

# SAYISAL ELEKTRONİK

---

Ege Ü. Ege MYO  
Mekatronik Programı

# ***BÖLÜM 2***

---

***Sayı Sistemleri***

***İkilik, Onaltılık ve İKO Sayılar***

# İkilik Sayı Sistemi

3

- Çoğu dijital sistemler 8, 16, 32, ve 64 bit gibi, 2'nin çift kuvvetleri bit gruplarını ele almaktadır.
- 8-bit ikilik sayı – her bitin ağırlıklı değerleri

|       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|
| $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
| 128   | 64    | 32    | 16    | 8     | 4     | 2     | 1     |

# İkilik Sayı Sistemi

4

- Örnek: 1011 1010 ikilik sayısını onluk eşdeğerine dönüştürün.

|       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|
| $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
| 128   | 64    | 32    | 16    | 8     | 4     | 2     | 1     |
| 1     | 0     | 1     | 1     | 1     | 0     | 1     | 0     |

---

$$128 + 0 + 32 + 16 + 8 + 0 + 2 + 0$$
$$= 186_{10}$$

# İkili Sayı Sistemi

5

- Kesirli İkili Sayılar
- Örnek: 1011.1010 ikili sayısını onluk eşdeğerine dönüştürün.

|       |       |       |       |   |          |          |          |          |
|-------|-------|-------|-------|---|----------|----------|----------|----------|
| $2^3$ | $2^2$ | $2^1$ | $2^0$ | • | $2^{-1}$ | $2^{-2}$ | $2^{-3}$ | $2^{-4}$ |
| 8     | 4     | 2     | 1     | • | $1/2$    | $1/4$    | $1/8$    | $1/16$   |

|   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 1 | 1 | • | 1 | 0 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|

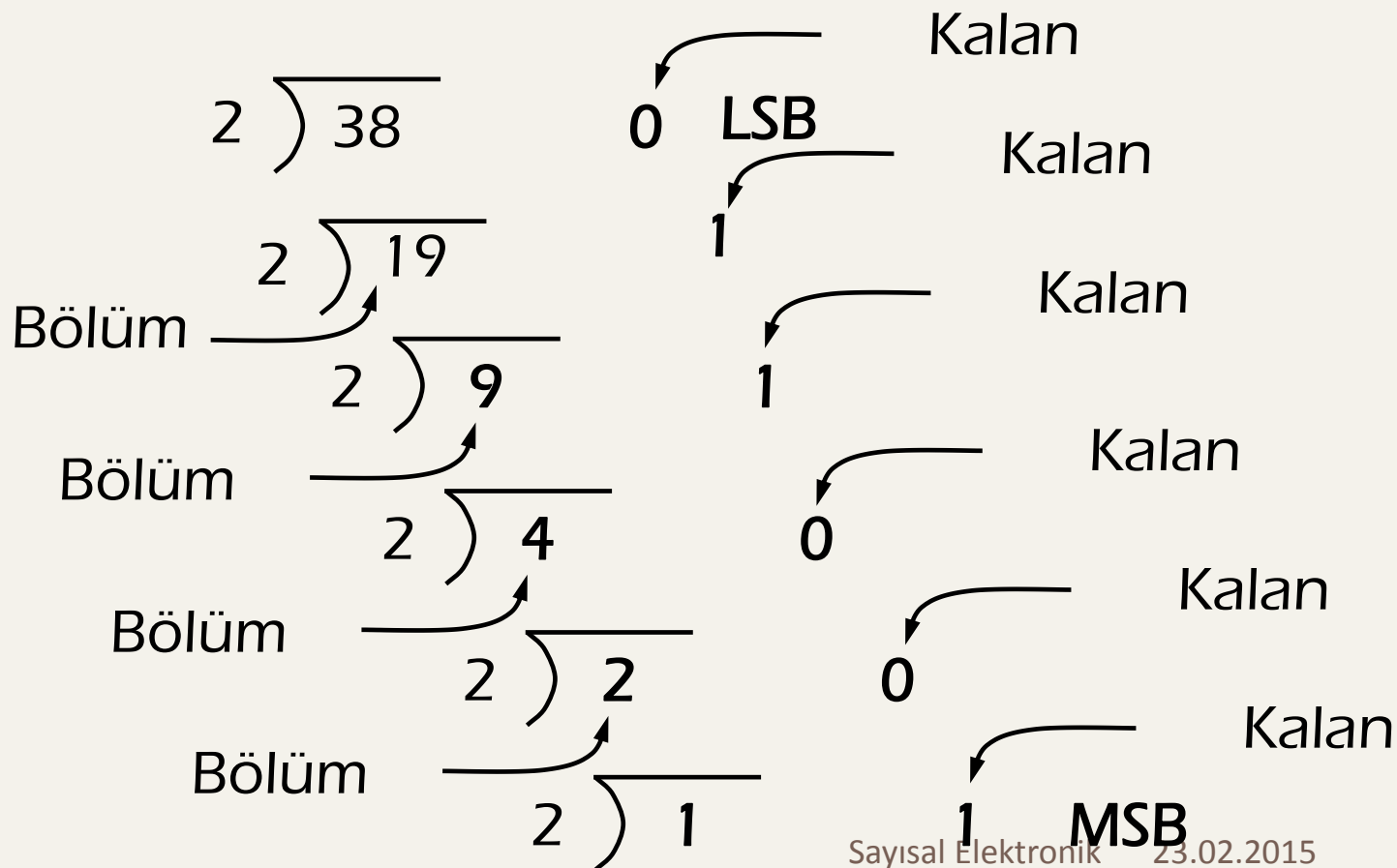
8 + 0 + 2 + 1 + 0.5 + 0 + 0.125 + 0

$$= 11.625_{10}$$

# Onluk-tan İkilik-e

6

$$\text{Örnek: } 38_{10} = \underline{100110}_2$$



# Onluk'tan İkilik'e

7

$$\text{Örnek } 38_{10} = \underline{100110}_2$$

|       |       |       |       |       |       |       |       |
|-------|-------|-------|-------|-------|-------|-------|-------|
| $2^7$ | $2^6$ | $2^5$ | $2^4$ | $2^3$ | $2^2$ | $2^1$ | $2^0$ |
| 128   | 64    | 32    | 16    | 8     | 4     | 2     | 1     |
| 0     | 0     | 1     | 0     | 0     | 1     | 1     | 0     |

