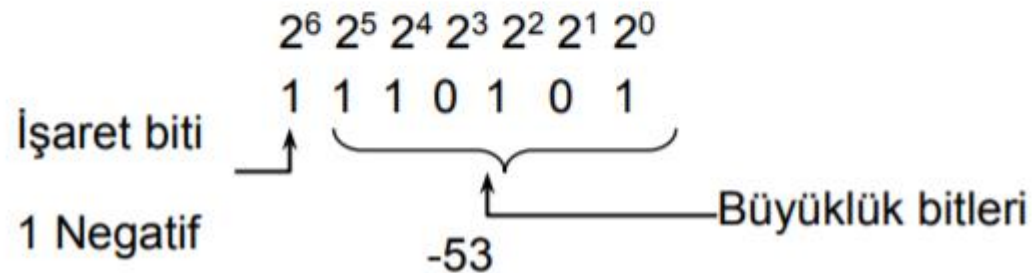
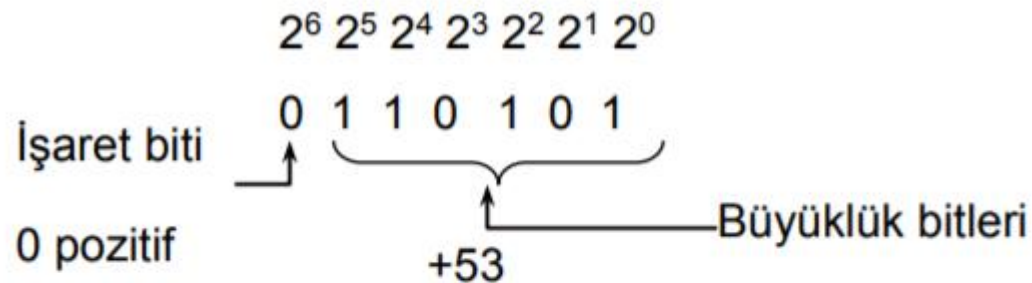
- Taşma farklı işaretli sayılarda meydana gelmez, aynı işaretli sayılarda meydana gelir.

$$\begin{array}{r} 70 = (0 \quad 100 \ 0110)_2 \\ + \quad 80 = (0 \quad 101 \ 0000)_2 \\ \hline 150 \ 1 \quad 001 \ 0110 \end{array}$$

- Ancak **8 bit registerda +150 değeri saklanabilir mi?**
 - » (Negatif ve pozitif sayıları saklayabileceğimizi düşünün!!)

Sayı Sistemleri

Taşma



Bit, Byte, Words, Double Word, Quad Word

- Bit: '1' '0'
- Byte: 8 bit
- Words: 16 bit
- Double word: 32 bit
- Quad word: 64 bit

Lojik Kapılar

- Bir lojik kapı bir çıkış, bir veya birden fazla giriş hattına sahiptir
- Çıkışı, giriş hatlarının durumuna bağlı olarak Lojik-1 veya Lojik-0 olabilir
- Bir Lojik kapının girişlerine uygulanan sinyale bağlı olarak çıkışının ne olacağını gösteren tabloya doğruluk tablosu (truth table) adı verilir.

Lojik Kapılar

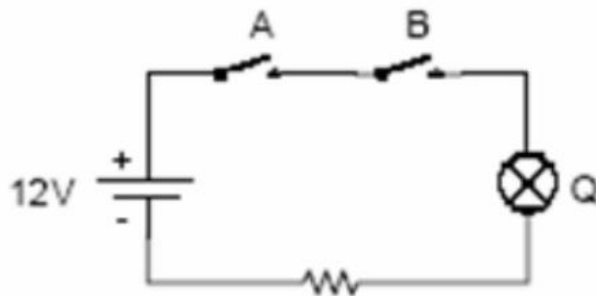
AND(VE) Kapısı

Yüksek	Alçak
1	0
Doğru	Yanlış
+5V	0V
Kapalı	Açık

- AND kapısının bir çıkış, iki veya daha fazla giriş hattı vardır.



a) Sembolü



c) Elektrik Eşdeğer Devresi

Girişler		Çıkış
A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

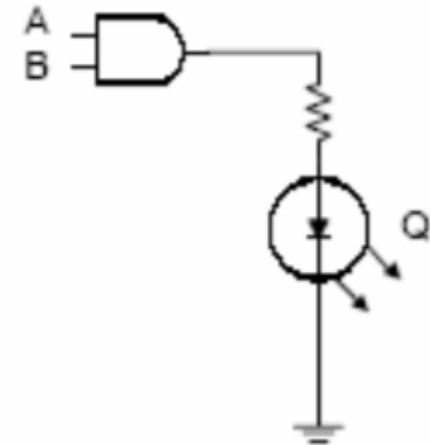
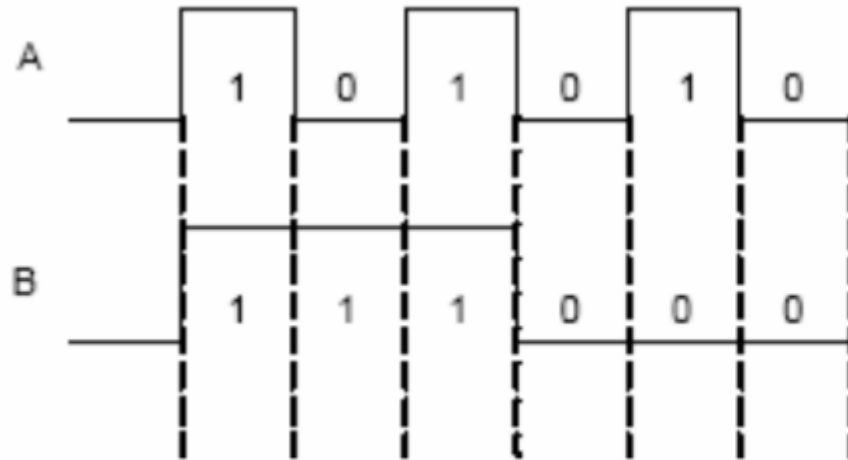
b) Doğruluk Tablosu

Lojik Kapılar

AND(VE) Kapısı

Yüksek	Alçak
1	0
Doğru	Yanlış
+5V	0V
Kapalı	Açık

- A ve B işaretleri AND kapısı girişlerine uygulanırsa çıkış dalga şekli (Q) nasıl olur?
- LED hangi zaman aralıklarında yanar?



