*Suma {połączenie) zbiorów S* oraz *T (S* u 7') to zbiór, którego elementy należą do iS' lub Uo *T;* np.

{1, 3, 4} u {0, 3, 5) M {0, 1, 3, 4, 5}

*Uzupełnienie zbioru S (S')* to zbiór elementów nie należących do *S,* np. jeśli 1= {0, 1, 2, 3, 4, 5}, to

{1, 3, 4}'={0,2, 5}

Zbiory *S* oraz *T* są *rozłączne,* jeśli 5 *n T* = 0-

***Różnica** zbiorów S* oraz *T (S-T* lub S| 7') to zbiór, którego elementy należą do S i nie należą do *T* np.

{1,3, 4}-{0,3, 5} -={1,4}

Można więc **napisać:** *S" -I—S.*

Jeśli we wzorach (1-1) do (1-16) zastąpi się sumę logiczną-—-sumą zbiorów, iloczyn logiczny — iloczynem zbiorów, negację — uzupełnieniem, stalą 0 — zbiorem pustym 0, stalą 1—zbiorem peinym /, to zależności te, podobnie jak i inne wzory dwu elementów ej algebry Boole'a( będą obowiązywały dla algebry zbiorów.

## 1.3. METODY KODOWANIA

### 1.3.1. SYSTEMY ZAPISU LICZB

Informację na wejścia i wyjściu układu przełączającego często dogodnie jest przedstawiać w postaci liczbowej, co umożliwia dyskretny charakter sygnałów. Jednakże stosowane do tego opisu liczby nie zawsze mogą być liczbami dziesiętnymi, skoro same sygnały są opisywane za pomocą dwóch tylko symboli — U i 1. Powstaje więc konieczność wprowadzenia *systemu dwójkowego zapisu liczb.*

Stosowany powszechnie *system dziesiętny zapisu* operuje dziesięcioma różnymi znakami dla przedstawienia cyfr, a każdej cyfrze, w zależności od jej pozycji względem przecinka, jest przypisana *waga,* będąca odpowiednią potęgą podstawy 10. Zapis liczby jest więc umownym zapisem współczynników przy odpowiednich potęgach 10

$$\text{-ŁlO} = \text{"«} \cdots \text{«2«l\#O} = \sum_{i=0}^{n} a_i \cdot 10^i$$

np.

2504 -: 240³+S;10³+0-10¹+4-I<J°

W liczbach ułamkowych podstawa występuje w potęgach ujemnych, a zatem ogólny zapis liczby dziesiętnej ma postać

u

L , 0 = «H ••• tfj«oi s_ifl_j ... a^i == 2_, «:-10ⁱ

Przyjęcie za podstawę liczby 10 jest sprawą umowną (prawdopodobnie podyktowaną niegdyś liczbą palców u rąk); równie dobrze może to być każda inna liczba, przy czym wiąże się to z liczbą różnych znaków-cyfr, która zawsze jest taka jak podstawa. Tak więc ogólny zapis liczb tworzonych na takiej zasadzie jak dziesiętna ma postać

L, = b„ ... &,&o.ft-i&-2 ••• ᵇ-> - Y. ˡ>rPˡ

*P* oznacza podstawę, a *P'—wagę* pozycji *i*.

Przy *P* -•- *2* dowolne liczby mogą być zapisywane za pomocą tylko dwóch znaków (zwykle 0 i 1), np.

10110 - 1.2⁴+G.2ᵃ+1.2³+1-2¹+0.2°

czyli $1011Ó_2=22_{10}$.

Zmniejszenie liczby znaków zostało okupione zwiększeniem ilości cyfr w liczbie, jednak dla maszyn pamiętanie i przetwarzanie licznych prostych sygnałów dwuwartościowych jest zawsze łatwiejsze niż operacje z mniejszą liczbą sygnałów wielowartościowych. System zapisu liczb *z P — 2* nosi nazwę *systemu dwójkowego (binarnego)*; podobnie można tworzyć system trójkowy, czwórkowy itd. Oto kilka liczb zapisanych w systemie dziesiętnym i dwójkowym:

| | | | |
|---|---|---|---|
| 0— | 0 | 7— | 111 |
| 1— | 1 | 8— | 1000 |
| 2— | 10 | 9— | 1001 |
| 3— | U | 10— | 1010 |
| 4— | 100 | | |
| 5— | 101 | 16— | 1000Ó |
| 6— | 110 | 32— | 100000 |

Warto zapamiętać, że liczba *2"* w systemie dwójkowym to jedynka i *n* zer.