---

$$0 + 0 + 32 + 0 + 0 + 4 + 2 + 0$$
$$= 38_{10}$$

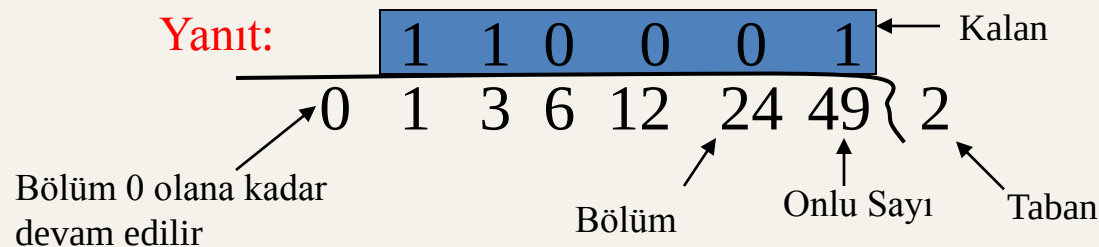
# İkili'ye Dönüşüm

8

Onlu sayıları bölme yöntemi ile istediğiniz sayı tabanına dönüştürebiliriz.

**Örnek:** 49 onlu sayıyı ikili sayıya dönüştürün.

**Çözüm:** Ters bölme yaparak dönüşümü pratik olarak yapabiliriz.



# Onaltılık Sayı Sistemi

- ❑ Çoğu dijital sistemler 8, 16, 32, ve 64 bit gibi, 2'nin çift kuvvetleri bit gruplarını ele almaktadır.
- ❑ Onaltılık sistemde 4 bitlik gruplar kullanır.
- ❑ 16 Tabanı
  - ▣ 16 olası simge
  - ▣ 0 ile 9 ve A ile F
- ❑ Uzun ikilik dizilerin kolay idaresi

# Onaltılık-tan Onluk-a Dönüşüm

10

- Her basamaktaki rakamı ağırlığı ile çarparak basamak değerini elde edin.

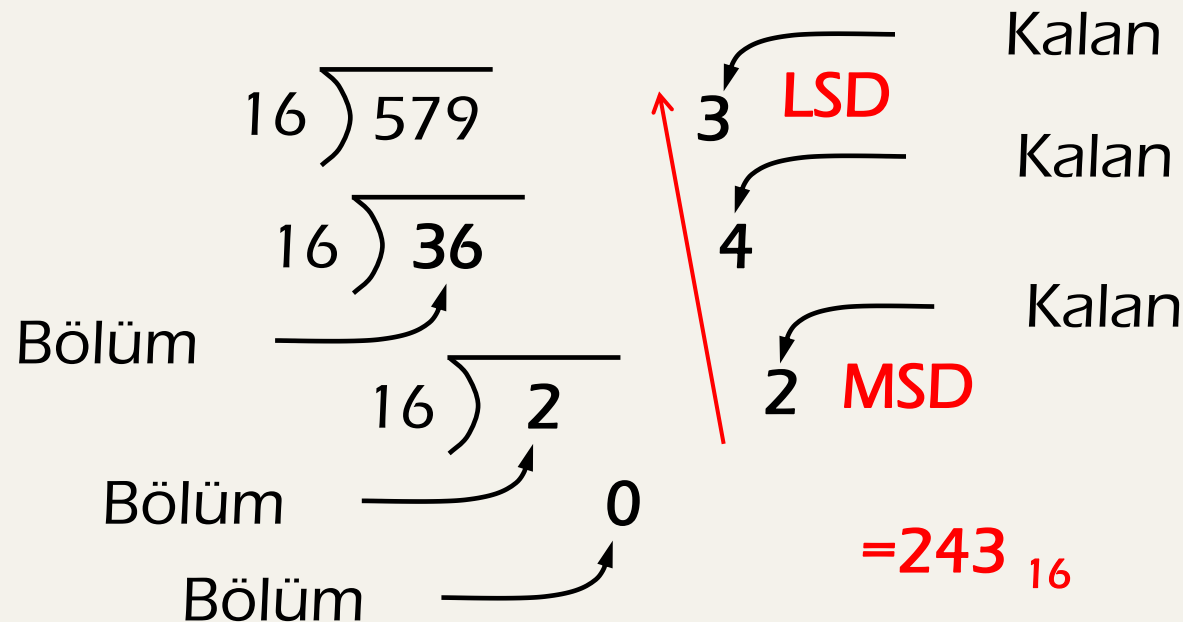
Örnek:

$$\begin{aligned} 243_{16} &= 2 \times (16^2) + 4 \times (16^1) + 3 \times (16^0) \\ &= 512 + 64 + 3 \\ &= 579_{10} \end{aligned}$$

# Onluk-tan Onaltılık-a Dönüşüm

11

- Sürekli sayı tabanına bölme yöntemini kullanın.
  - ▣ Onluk sayıyı 16'ya bölün.
  - ▣ İlk kalan LSB; sonuncusu MSB'dir.
- **Örnek:**  $579_{10}$  Sayısını 16 tabana dönüştürelim,



# Onaltılık-tan İkilik-e Dönüşüm

12

□ Örnek:

$$\begin{aligned} 9F2_{16} &= \quad \quad 9 \quad \quad F \quad \quad 2 \\ &= 1001 \quad 1111 \quad 0010 \\ &= 100111110010_2 \end{aligned}$$