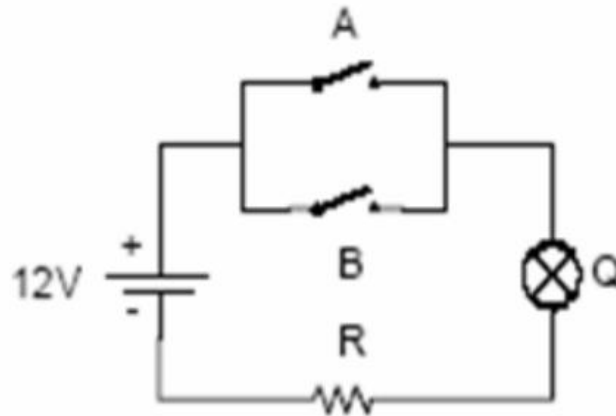
Lojik Kapılar

OR (VEYA) Kapısı

- OR kapısının iki veya daha fazla giriş, bir çıkış hattı vardır.



a) Sembolü



c) Elektrik Eşdeğer Devresi

Girişler		Çıkış
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

b) Doğruluk Tablosu

Yüksek	Alçak
1	0
Doğru	Yanlış
+5V	0V
Kapalı	Açık

Yüksek	Alçak
1	0
Doğru	Yanlış
+5V	0V
Kapalı	Açık

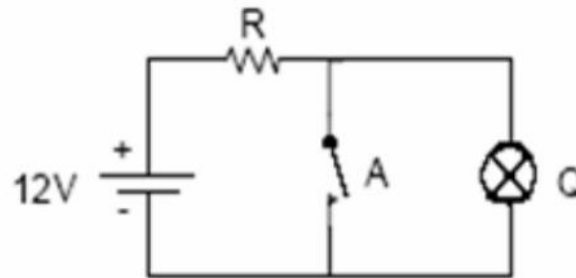
Lojik Kapılar

NOT (INVERTER-DEĞİL) Kapısı

- NOT kapısı bir giriş ve bir çıkış hattına sahiptir.
- Çıkış işareti giriş işaretinin tersi (değili-tümleyen) olur.



a) Sembolü



c) Elektrik Eşdeğer Devresi

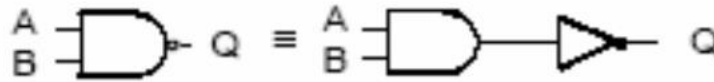
Giriş	Çıkış
A	Q
0	1
1	0

b) Doğruluk Tablosu

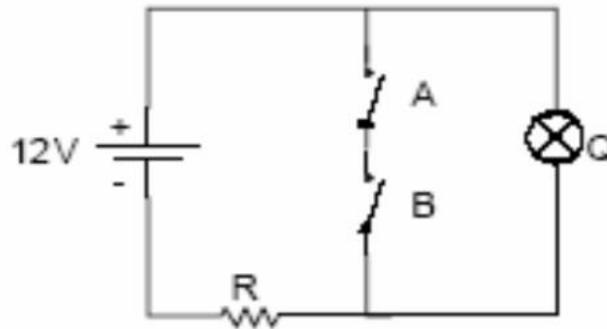
Lojik Kapılar

NAND (VE DEĞİL) Kapısı

- NAND kapısının en az iki giriş ve bir çıkışı vardır. Lojik AND fonksiyonunun DEĞİL'i olarak tanımlayabiliriz.



(a) Sembolü



(c) Elektrik eşdeğer devresi

Girişler		Çıkış
A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

(b) Doğruluk Tablosu

Yüksek	Alçak
1	0
Doğru	Yanlış
+5V	0V
Kapalı	Açık

Lojik Kapılar

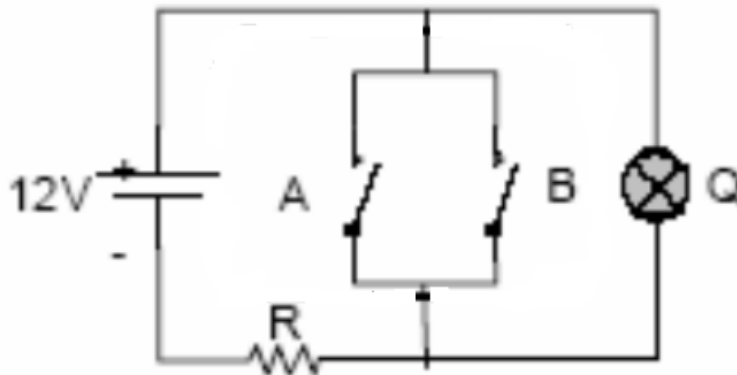
NOR (VEYA DEĞİL) Kapısı

- NOR kapısının en az iki giriş ve bir çıkış hattı vardır. Lojik fonksiyon olarak OR fonksiyonunun DEĞİL'i olarak tanımlayabiliriz.



Girişler		Çıkış
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

b) Doğruluk Tablosu



(c) Elektrik eşdeğer devresi

Lojik Kapılar

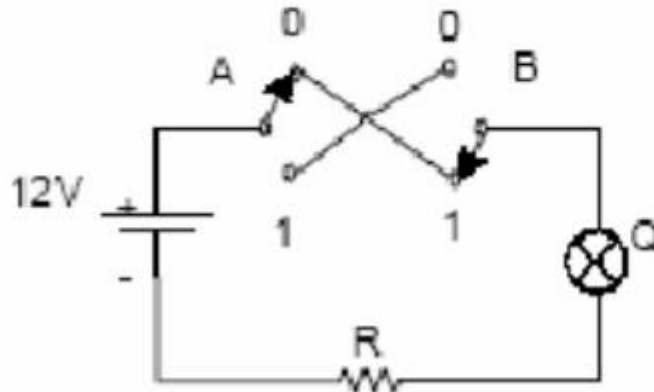
XOR(ÖZEL VEYA) Kapısı

- Bir XOR kapısının iki veya daha fazla giriş, bir çıkış hattı vardır.