Istnieje wicie sposobów konwersji liczb z systemu dziesiętnego do dwójkowego i odwrotnie, ale najprostsza jest metoda polegająca na sumowaniu albo wydzielaniu wag. Należy tu jedynie pamiętać, że pozycje liczby na lewo od przecinka mają wagi, kolejno: 1, 2, 4, S, 16, 32, .,., a na prawo od przecinka: 1/2, 1/4, 1/S itd. Zamiana liczby dwójkowej na dziesiętną polega na dodawaniu tych wag, dla których cyfra ma wartość 1, np.

10110, 1

1 - 1/2      101 10,1

1 — 2    lub    16 + 4 + 2+1/2 - 22'$^{,2}$

1 — 4

1 — 16

22'$^{,2}$

Zamiana liczby dziesiętnej na dwójkową polega na wyszukaniu największej liczby $2^k$, takiej że 2* ^ $L_{ia}$, oraz na sprawdzaniu, czy następne, mniejsze liczby 2' mieszczą się w różnicy $L_{l0}$~2*, np.:

| | | |
|---|---|---|
| 26,75 | 16 < 26,75 < 32 | |
| | 16 | —1 |
| | 16 + 8 - 24 < 26,75 | —1 1 |
| | 24 + 4 > 26,75 | —1 1 0 |
| | 24 + 2 = 26 < 26,75 | —1 1 0 1 |
| | 26 + 1 > 26,75 | - - 1 1 0 1 0 |
| | 26 + 1/2 = 26,5 < 26,75 | — 1 1 0 1 0,1 |
| | 26,5+1/4 - 26,75 | — 1 1 0 1 0,1 1 |

Oczywiście w praktyce nie robi się tak rozbudowanego zapisu, gdyż większość działań można wykonać w pamięci.

Liczby dwójkowe są zwykle długie, co utrudnia zapis, zwiększa możliwość pomyłek i wydłuża czas przy opisywaniu sygnałów. Aby uniknąć tych wad, wprowadza się niekiedy (wyłącznie w charakterze pomocniczym, przy sporządzaniu dokumentacji) grupowanie trzech lub czterech cyfr dwójkowych i oznaczanie ich jednym symbolem.

Łatwo zauważyć, że sprowadza się to do wyrażenia liczby w *systemie ósemkowym lub szestnastkowym,* np.:

$$87_{,o} = 1010111_{3}. = 001\ \ 010\ \ 1\ I\ 1_{3} = 127_{a}$$

$$10101\ 1\ 1_{a} = 0101\ \ 0\ 111_{,} = 67_{16}$$

"W zapisie ósemkowym stosuje się oczywiście znaki systemu dziesiętnego, ale dla zapisu szesnastkowego znaków tych jest zbyt mało. Do oznaczenia cyfr większych od 9 (10, 11,,..,15) są zazwyczaj używane litery *A, B,* ..., *F,* np.

$$156_{10} = 1001\ \ 1100_{3} = 9Ci«$$

W niektórych urządzeniach cyfrowych operuje się liczbami binarnymi o 8 cyfrach binarnych; grupa tych ośmiu cyfr to tzw. *byte* (czyt, bait). Ponieważ cyfra binarna występuje często pod nazwą *bit,* więc byte to 8 bitów.

Niekiedy jest stosowany system zapisu liczb o podstawie *P* «= 1 - *system jedynkozoy (unitarny).* Wszystkie wagi *P'* tego systemu mają wartość 1, więc liczbę *N* wyraża się przez ciąg *N* jedynek, np.:

**3»-111j 5O = 11111,**

W celu oddzielenia od siebie takich ciągów stosuje się często zera, lecz mają one tu charakter przerywników lub wypełniaczy, a nie cyfr.