# İkilik-ten Onaltılık-a Dönüşüm

13

- LSB'den başlayarak bitleri dörderli gruplayın.
- Her grubun onaltılık basamak karşılığını yazın.
- Gerekirse, son grubu dörde tamamlayacak kadar sıfırı MSB'nin solundan başlayarak ekleyin.
- Örnek:  $1110100110_2$  sayısını 16'lı tabana dönüştürün

= 0011 1010 0110

= 3 A 6

= 3A6<sub>16</sub>

# İkili Toplama

14

İkili toplamanın kuralı;

$$0 + 0 = 0$$

$$\text{Toplam} = 0, \text{Elde} = 0$$

$$0 + 1 = 1$$

$$\text{Toplam} = 1, \text{Elde} = 0$$

$$1 + 0 = 1$$

$$\text{Toplam} = 1, \text{Elde} = 0$$

$$1 + 1 = 10$$

$$\text{Toplam} = 0, \text{Elde} = 1$$

Önceki toplamdan elde var ise;

$$1 + 0 + 0 = 01$$

$$\text{Toplam} = 1, \text{Elde} = 0$$

$$1 + 0 + 1 = 10$$

$$\text{Toplam} = 0, \text{Elde} = 1$$

$$1 + 1 + 0 = 10$$

$$\text{Toplam} = 0, \text{Elde} = 1$$

$$1 + 1 + 1 = 11$$

$$\text{Toplam} = 1, \text{Elde} = 1$$

Örnek: 00111 ve 10101 ikili sayılarını toplayın. Onlu karşılığını bulun.

$$\begin{array}{r} \textcolor{red}{0} \textcolor{red}{1} \textcolor{red}{1} \textcolor{red}{1} \\ 00111 \\ 10101 \\ \hline 11100 \end{array} \quad \begin{array}{r} 7 \\ 21 \\ \hline 28 \end{array} = 28$$

# İkili Çıkarma

15

İkili çıkarmanın kuralı;

$$0 - 0 = 0$$

$$1 - 1 = 0$$

$$1 - 0 = 1$$

$$10 - 1 = 1 \text{ Borçlu } 1$$

Örnek: 00111 sayısından 10101 sayısını çıkarın onlu karşılığını hesaplayın.

$$\begin{array}{r} \phantom{0}1\phantom{0}1\phantom{0}1 \\ \phantom{0}1\phantom{0}1\phantom{0}1 \\ \underline{00111} \\ 01110 = 14 \end{array}$$

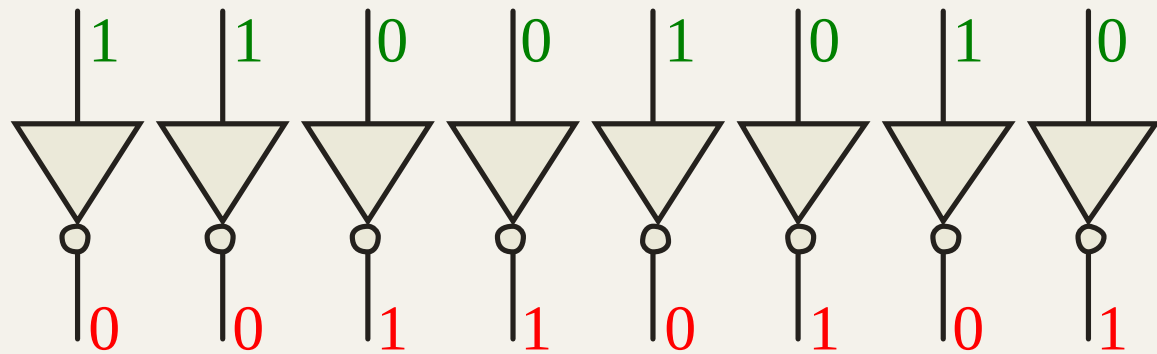
# 1'e Tümleme

16

İkili sayıların 1'e tümlenmesi her bitin tersinin (değilinin) alınması ile yapılır.

Örneğin:      11001010  
                  00110101

Sayısal devrelerde 1'e tümleme değil geçitleri ile yapılır.



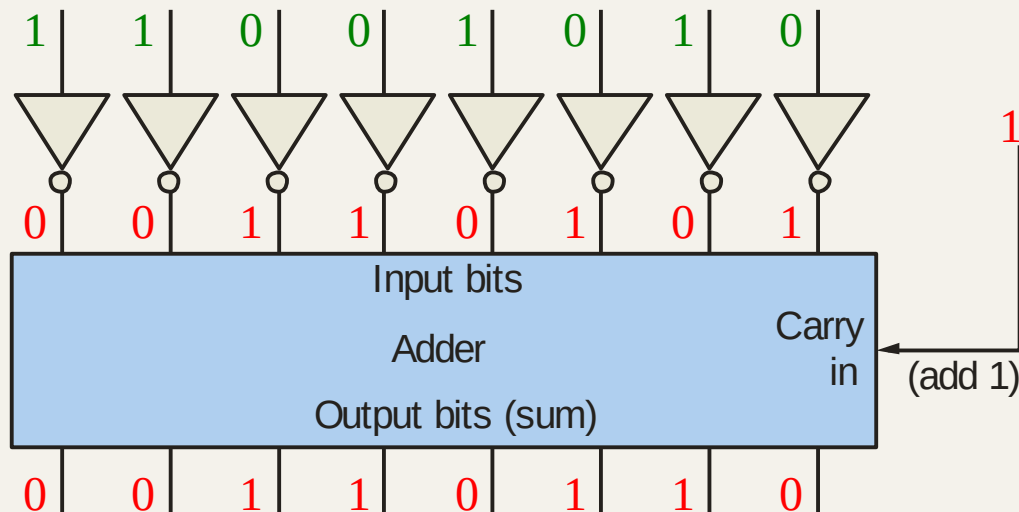
# 2'ye Tümlleme

17

İkili sayıların 2'ye tümleyeni 1'e tümlenmiş haline 1 eklenmesi ile elde edilir.

