

3. A KÓDOLÁS

A kódolás folyamán az információt adattá alakítjuk, valamely szimbólum hozzárendelési szabály szerint. Az információ egyes elemeihez csak egy szimbólum rendelhető. A kódolás magában foglalja a hozzárendelési szabályt és a szimbólumkészletet. Hétköznapi életünkben is sok példát tudunk mondani a kódolásra. Az ASCII karakterkódolás esetén 256 karakterhez (betűk, számok, írásjelek és vezérlő kód) rendelünk egy-egy nyolc bites számot. Településekhez is rendelhetünk szimbólumokat, ezek a települések irányítószámai. Gondolataink megfogalmazása után hangként jelenik meg a kódolt információ. A már kódolt információt is újra tudjuk kódolni egy új eljárásban, ez a művelet a transzkódolás. A kódolás inverz művelete a dekódolás.

A kódolt információ jellege szerint a kódok lehetnek:

- Numerikus kódok, amelyek számok kifejezésére alkalmasak. Pl:
 - Bináris: a digitális technikában kétértékű kódokkal dolgozunk. A kétállapotú áramköri elemek működése igazodik ehhez legjobban. Egy-egy kétállapotú elem bináris alapegységet képez, ez az úgynevezett bit, melynek értéke 0 és 1 lehet. A digitális technikában az adat megadásához több bitre van szükség. Ilyen bitsorozat szervezés eredménye a byte, mely 8 bitet tartalmaz.
 - BCD kódok;
 - Hexadecimális kód;
 - Komplement kód.
- Alfabetikus kódok: pl.: ASCII¹, ahol egy-egy 8 bites (vagyis 1 byte-os) szám jeleníti meg az alfabetikus karaktert.

Bináris prefixumok

Ellentétben az SI prefixumokkal (melyek 10 olyan hatványai, ahol a kitevő osztható hárommal, maradék nélkül), a digitális technikában a kettő hatványait (ahol a hatványkitevő 10, illetve 10 és egy pozitív egész szám szorzata) alkalmazzuk, hiszen a számrendszer alapja a kettes.

Pl.:

$$1 \text{ kibibájt} = 2^{10} \text{ byte} = 1024 \text{ byte}$$

$$1 \text{ Mebibyte} = 1 \text{ kibibyte} \cdot 2^{10} = 2^{10} \cdot 2^{10} \text{ byte} = 2^{20} \text{ byte} = 1048576 \text{ byte}$$

Szimbólum	Név	Jelentés	Értéke
Ki	kibi-	bináris kilo-	2^{10}
Mi	mebi-	bináris mega-	2^{20}
Gi	gibi-	bináris giga-	2^{30}
Ti	tebi-	bináris tera-	2^{40}
Pi	pebi-	bináris peta-	2^{50}
Ei	exbi-	bináris exa-	2^{60}
Zi	zebi-	bináris zetta-	2^{70}
Yi	yobi-	bináris yotta-	2^{80}

3.1. táblázat

1 ASCII: <https://hu.wikipedia.org/wiki/ASCII>

3.1. Numerikus kódok

A numerikusnak nevezzük azokat a kódrendszereket, amelyekben kizárólag számjegyes információkat kódolnak.

3.1.1. Bináris kód

A legfontosabb kód, lényegében minden további kódnak az alapja. Bináris kódú felírásakor a numerikus információt bináris számjegyekkel adjuk meg. Ilyen például az, amikor a tízes számrendszerbeli számhoz hozzárendeljük a bináris megfelelőjét. $\rightarrow$ decimális – bináris átalakítás.

3.1.2. Hexadecimális kód

A numerikus információt hexadecimális számrendszerben ábrázoljuk.

3.1.3. Komplementum kód

A komplementum szó alapja a francia *complémentaire* szó, jelentése (egymást) kiegészítő. A bináris kódoknál 1-re, illetve 2-re lehet elvégezni a kiegészítést.

A komplementum kód alkalmazása

A pozitív számokat egyenes kódban ábrázoljuk. Amennyiben a negatív számokat egyenes kódban ábrázoljuk, akkor előjelbit (a bináris szám előtti 1-es) alkalmazása válik szükségessé.

A negatív számokat emellett úgynevezett 1-es komplementum kódban is tudjuk ábrázolni. Ebben az esetben az egyenes kód bitjeit invertáljuk. Ekkor lényegében az egyenes kód és az 1-es komplementum kód bitjei 1-re egészítik ki egymást. Az előjelbit változatlan.