Czynność przypisywania różnym informacjom odpowiednich symboli jest nazywana *kodowaniem,* a zestaw symboli dla danej informacji — *kodem* tej informacji, Opisany wyżej dwójkowy system zapisu daje więc *kod dwójkowy* dowolnych liczb; jest to tzw. *naturalny kod dwójkowy.* Nazwę tę trzeba wprowadzić dla odróżnienia od innych kodów dwójkowych, gdyż możliwych relacji między liczbą, a ciągiem zer i jedynek może być bardzo dużo, Z wielu tych relacji korzysta się w praktyce dla uzyskania pewnych specjalnych cech kodu, potrzebnych w danym przypadku. Najczęściej chodzi tu o takie dobranie kodu, by urządzenia do dalszego przetwarzania informacji byty możliwie najprostsze, a więc — by łatwa była realizacja:

— operacji arytmetycznych,
—• zliczania impulsów,
— odczytywania kodu (dekodowania),
— zabezpieczania przed zakłóceniami sygnału,
— przystosowania do urządzeń zewnętrznych, itp.

### 1.3.2. KODY DWÓJKOWO-DZIESIĘTNE

Podstawową zaletą naturalnego kodu dwójkowego jest prostota budowanych w tym **kodzie** układów zliczania impulsów, bardzo często występujących w urządzeniach cyfrowych. Jednakże te proste układy mają właściwość zliczania ***modulo*** 2" (H •—• liczba bitów kodu), tzn. liczby JV, gdy $N$ < 2" są w nich przedstawiane w naturalnym kodzie dwójko- •wym, a gdy $N$ 5= 2" — są przedstawione jako reszty z dzielenia $N$ przez 2". Na przykład w 4-bitowym urządzeniu liczba dziesiętna 12 występuje jako 1 1 0 0, natomiast liczba 20 jako 0 1 0 0 (20—*2** ----- 4). Dobierając odpowiednio duże *n* można uniknąć niejednoznaczności, ale taki sposób pracy układów nie zawsze jest odpowiedni.

Układy przełączające bardzo często współpracują z urządzeniami w których, dla wygody człowieka, informacja musi być przedstawiona w dziesiętnym systemie zapisu liczb (np. wskaźniki cyfrowe, drukarki wyników itp.). Bezpośrednio przejście od naturalnego kodu dwójkowego do sysemu dziesiętnego jest trudne tto zrealizowania technicznego, gdyż cyfry dziesiętne nie mają żadnego trwatego odpowiednika w ciągu symboli binarnych. Realizacje są znacznie prostsze, gdy każdej cyfrze dziesiętnej przyporządkuje się na stale określoną liczbę binarną, a więc gdy kodować się będzie nie całą liczbę dziesiętną, lecz każdą cyfrę oddzielnie. Takie kody są znane pod nazwą *kodów dwójkozoo-dziesięfnych.* Na przykład

$$37_{10} = 1001011_2$$

ale

$$37_{10} = 0011\ 0111_{2/10}$$

jeśli 3 i 7 zakoduje się 4-bitowym kodem, naturalnym. Zwiększa się wprawdzie potrzebna liczba bitów (np. do wyrażenia dowolnej trzycyfrowej liczby dziesiętnej trzeba 10 bitów kodu naturalnego lub 12 bitów kodu dwójkowo-dziesiętnego), lecz prostota przetwarzania całkowicie to wynagradza.

Dziesięć cyfr dziesiętnych można zakodować binarnie na wiele różnych sposobów i dlatego liczba możliwych kodów dwójkowo-dziesiętnych jest bardzo duża. Kilka częściej spotykanych kodów zestawiono w tabl. 1-2. Cechą szczególną większości z nich jest to, że stanowią odcinki czterobitowego kodu naturalnego utworzone przez usunięcie kolejnych sześciu liczb, różnych w różnych kodach. Cecha ta ułatwia realizację układów zliczania impulsów modulo 10, a nie modulo 16.