2'ye tümleyenini alalım;

2'ye tümlleme için 1 toplanır.



$$\begin{array}{r} 11001010 \\ 00110101 \text{ (1'e Tümlenyeni)} \\ \hline +1 \\ 00110110 \text{ (2'ye Tümlenyeni)} \end{array}$$

# İşaretli İkili Sayılar

18

İşaretli sayıları ikili olarak kodlamanın bir çok yöntemi vardır. Tüm yöntemlerde MSB biti işareti gösterir 0 ise pozitif 1 ise negatif sayıdır.

Bilgisayar 2'ye tümleyen formatını kullanarak işaretli sayıları kodlar. Pozitif sayıların işaret biti 0'dır ve sayının büyüklüğü gerçek formdadır. Negatif sayılarda işaret biti 1'dir ve büyüklük 2'ye tümlenmiş formdadır.

Örnek: 58 sayısını 8-bit olarak kodlayalım

İşaret biti → 00111010 (gerçek değer).  
Büyüklük bitleri

# İşaretili İkili Sayılar

19

Negatif sayıların büyüklükleri 2'ye tümlenmiş şekilde kodlanır.

Negatif -58 sayısını kodlayalım:

$$-58 = 11000110 \text{ (complement form)}$$

İşaret biti

Büyüklük Bitleri

Bu kodlama sistemini okumanın kolay yöntemi işaret bitinin 8 bit sayılarsa ağırlığı 128'dir bunu negatif yazın diğer bitlerin ağırlıklarını pozitif yazın ve toplayın.

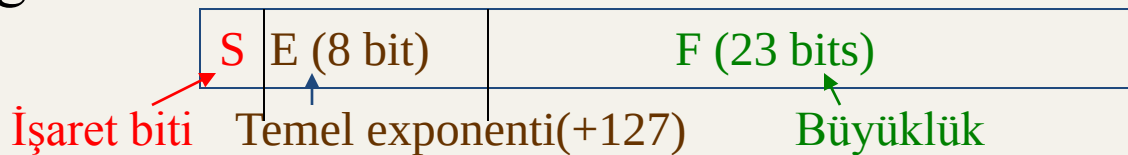
**Örnek:** -58 sayısını işaretili sayı olarak kodlayın.

**Çözüm:** Sütun ağırlıkları:  $-128 \ 64 \ 32 \ 16 \ 8 \ 4 \ 2 \ 1$   
 $1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0$   
 $-128 + 64 \qquad \qquad +4 + 2 \qquad = -58$

# Kayar Noktalı (Floating Point ) Sayılar

20

Kayar noktalı sayılar çok büyük ve çok küçük sayıları ifade etmek için gereklidir. Bu gösterime bilimsel gösterimi denir. 32-bit tek hassas formatı.



**Örnek:** Işığın hızını ikili olarak kodlayın,  $c$ , ( $c = 0.2998 \times 10^9$ )

**Çözüm:** ikili,  $c = 0001\ 0001\ 1101\ 1110\ 1001\ 0101\ 1100\ 0000_2$ .

Bilimsel format,  $c = 1.001\ 1101\ 1110\ 1001\ 0101\ 1100\ 0000 \times 2^{28}$ .

$S = 0$  sayı pozitif.  $E = 28 + 127 = 155_{10} = 1001\ 1011_2$ . F, bit ilk 1 alındıktan sonraki 23 bittir.

floating point gösterimi,  $C =$

|   |          |                              |
|---|----------|------------------------------|
| 0 | 10011011 | 001 1101 1110 1001 0101 1100 |
|---|----------|------------------------------|

# İşaretili Sayılar ile Aritmetik

21

Negatif sayılar 2'ye tümlenmiş olarak kodlandığında toplama ve çıkarma daha basit yapılabilir.

Toplamanın kuralı: iki işaretili sayıyı topla oluşan elde var ise at geri kalan değer işaretili toplamdır.

Örnekler:

$$00011110 = +30$$

$$00001111 = +15$$

$$\hline 00101101 = +45$$

$$00001110 = +14$$

$$11101111 = -17$$

$$\hline 11111101 = -3$$

$$11111111 = -1$$

$$11111000 = -8$$

$$\hline 11111011 = -9$$

 Eldeyi at

# İşaretili Sayılar ile Aritmetik

22

Toplamın biti ayrılan bit sayısından fazla olduğunda büyüklük işaret bitine taşar ve hata oluşur. Bu durum sadece her iki sayı aynı işaret bitine sahipse oluşur.

Örnekler:

|                            |             |                           |
|----------------------------|-------------|---------------------------|
| 01000000 = +64             |             | 10000001 = -127           |
| 01000001 = +65             |             | 10000001 = -127           |
| <hr/>                      |             |                           |
| 10000001 = <del>-127</del> | Eldeyi at → | <del>100000010 = +2</del> |

yanlış! Sonuç ve işaret bitleri hatalı.

# İşaretili Sayılar ile Aritmetik

23

Çıkarmanın kuralı: Çıkanı 2'ye tümle ve çıkarılan ile topla oluşan elde var ise at, sonuç işaretlidir.