Yüksek	Alçak
1	0
Doğru	Yanlış
+5V	0V
Kapalı	Açık



a) Sembolü



c) Elektrik Eşdeğer Devresi

Girişler		Çıkış
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

b) Doğruluk Tablosu

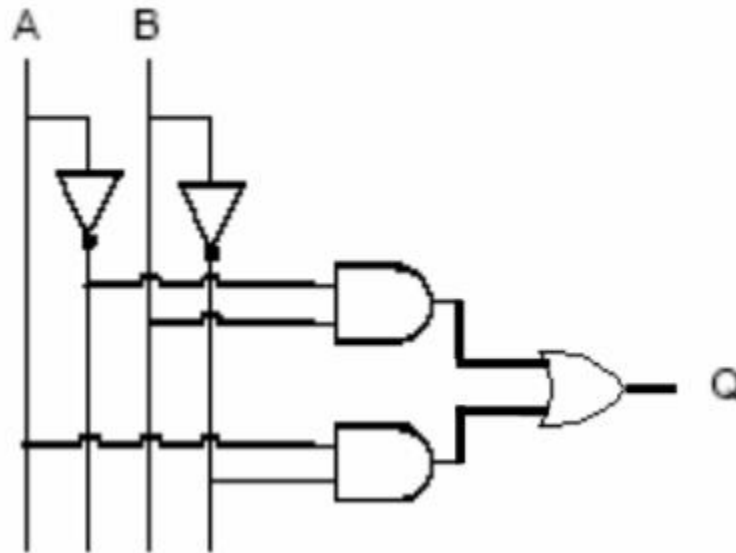
Yüksek	Alçak
1	0
Doğru	Yanlış
+5V	0V
Kapalı	Açık

Lojik Kapılar

XOR(ÖZEL VEYA) Kapısı

- XOR kapısı NOT-AND-OR kapıları ile ifade edilebilir. Bu durumda bir XOR fonksiyonunu;

$Q = \overline{A} \cdot B + A \cdot \overline{B}$ şeklinde tanımlayabiliriz.

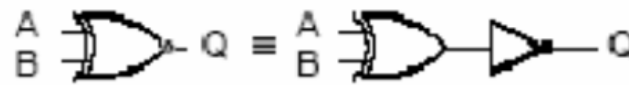


Yüksek	Alçak
1	0
Doğru	Yanlış
+5V	0V
Kapalı	Açık

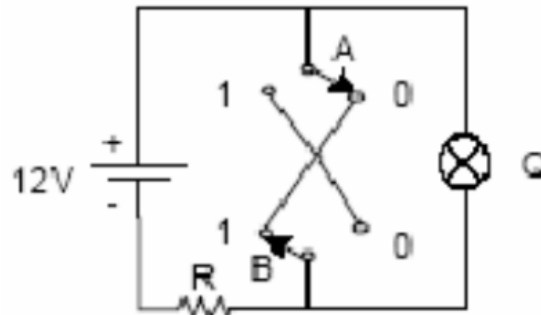
Lojik Kapılar

XNOR(ÖZEL VEYA DEĞİL) Kapısı

- Bir XNOR kapısının iki veya daha fazla giriş, bir çıkış hattı vardır. Lojik fonksiyon olarak XOR işleminin DEĞİLİ dir.



a) Sembolü



c) Elektrik Eşdeğer Devresi

Girişler		Çıkış
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

b) Doğruluk Tablosu

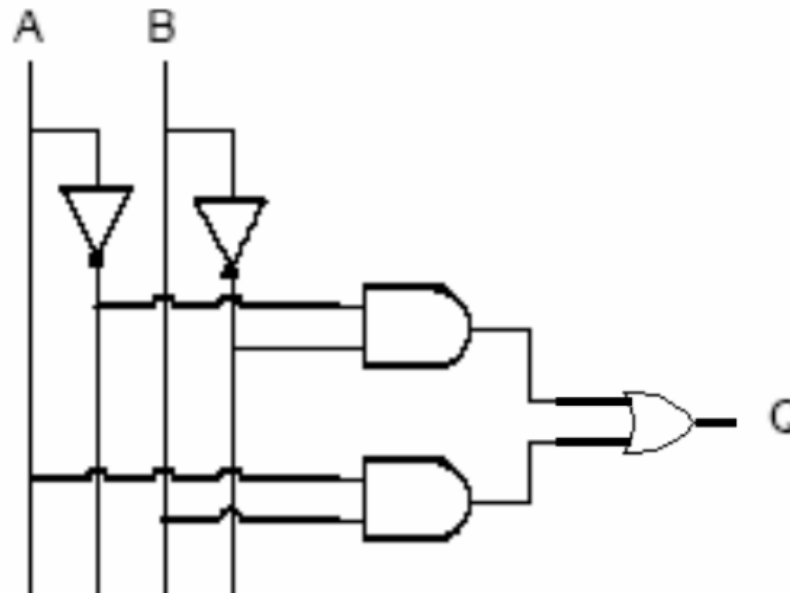
Yüksek	Alçak
1	0
Doğru	Yanlış
+5V	0V
Kapalı	Açık

Lojik Kapılar

XNOR(ÖZEL VEYA DEĞİL) Kapısı

- XNOR kapısı NOT-AND-OR kapıları ile ifade edilebilir. Bu durumda bir XNOR fonksiyonunu;

$$Q = \bar{A} \cdot \bar{B} + A \cdot B$$



Lojik Kapılar

To go from verbal definition to logic function

- AND → **if and only if all TRUE** (1)
- OR → **either or**
- NOT → **FALSE**
- NAND (not AND) → if and only if **all inputs are '1'**; the output is '0'
- NOR (not OR) → if and only if **all inputs are '0'**; the output is '1'
- XOR → if inputs **are not equal each other** ; the output is '1'
- XNOR → if **both inputs** are **the same** output is '1'