Egyenes kódú pozitív szám:	$47_{10} = 101111_2$
Egyenes kódú negatív szám:	$-47_{10} = 1\ 101111_2$
1-es komplementum kódú negatív szám:	$-47_{10} = 1\ 010000_2$

A 2-es komplementum kód képzése: az 1-es komplementumú kódhoz hozzáadunk 1-et, miközben az előjelbit változatlan.

$-47 =$	1 10111	egyenes kód
	1 01000	1-es komplementum kód
	+1	
	1 01001	2-es komplementum kód

3.1.4. BCD kódok (Binary Code Digit)

A BCD kódokat úgy képezzük, hogy a egy számjegyű decimális számokhoz (0-9) 4, vagy több bitből álló kódszavakat rendelünk.

3.1.4. a) N-BCD (Normál BCD) kód (Normal Binary Code Digit)

A decimális – BCD átalakítást akképp tudjuk elvégezni, hogy a decimális szám helyiértékenkénti bináris megfelelőjét írjuk fel. Ebből következik – mivel a decimális számrendszerben 0 és 9 közötti számok vannak, az előtti 4 bites számok az úgynevezett nem létező kódok. Nevezik még egyenes kódnak, illetve 8421 BCD kódnak.

BCD kódok:

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

BCD nem létező kódok:

10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

Decimális – BCD átalakítás.

Pl: 6257₁₀

$$\begin{array}{cccc} 6 & 2 & 5 & 7 \\ 0110 & 0010 & 0101 & 0111 \end{array}_{BCD} = 110001001010111_{BCD}$$

N-BCD (Normál BCD) kódú számok összeadása

Amennyiben a két összeadandó szám összege egy helyiértéken ábrázolható, vagyis nem nagyobb, mint 9, az összeadás két bináris szám összeadása. Az összeadás helyiértékenként zajlik.

$$\begin{array}{rcl} 3 & 0011 & \\ +6 & 0110 & \\ \hline 9 & 1001 & \end{array} \qquad \begin{array}{rcl} 34 & 0011 & 0100 \\ +45 & 0100 & 0101 \\ \hline 79 & 0111 & 0101 \end{array}$$

Ha az összeadáskor nem létező kód keletkezik (nagyobb az összeg, mint 9), akkor a kapott 4 bites csoporthoz 6-ot kell hozzáadni. A keletkezett átvitelt hozzáadjuk a következő helyiértékhez. Ugyanezt az eljárást kell követnünk a több helyiértékes számok esetében is.

$$\begin{array}{rcl} 8 & 1000 & \\ +3 & 0011 & \\ \hline 11 & 1011 & \text{nemlétező kód} \\ +6 & 0110 & +6 a korrekció \\ \hline 0001 & 0001 & \text{a 11 BCD kódban} \end{array} \qquad \begin{array}{rcl} 65 & 0110 & 0101 \\ +27 & 0010 & 0111 \\ \hline 92 & 1000 & 1100 \\ +6 & & 0110 \\ \hline 1001 & 0010 & \text{nemlétező kód (1100)} \\ & & +6 a korrekció \\ & & \text{a 92 BCD kódban} \end{array}$$

3.1.4. b) Háromtöbblletes (Excess-3, XS-3) kód

A decimális számokhoz a hárommal nagyobb szám bináris megfelelőjét rendeljük (2.2. táblázat). A háromtöbblletes kód csak 10 (0–9) kódot alkalmaz, mint a BCD kód, a többi nemlétező kód. A tíz bináris (0 – 9 decimális) számjegy középvonalára nézve (a 4-es és a 5-ös szám között húzott szaggatott vonal) a tükrös kódok egymásnak komplementeinek (bitenkénti inverzei).

pl.: 26_{10}

2	6
+3	+3
5	9

A műveletet helyiértékenként végezzük,
majd bináris kódban felírjuk a kapott számot

0101	1001
------	------

$26_{10} = 1011001_{(XS-3)}$

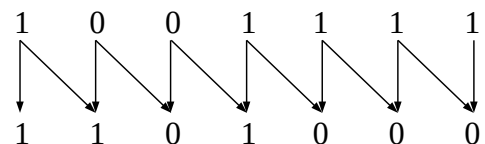
3.1.4. c) Gray- (N-Gray-, Normal-Gray-) kód

A Gray-kód olyan kód, amelyben a decimális számokhoz rendelt – egymás után következő – bináris számok csak egy bitben különböznek. Ezt a kódot minimális változású kódnak, vagy egylépéses kódnak is nevezik.