Önekler:

$$\begin{array}{rcl} 00011110 & (+30) & 00001110 & (+14) & 11111111 & (-1) \\ - 00001111 & -(+15) & - 11101111 & -(-17) & - 11111000 & -(-8) \\ \hline \end{array}$$

2'ye tümle ve topla:

$$\begin{array}{rcl} 00011110 & = & +30 \\ 11110001 & = & -15 \\ \hline \cancel{1}00001111 & = & +15 \\ \uparrow & & \\ \text{Eldeyi at} & & \end{array} \quad \begin{array}{rcl} 00001110 & = & +14 \\ 00010001 & = & +17 \\ \hline 00011111 & = & +31 \end{array} \quad \begin{array}{rcl} 11111111 & = & -1 \\ 00001000 & = & +8 \\ \hline \cancel{1}00000111 & = & +7 \\ \uparrow & & \\ \text{Eldeyi at} & & \end{array}$$

# Sayı Sistemleri Dönüşüm Tablosu

24

| Onlu | İkili | Sekizli | Onaltılı | BCD       |
|------|-------|---------|----------|-----------|
| 0    | 0     | 0       | 0        | 0         |
| 1    | 1     | 1       | 1        | 0001      |
| 2    | 10    | 2       | 2        | 0010      |
| 3    | 11    | 3       | 3        | 0011      |
| 4    | 100   | 4       | 4        | 0100      |
| 5    | 101   | 5       | 5        | 0101      |
| 6    | 110   | 6       | 6        | 0110      |
| 7    | 111   | 7       | 7        | 0111      |
| 8    | 1000  | 10      | 8        | 1000      |
| 9    | 1001  | 11      | 9        | 1001      |
| 10   | 1010  | 12      | A        | 0001 0000 |
| 11   | 1011  | 13      | B        | 0001 0001 |
| 12   | 1100  | 14      | C        | 0001 0010 |
| 13   | 1101  | 15      | D        | 0001 0011 |
| 14   | 1110  | 16      | E        | 0001 0100 |
| 15   | 1111  | 17      | F        | 0001 0101 |

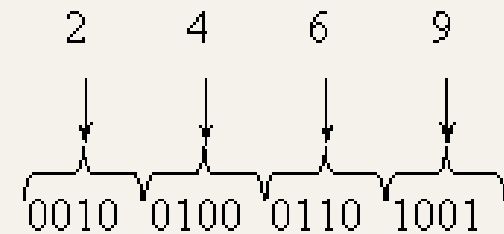
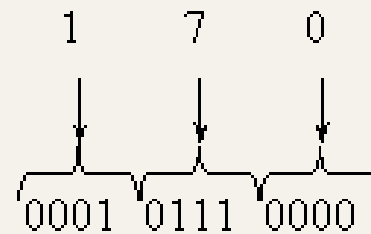
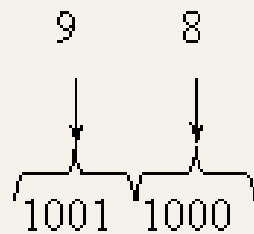
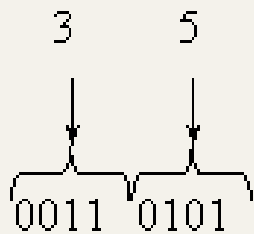
# İKİLİK KODLU ONLUK (İKO) SAYILAR

25

- Yaygın olarak kullanılan adı ile BCD (Binary Coded Decimal) sayılar günlük hayatta kullandığımız sayıları sayısal sistemlere uyarlamak için geliştirilmiş bir kodlama sistemidir.
- 8421 kodu adı da verilen bu kodlama sisteminde 0-9 arası rakamlar 4 bit ikilik sayılar ile kodlanır. Kodlama sırasında ikilik sistemin ağırlıkları kullanılır.
- İkilik kodlu onluk demek, 0'dan 9'a kadar olan sayılardan her birinin, ikilik bir kodla gösterilmesi demektir. 8421 adlandırması dört bitin ikilik ağırlıklarını belirtir

| ONLUK | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    | 9    |
|-------|------|------|------|------|------|------|------|------|------|------|
| İKO   | 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 |

**Örnek:** 35, 98, 170, 2469 sayılarını İKO olarak kodlayın.



# SAYISAL KODLAR

27

- Şu ana kadar öğrendikleriniz dışında da bir çok kod vardır. Bunların bazıları nümerik bazıları ise alfanümerik kodlardır. Nümerik kodlarla sadece sayılar kodlanabilir, alfanümerik kodlarla ise semboller, komutlar ve sayılar kodlanabilir. Bu bölümde İKO'ya göre daha az kullanılan nümerik kodlardan Gray (Yansıtılmış) kodu ve 3-artı kodlarını öğreneceksiniz.
- **GRAY KODU**
- Gray kodu rakamların ağırlığı olmayan ve aritmetik olmayan bir koddur. En önemli özelliği bir durumdan diğer duruma geçerken sadece bir bit değer değiştirebilir. Bu özelliği bazı uygulamalarda özellikle bir durumdan diğer duruma geçerken birden fazla bitin değişmesi ile oluşan hatalı veya şüpheli sonucun istenmediği uygulamalarda kullanılmasına olanak sağlamıştır. Mil dönme hareketinin belirleyen arttırmalı kodlayıcılarda bu kod kullanılmaktadır. Tabloda ikilik, onluk ve gray kodu karşılıkları gösterilmiştir.