Lojik Kapılar

To go from verbal definition to logic function

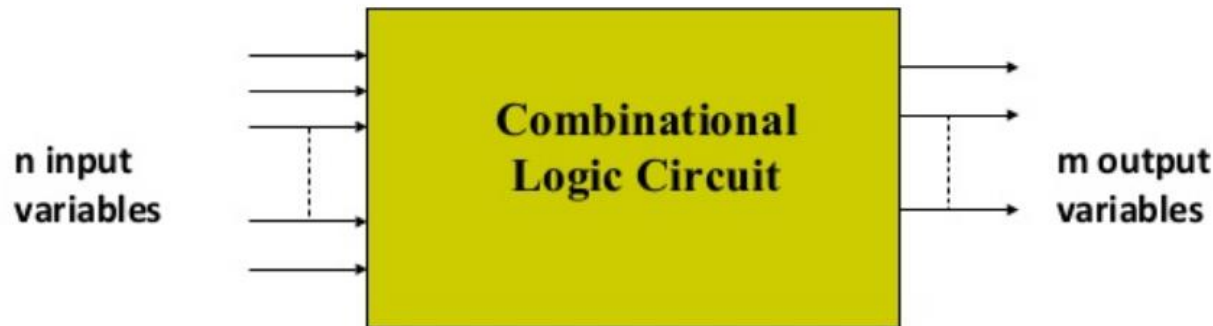
- G is true if and only if all of A,B,C are true or both D and E are false
- F is true when either A is true or both B and C are true or B is true and D is false
- H is false if and only if all A,B are false or C is true or D is false

IC (Integrated Circuit-Entegre Devre)

- Lojik kapılar sayısal sistemlerin temel elemanlarıdır.
- Bir çok lojik kapının oluşturduğu bir sayısal devre bir silisyum yonga üzerine entegre devre (integrated circuit - IC) olarak yapılır
- Tek bir yonga içersine yerleştirilen kapı sayısına göre entegre devreler entegresyon ölçeğini göstermesi açısından dört ayrı grupta incelenebilirler.
 1. SSI (Küçük Ölçekli Entegrasyon - Small Scale Integration) En fazla 20 lojik kapı içeren entegre devrelerdir.
 2. MSI (Orta Ölçekli Entegrasyon - Medium Scale Integration) 1000 bellek bitinden daha az ve 20 ila 100 kapı içeren entegre devrelerdir. Örneğin; sayıcılar, kaydırmalı kaydediciler, kod çözücüler v.b.
 3. LSI (Büyük Ölçekli Entegrasyon-Large Scale Integration) 1000'den 16000'e kadar bellek biti, 100 ila 5000 lojik kapı içeren entegre devreleridir. Örneğin 8- bitlik mikroişlemci, bellek yongaları v.b
 4. VLSI (Çok Büyük Ölçekli Entegrasyon-Very Large Scale Integration) 5000 lojik kapıdan daha fazla kapı içeren entegre devreleridir. Örneğin 16-bitlik mikroişlemci, yüksek yoğunluklu bellek yongaları v.b

Combinational Logic Design (Birleşik Devre Tasarımı)

- Bir birleşik devrede lojik kapılar vardır ve bunlar birbirine bağlanmıştır.
- n tane binary giriş değişkeni dış kaynaktan gelir, m tane binary çıkış değişkeni dış hedefe gider.
- **Bellek** yoktur!
- Bir birleşik devrede 2^n elemanlı giriş kümesine karşılık gelen binary çıkış değerleri kümesi bulunur. Devre; her bir çıkış değişkenine karşılık gelen m tane Boole fonksiyonu ile de tanımlanabilir
- Herhangi bir andaki **çıkışlar bütünüyle o andaki girişlere** bağlıdır



Birleşik Devre Blok Şeması

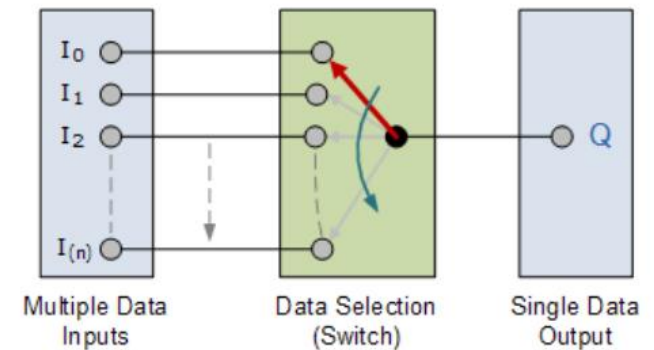
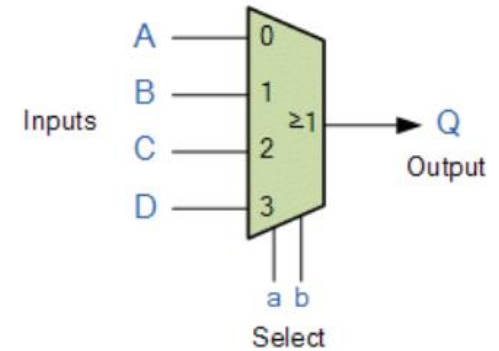
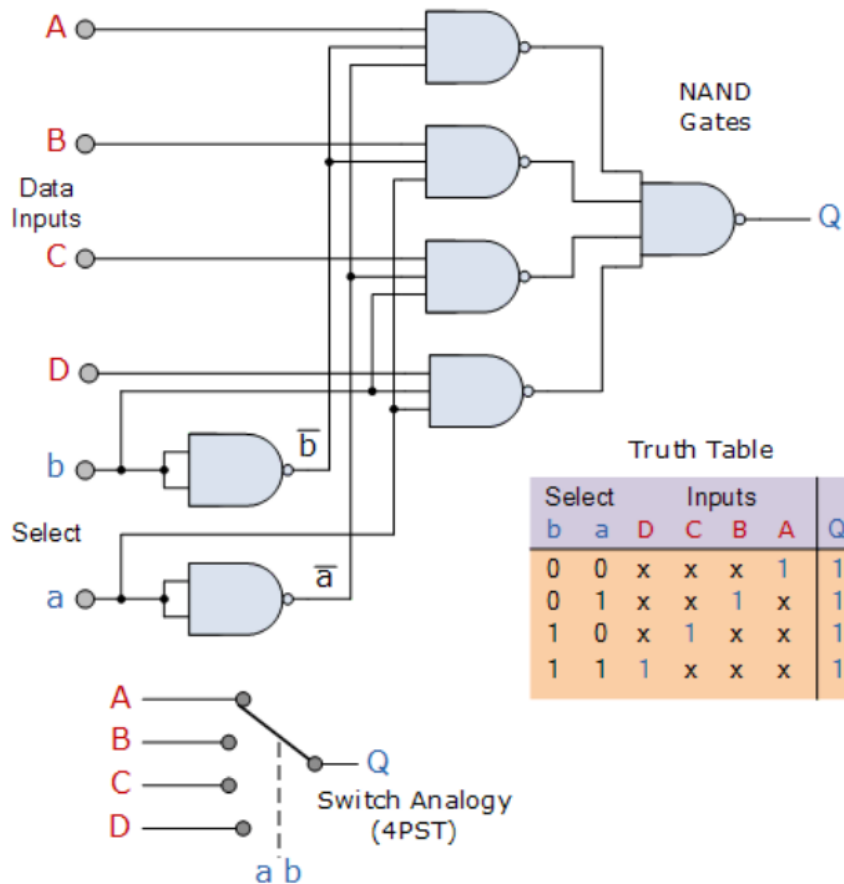
Combinational Logic Design (Birleşik Devre Tasarımı)