Decimális (bináris) – Gray átalakítás

- Képezzük a decimális számból a bináris megfelelőjét!
- A Gray-kódú felírás
 - az első bit ugyanaz, mint a bináris szám első bitje;
 - a második bit a bináris szám első és második bitjének KIZÁRÓ-VAGY kapcsolata;
 - a harmadik bit a bináris szám második és harmadik bitjének KIZÁRÓ-VAGY kapcsolata;
 - a többi bit esetében is hasonlóan járunk el, rendre balról jobbra haladva.

Pl: $79_{10} = 1001111_2 = 1101000_{(Gray)}$



Gray – bináris átalakítás

- az első bit ugyanaz, mint a Gray-kódú szám első bitje;
- ha a második bit 0, akkor a második bit ugyanaz, mint az első;
- ha a második bit 1, akkor a második bit az első bit invertált értéke;
- a többi bit esetében is hasonlóan járunk el, rendre balról jobbra haladva.

Pl: $1101000_{(Gray)} = 1001111_2$

1 1 0 1 0 0 0 (Gray)

1 0 0 1 1 1 1 ₂

A bináris szám:

- első bitje: ugyanaz, mint a Gray-kódú számé;
- második bitje: mivel a Gray-kódú szám második bitje 1, ezért a bináris szám előző bitjének inverzével egyenlő ($1 \rightarrow 0$);
- harmadik bitje: mivel a Gray-kódú szám harmadik bitje 0, ezért ugyanaz, mint a bináris szám előző bitje ($0 \rightarrow 0$);
- negyedik bitje: mivel a Gray-kódú szám negyedik bitje 1, ezért a bináris szám előző bitjének inverzével egyenlő ($0 \rightarrow 1$);
- ötödik bitje: mivel a Gray-kódú szám ötödik bitje 0, ezért ugyanaz, mint a bináris szám előző bitje ($1 \rightarrow 1$);
- hatodik bitje: mivel a Gray-kódú szám hatodik bitje 0, ezért ugyanaz, mint a bináris szám előző bitje ($1 \rightarrow 1$);
- hetedik bitje: mivel a Gray-kódú szám hetedik bitje 0, ezért ugyanaz, mint a bináris szám előző bitje ($1 \rightarrow 1$).

	N-BCD 8–4–2–1 egyenes				Stibitz 3 többletes Excess-3, XS-3				Gray				Aiken 2–4–2–1			
0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
1	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0	1
2	0	0	1	0	0	1	0	1	0	0	1	1	0	0	1	0
3	0	0	1	1	0	1	1	0	0	0	1	0	0	0	1	1
4	0	1	0	0	0	1	1	1	0	1	1	0	0	1	0	0
5	0	1	0	1	1	0	0	0	0	1	1	1	1	0	1	1
6	0	1	1	0	1	0	0	1	0	1	0	1	1	1	0	0
7	0	1	1	1	1	0	1	0	0	1	0	0	1	1	0	1
8	1	0	0	0	1	0	1	1	1	1	0	0	1	1	1	0
9	1	0	0	1	1	1	0	0	1	1	0	1	1	1	1	1
10	1	0	1	0	1	1	0	1	1	1	1	1	–	–	–	–
11	1	0	1	1	1	1	1	0	1	1	1	0	–	–	–	–
12	1	1	0	0	1	1	1	1	1	0	1	0	–	–	–	–
13	1	1	0	1	0	0	0	0	1	0	1	1	–	–	–	–
14	1	1	1	0	0	0	0	1	1	0	0	1	–	–	–	–
15	1	1	1	1	0	0	1	0	1	0	0	0	–	–	–	–

3.2. táblázat A BCD kódok összefoglaló táblázata (keretekben a nem létező kódok)

3.1.4. d) Aiken kód

Olyan kódrendszer, melyben az egyes bitek súlyozása: 2–4–2–1, melyből következik, hogy a legnagyobb kódolható decimális szám a 9. A kódok 0–4 tartományban azonosak a bináris megfelelővel. A tíz bináris számjegy középvonalára nézve (a 4-es és a 5-ös szám között húzott szaggatott vonal) a tükrös kódok egymásnak komplementesei (bitenkénti inverzei).