# Gray Kodlar

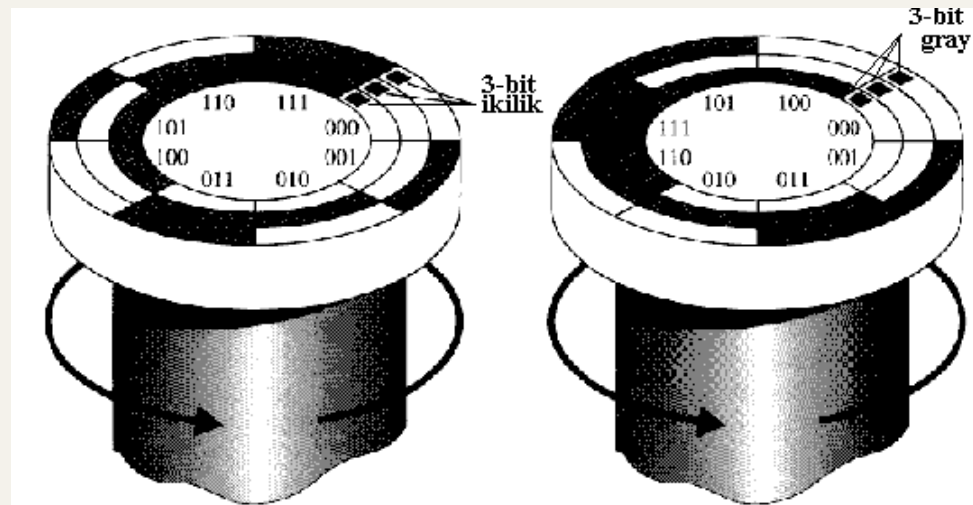
28

- Gray kodlar *minimum değişimli* kodlardır.
  - ▣ Bir sayı gösteriminden diğerine yalnızca bir bit değişir.
  - ▣ Ana kullanım amacı rastgele olmayan giriş değerlerinin beklendiği, diğer bir deyişle,  $n$  değerinin  $n-1$  veya  $n+1$ 'e değiştiği nümerik giriş kodlama uygulamalarıdır.
    - Freze tezgahı tabla konumu
    - Açısal mil konumu

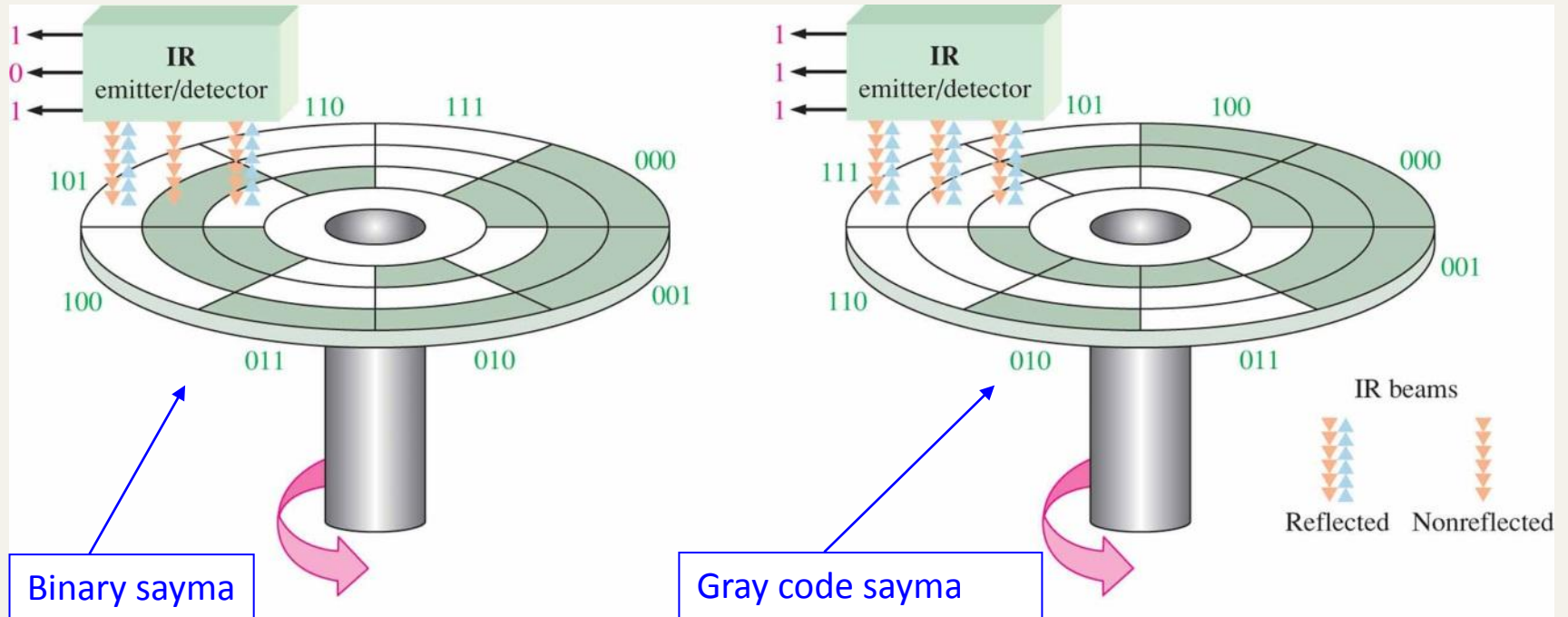
# Gray Kodlar

29

| İkilik | Gray |
|--------|------|
| 0000   | 0000 |
| 0001   | 0001 |
| 0010   | 0011 |
| 0011   | 0010 |
| 0100   | 0110 |
| 0101   | 0111 |
| 0110   | 0101 |
| 0111   | 0100 |
| 1000   | 1100 |
| 1001   | 1101 |
| 1010   | 1111 |
| 1011   | 1110 |
| 1100   | 1010 |
| 1101   | 1011 |
| 1110   | 1001 |
| 1111   | 1000 |