- Adders (Half Adder, Full Adder, Binary/Parallel Adder)
- Subtractors (Half Subtractor, Full Subtractor, Binary/Parallel Subtractor)
- Multiplexer (Mux)
- Encoder (2^n to n mapping circuit)
- Decoder (n to 2^n mapping circuit)

Multiplexer

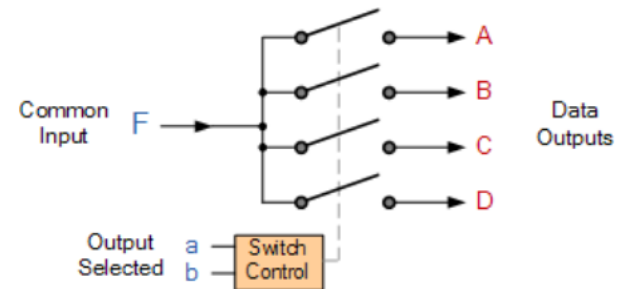
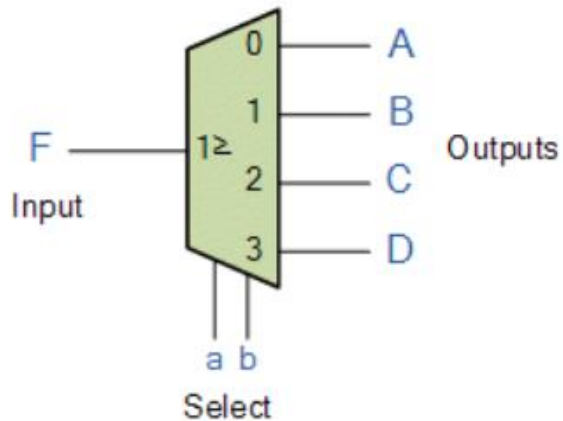
Çoklayıcı Veri Seçiciler

- 'selector under some conditions' (if else gate)
- $2^n \times 1$ 'lik MUX için n tane seçiciye ihtiyaç vardır



Demultiplexer

1-to-4 Channel De-multiplexer



Output Select		Data Output Selected
a	b	
0	0	A
0	1	B
1	0	C
1	1	D

$$F = \bar{a}\bar{b}A + \bar{a}bB + a\bar{b}C + abD$$

MUX

To go from verbal definition to logic function

F is equal to G when A and B are true

else

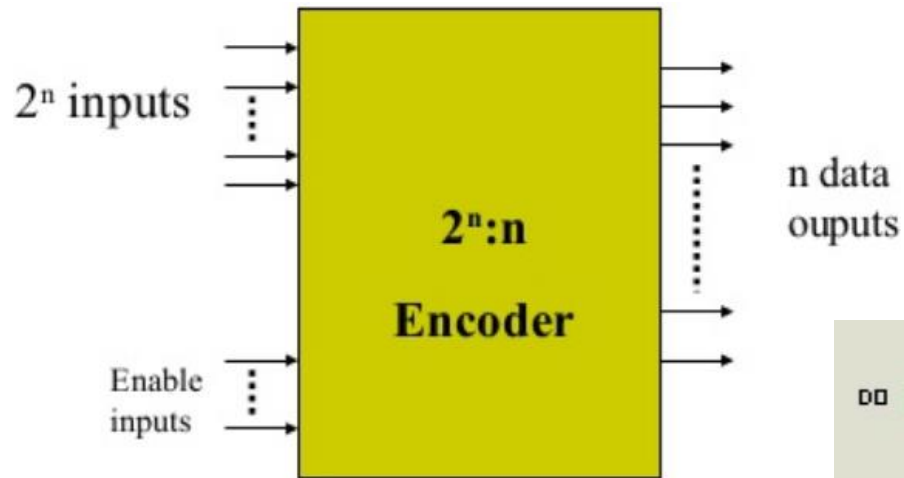
F is equal to H

where

G is true when C is equal to D and

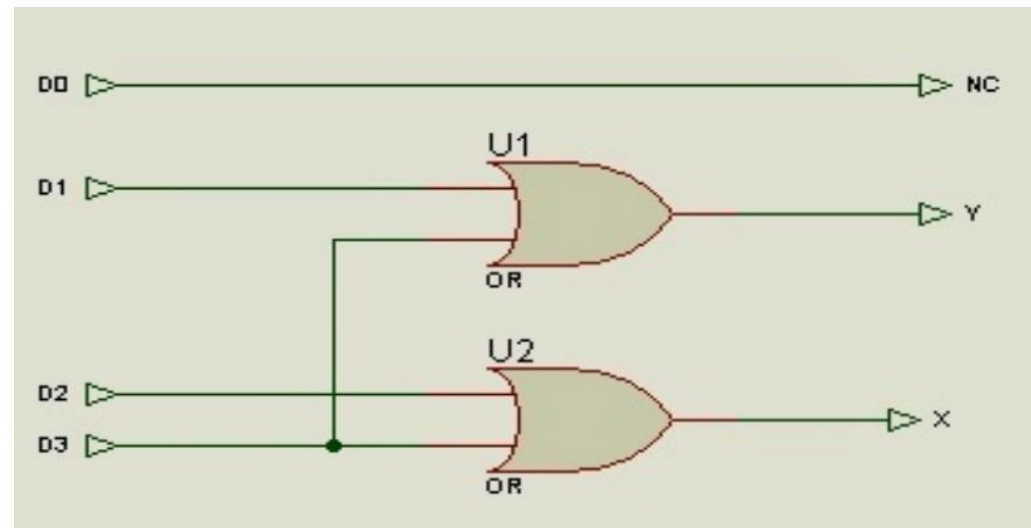
H is true when A and C are different

Encoder Kodlayıcı



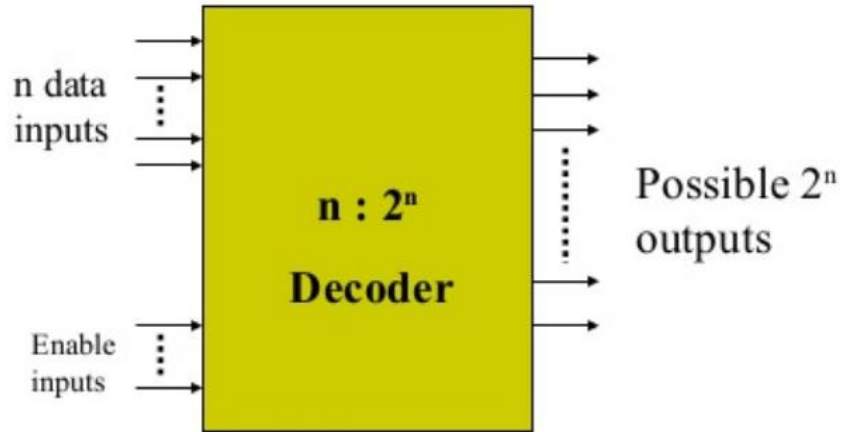
Encoder Blok Şeması

GİRİŞLER				ÇIKIŞLAR	
D3	D2	D1	D0	X	Y
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1



Decoder

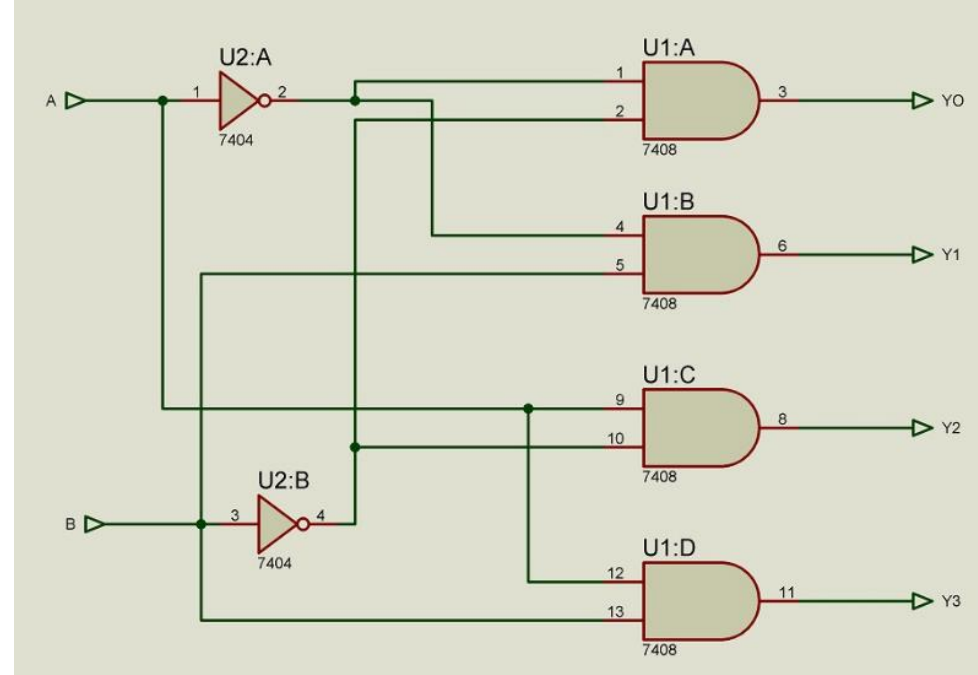
Kod Çözücü



Decoder Blok Şeması

En	A	B	Y ₃	Y ₂	Y ₁	Y ₀
0	X	X	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0		1	0	0
1	1	1	1	0	0	0

2 to 4 Decoder Doğruluk Tablosu

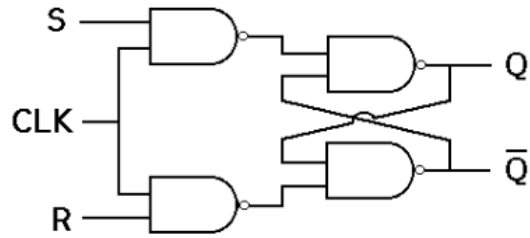


Flip-Floplar

Yaz-Bozlar

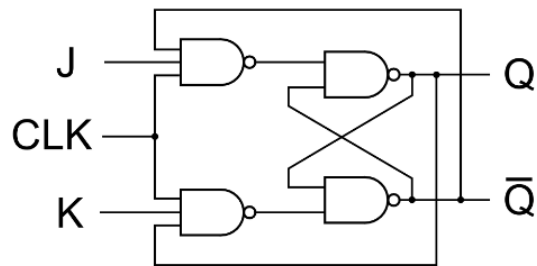
- **1 bitlik bilgi** saklayabilir iki çıkışı vardır. Biri öğenin içindeki değeri diğeri (Q) ise bunun tümleyenini (NOT) (Q') verir.
- Saat vuruşlu eş zamanlı devrelerdir.
- Bir flip-flop içindeki binary değeri saat vuruşu gelene kadar korur

Flip-Floplar



SR Flip-Flop

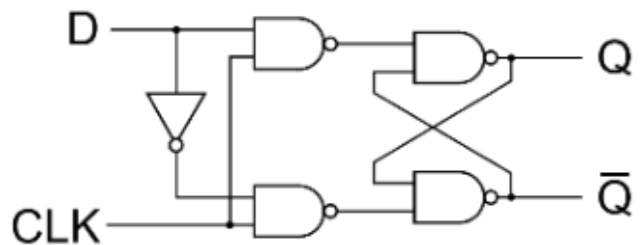
S	R	Q	Q'
0	0	0	1
0	1	0	1
1	0	1	0
1	1	∞	∞