3 bitlik arttırmalı enkoder



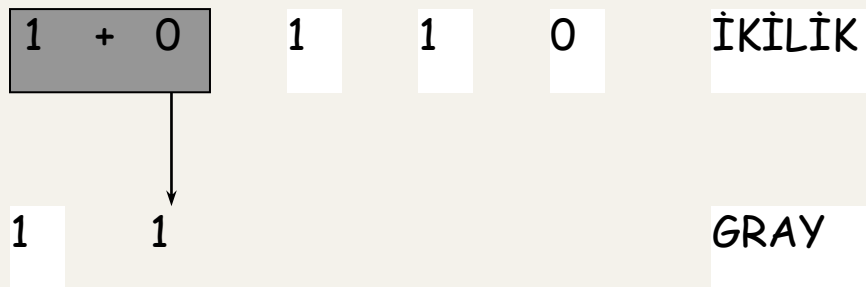
# İKİLİK-GRAY DÖNÜŞÜM

31

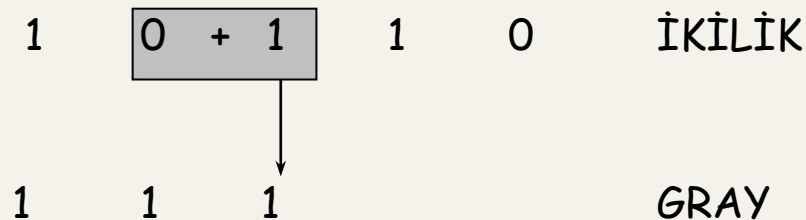
- Gray koduna ikilikten dönüşüm yapmak için aşağıdaki kurallar uygulanır.
- En yüksek değerli bit (MSB), en soldaki bit, ikilik ile Gray'de aynıdır.
- Soldan sağa giderken yan yana olan iki bit topla ve gray biti olarak yaz. Oluşan eldeyi at.
- **ÖRNEK 1:** 101102 sayısını gray koduna dönüştürün.
- *Adım 1:* MSB'yi ikilikten aynen Gray'e yaz.



- *Adım 2:* Soldan başlayarak ilk iki biti topla, graye soldan ikinci bit olarak yaz.



- *Adım 3:* Soldan başlayarak ikinci ve üçüncü biti topla, graye soldan üçüncü bit olarak yaz.



- Adım 4: Soldan başlayarak üçüncü ve dördüncü biti topla, graye soldan dördüncü bit olarak yaz.

|   |   |       |   |        |
|---|---|-------|---|--------|
| 1 | 0 | 1 + 1 | 0 | İKİLİK |
|   |   | ↓     |   |        |
| 1 | 1 | 1     | 0 | GRAY   |

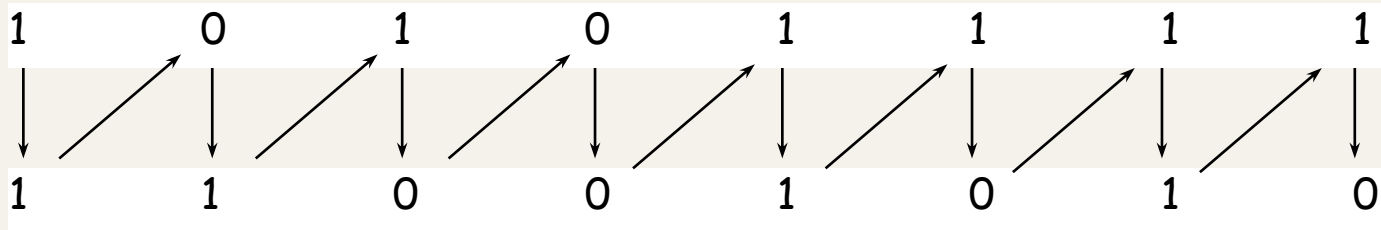
- Adım 5: Soldan başlayarak dördüncü ve beşinci biti topla, graye soldan beşinci bit olarak yaz.

|   |   |   |       |   |        |
|---|---|---|-------|---|--------|
| 1 | 0 | 1 | 1 + 0 | 0 | İKİLİK |
|   |   |   | ↓     |   |        |
| 1 | 1 | 1 | 0     | 1 | GRAY   |

# GRAY-İKİLİK DÖNÜŞÜM

34

- ❑ İkilik-gray dönüşümüne benzer bir kural uygulanarak dönüşüm gerçekleştirir. Dönüşümün kurallarını aşağıdaki gibi özetleyebiliriz.
- ❑ En yüksek değerli bit (MSB), en soldaki bit, ikilik ile grayde aynıdır.
- ❑ İkilinin en soldaki biti ile Gray kodunun soldan ikinci biti toplanarak ikiliye ikinci bit olarak yazılır. Elde varsa atılır.
- ❑ **Örnek:** Aşağıdaki dönüşümü yapın.



# 3-ARTI KOD

35

- İKO koda benzer bir koddur, sadece İKO'nun 4 bitlik ikilik kodlanmış sayısına 3 eklenerek elde edilir. Bitlerin herhangi bir ağırlığı yoktur. Tabloda 15'e kadar İKO ve 3-ARTI kodlar verilmiştir. En büyük özelliği bu kod sisteminde hiçbir zaman tüm bitlerin aynı anda sıfır olmamasıdır.

| Onlu | İkili | Sekizli | Onaltılı | BCD       | 3-Artı    |
|------|-------|---------|----------|-----------|-----------|
| 0    | 0     | 0       | 0        | 0         | 0011      |
| 1    | 1     | 1       | 1        | 0001      | 0100      |
| 2    | 10    | 2       | 2        | 0010      | 0101      |
| 3    | 11    | 3       | 3        | 0011      | 0110      |
| 4    | 100   | 4       | 4        | 0100      | 0111      |
| 5    | 101   | 5       | 5        | 0101      | 1000      |
| 6    | 110   | 6       | 6        | 0110      | 1001      |
| 7    | 111   | 7       | 7        | 0111      | 1010      |
| 8    | 1000  | 10      | 8        | 1000      | 1011      |
| 9    | 1001  | 11      | 9        | 1001      | 1100      |
| 10   | 1010  | 12      | A        | 0001 0000 | 0100 0011 |
| 11   | 1011  | 13      | B        | 0001 0001 | 0100 0100 |
| 12   | 1100  | 14      | C        | 0001 0010 | 0100 0101 |
| 13   | 1101  | 15      | D        | 0001 0011 | 0100 0110 |
| 14   | 1110  | 16      | E        | 0001 0100 | 0100 0111 |
| 15   | 1111  | 17      | F        | 0001 0101 | 0100 1000 |

# Alfasayısal Gösterim

36

- Alfasayısal ve sayısal karakterleri göstermek için kullanılan ikilik kodlardır.
- En çok kullanılan iki gösterim:
  - ▣ ASCII, 7 bit kod, 128 sembol
  - ▣ EBCDIC, 8 bit kod, 256 sembol
- Sembol değerleri karşılaştırılırken sorunlar çıkabilir
- ASCII sistemde 'A' ile 'a'nın karşılaştırılması EBCDIC sistemde farklı sonuçlara neden olabilir.