JK Flip-Flop

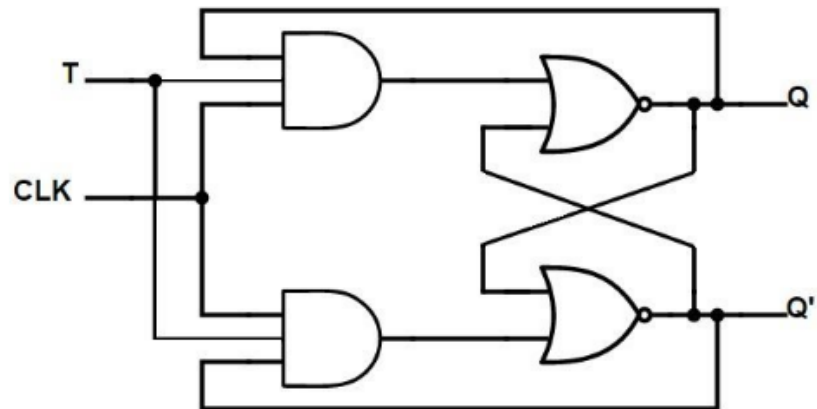
J	K	Q	Q'
0	0	0	0
0	1	0	0
1	0	0	1
1	1	0	1
0	0	1	1
0	1	1	0
1	0	1	1
1	1	1	0

Flip-Floplar



D Flip-Flop

Clock	D	Q	Q'
↓ » 0	0	0	1
↑ » 1	0	0	1
↓ » 0	1	0	1
↑ » 1	1	1	0

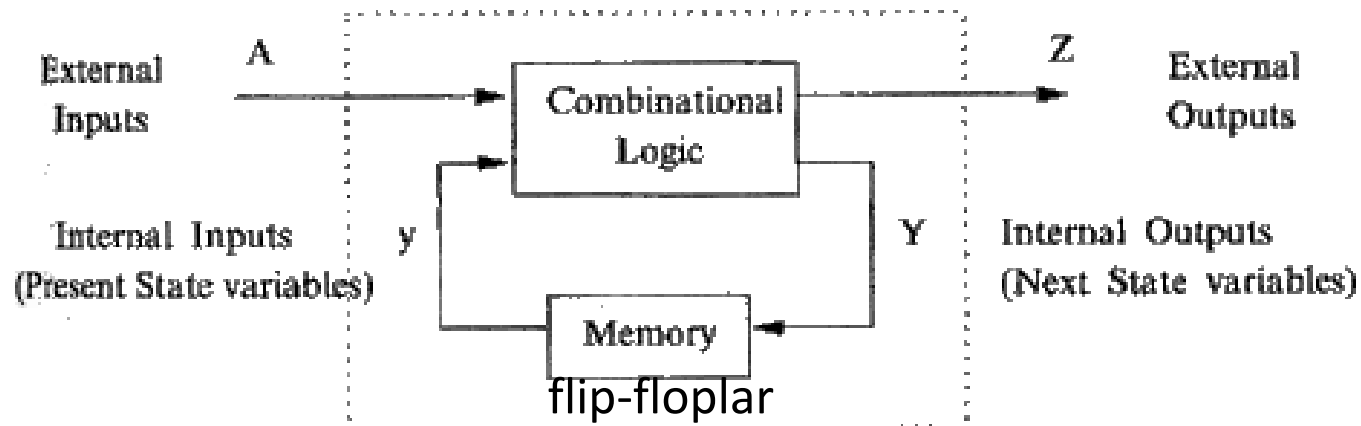


T	Q	Q (t+1)
0	0	0
1	0	1
0	1	1
1	1	0

SEQUENTIAL LOGIC

Sıralı Devreler

- Lojik kapılar + flip-floplar
- External Outputs $Z = \text{functions}(\text{external inputs, internal inputs})$, internal inputs=present state
- Next State $Y = \text{functions}(\text{inputs, present state})$, internal inputs=present state
- İstenilen tip sayıda flip-flop devreye eklenebilir
- Birleşik devredeki (combinational logic) kapılar dışarıya gönderilecek binary işaretin değeri ile flip-floplarda saklanacak binary değerleri belirler. Bu işlem her saat vuruşundan sonra gerçekleşir. Flip-flopların çıkışları combinational devrenin girişlerine uygulanarak devrenin davranışı belirlenir.



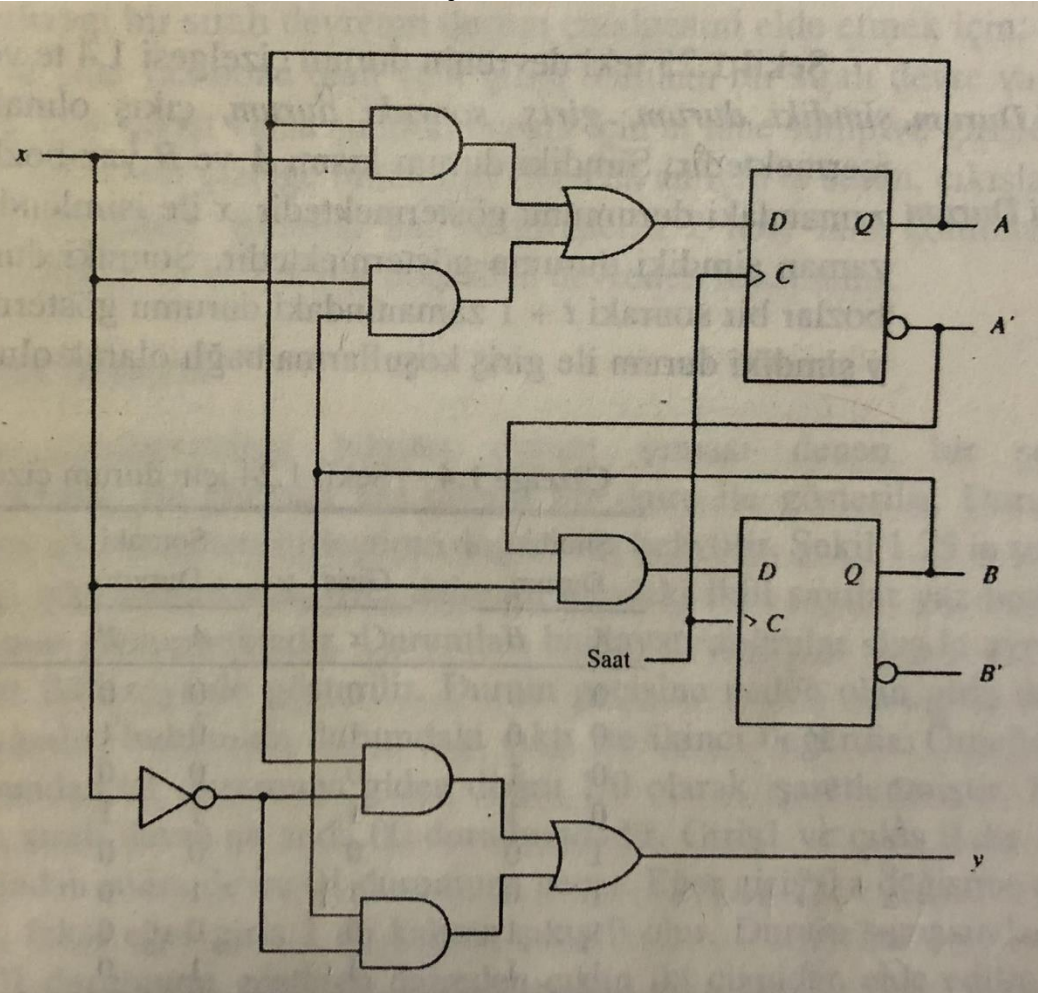
Sıralı Devre Blok Şeması

SEQUENTIAL LOGIC

Sıralı Devreler(Örnek)

- Sıralı devrenin durum çizelgesi?
- Sıralı devrenin durum şeması?

Clock	D	Q	Q'
↓ » 0	0	0	1
↑ » 1	0	0	1
↓ » 0	1	0	1
↑ » 1	1	1	0



SEQUENTIAL LOGIC

Sıralı Devreler(Örnek)

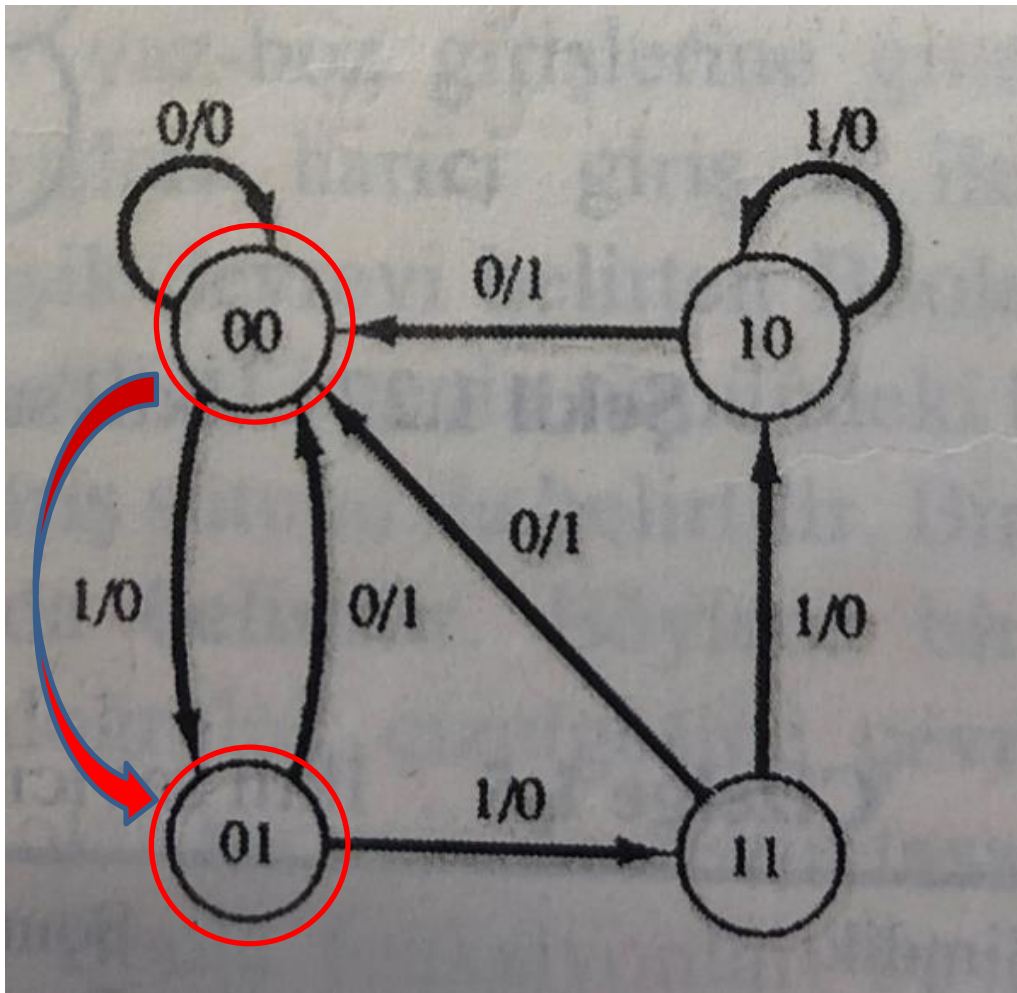
Sıralı devrenin durum çizelgesi

Şimdiki Durum		Giriş x	Sonraki Durum		Çıkış y
A	B		A	B	
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

SEQUENTIAL LOGIC

Sıralı Devreler(Örnek)

Sıralı devrenin durum şeması



(giriş/çıkış)

- Örn.

1/0 : giriş değeri $x=1$, çıkış değeri $y=0$

Bu durumda **present state 00** iken **next state 01** oluyor.

Kaynaklar

- http://kisi.deu.edu.tr//ozlem.karaca/sunumlar/lojik_devreler.pdf
- <https://www.elektrikport.com/teknik-kutuphane/encoder--decoder-nedir/16911#ad-image-0>
- https://www.electronics-tutorials.ws/combination/comb_3.html
- https://www.electronics-tutorials.ws/combination/comb_2.html
- <https://www.electronicsforu.com/resources/learn-electronics/flip-flop-rs-jk-t-d>
- <https://techdifferences.com/difference-between-compiler-and-interpreter.html>
- <https://medium.com/@abdulsamet.ileri/compilers-ve-interpreters-eacd14a227c4>