# Eşlik Biti

37

- Veri iletim hatalarını algılamak için ASCII koda iliştirilmiş fazladan bir bit bulunur.
  - ▣ Karakterdeki 1'lerin sayısı çift ise  $P = 0$ , tekse  $P = 1$  (çift eşlik biti)
  - ▣ Karakterdeki 1'lerin sayısı tek ise  $P = 0$ , çiftse  $P = 1$  (tek eşlik biti)
- Tek bir bit değişirse, algılamamanın sonunda eşlik biti yanlış olacaktır.

|                   | <u>Tek eşlik</u> | <u>Çift eşlik</u> |
|-------------------|------------------|-------------------|
| ASCII A = 1000001 | 01000001         | 11000001          |
| ASCII T = 1010100 | 11010100         | 01010100          |

# Alfasayısal Gösterim

38

- Alfasayısal ve sayısal karakterleri göstermek için kullanılan ikilik kodlardır.
- En çok kullanılan iki gösterim:
  - ▣ ASCII, 7 bit kod, 128 sembol
  - ▣ EBCDIC, 8 bit kod, 256 sembol
- Sembol değerleri karşılaştırılırken sorunlar çıkabilir (collation:derleme)
  - ▣ ASCII sistemde 'A' ile 'a'nın karşılaştırılması EBCDIC sistemde farklı sonuçlara neden olabilir.

# Eşlik Biti

39

- Veri iletim hatalarını algılamak için ASCII koda iliştilirilmiş fazladan bir bit bulunur.
  - ▣ Karakterdeki 1'lerin sayısı çift ise  $P = 0$ , tekse  $P = 1$  (çift eşlik biti)
  - ▣ Karakterdeki 1'lerin sayısı tek ise  $P = 0$ , çiftse  $P = 1$  (tek eşlik biti)
- Tek bir bit değişirse, algılamamanın sonunda eşlik biti yanlış olacaktır.

|                   | <u>Çift eşlik</u> | <u>Tek eşlik</u> |
|-------------------|-------------------|------------------|
| ASCII A = 1000001 | 01000001          | 11000001         |
| ASCII T = 1010100 | 11010100          | 01010100         |

# ASCII

40

| MSB<br>LSB |     |                 |     |     |     |     |     |     |
|------------|-----|-----------------|-----|-----|-----|-----|-----|-----|
|            | 000 | 001             | 010 | 011 | 100 | 101 | 110 | 111 |
| 0000       | NUL | DLE             | SP  | 0   | @   | P   | `   | p   |
| 0001       | SOH | DC <sub>1</sub> | !   | 1   | A   | Q   | a   | q   |
| 0010       | STX | DC <sub>2</sub> | "   | 2   | B   | R   | b   | r   |
| 0011       | ETX | DC <sub>3</sub> | #   | 3   | C   | S   | c   | s   |
| 0100       | EOT | DC <sub>4</sub> | \$  | 4   | D   | T   | d   | t   |
| 0101       | ENQ | NAK             | %   | 5   | E   | U   | e   | u   |
| 0110       | ACK | SYN             | &   | 6   | F   | V   | f   | v   |
| 0111       | BEL | ETB             | '   | 7   | G   | W   | g   | w   |
| 1000       | BS  | CAN             | (   | 8   | H   | X   | h   | x   |
| 1001       | HT  | EM              | )   | 9   | I   | Y   | i   | y   |
| 1010       | LF  | SUB             | *   | :   | J   | Z   | j   | z   |
| 1011       | VT  | ESC             | +   | ;   | K   | [   | k   | {   |
| 1100       | FF  | FS              | ,   | <   | L   | \   | l   |     |
| 1101       | CR  | GS              | -   | =   | M   | ]   | m   | }   |
| 1110       | SO  | RS              | .   | >   | N   | ↑   | n   | ~   |
| 1111       | SI  | US              | /   | ?   | O   | —   | o   | DEL |

Definitions of control abbreviations:

|                                  |                           |     |                      |
|----------------------------------|---------------------------|-----|----------------------|
| ACK                              | Acknowledge               | FS  | Form separator       |
| BEL                              | Bell                      | GS  | Group separator      |
| BS                               | Backspace                 | HT  | Horizontal tab       |
| CAN                              | Cancel                    | LF  | Line feed            |
| CR                               | Carriage return           | NAK | Negative acknowledge |
| DC <sub>1</sub> –DC <sub>4</sub> | Direct control            | NUL | Null                 |
| DEL                              | Delete idle               | RS  | Record separator     |
| DLE                              | Data link escape          | SI  | Shift in             |
| EM                               | End of medium             | SO  | Shift out            |
| ENQ                              | Enquiry                   | SOH | Start of heading     |
| EOT                              | End of transmission       | SP  | Space                |
| ESC                              | Escape                    | STX | Start text           |
| ETB                              | End of transmission block | SUB | Substitute           |
| ETX                              | End text                  | SYN | Synchronous idle     |
| FF                               | Form feed                 | US  | Unit separator       |
|                                  |                           | VT  | Vertical tab         |