Tablica 1-2

Kody dwójko KO - dziesiętne

| • Kod | I | II | III | IV | V | vi |
|---|---|---|---|---|---|---|
| Bity | DCBA | DCBA | DCBA | DCBA | DCtiA | BDCBA |
| **Wagi** | 8421 | | 2421 | 2+21 | | |
| i) | 0000 | 0000 | 00(10 | 01 KU 1 | 01)11 ~ | 00000 |
| l | 0001 | 0001 | nooi | 01)01 | oino | 00001 |
| 2 | 0010 | 1)1)11) | HM 0 | no io | 0101 | 00011 |
| 3 | (I0U | 0011 | octu | nntt | ono | 00111 |
| 4 | oi no | 0100 | 0100 | 0100 | DIII | 01111 |
| 5 | 0101 | 11101 | 0101 | 1011 | 1000 | 11111 |
| 6 | 0110 | 0110 | 0110 | 1100 | 3001 | tiuo |
| 7 | 0111 | 0111 | oni | 1101 | 1010 | 11100 |
| H | 1000 | 100(1 | 1110 | 1110 | 1011 | 110011 |
| 9 | 1001 | 1111 | mi | 1111 | tioo | 10000 |

Kod I jest **początkowym** odcinkiem 4-bitowego kodu naturalnego, ma wagi S421 i dlatego jest znany jako *kod „8421"*[1]).

Kod II jest taki sam jak kod I w zakresie 0...S, natomiast [1]J jest zakodowane z opuszczeniem sześciu pozycji, czyli jak 15 w kodzie naturalnym. T;t zmiana powoduje, że kodu nie można opisać wagami poszczególnych pozycji.

Kod III ma przeskok o 6 pozycji między 7 **a** 8, co zmienia wagę pierwszej pozycji. Zwany jest *kodem ,,2421"*.

Następny kod IV (tzw. *kod Aikena)* dzięki przeskokowi między 4 a 5 ma w tym miejscu oś symetrii z negacją: każda liczba nad osią różni się od symetrycznej pod osiij negacją wszystkich argumentów. Ta właściwość, tzw. ***samouzupełnianie,*** jest bardzo cenna przy realizacji operacji arytmetycznych. Ma ją również następny kod V, utworzony z przesunięcia kodu naturalnego o 3 i dlatego zwany kodem *„plus* 3" *( + 3, excess 3).*

Kod dwójkowo-dziesiętny nie musi mieć tylko 4 bitów (zwanych *tetradą);* przez zwiększenie tej liczby można uzyskać pewne dodatkowe

[11] W literaturze anglosaskiej—j:iko BCD (Biimry Codcd **Dccirnali,**

zalety. Przykładem może być kod VI *(j/ulnisaiia,* ***pseudopierścieniowy),*** o specyficznym rozkładzie zer i jedynek. Cechą tego kodu jest łatwość zliczania i dekodowania. Inne kody dwójkowo-dziesiętne **zostaną** opisano niżej.

W tablicy 1-2 **przedstawiono** kody **o** zróżnicowanej budowie. Kody I, III **i IV** to **tzw.** ***kody pozycyjne,*** **w** których każdej pozycji odpowiada określona waga, *a* wyrażoną kodem liczbę **uzyskuje** się przez sumowanie wag, dla których cyfra jest jedynką. W kodzie I występują wagi **natutal-iie—kolejne** potęgi **podstawy** 2; kody III i IV mają wagi sztuczne. Pozostałe kotły w **tabl,** 1-2 to *kodv symboliczne,* **w** których relacja między cyfrą dziesiętną a ciągiem zer i jedynek jest umowna i nie podlega jednoznacznym regułom o wartości poszczególnych pozycji.

Wagi kodu pozycyjnego mogą mieć również wartości ujemne, np. kod symetryczny z negacją można zbudować **za** pomocą wag H, 4, — 2, — 1.

Opisane wyżej niektóre zasady budowania kodów dwójkowo-dziesiętnych mogą być w prosty sposób stosowane do tworzenia innych kodów z ograniczonym zakresem: dwójkowo-piątkowych, dwójkowo-szóstkowych itp. Kod VI daje zawszu parzystą liczbę kombinacji.

Z **kodów (j** *N* **kombinacjach** można tworzyć kotły o $2^Ł$ -N kombinacjach przez proste **dopisanie** *k* hitów i powtarzanie $A^r$. Na przykład z kodu dwójkowo-piątko w ego ,,421" przez dodanie jednego bitu można utworzyć kod dwójkowo-dziesiętny ($2^1$ -5) o wagach 5421.

### 1.3.3. KODY Z ZABEZPIECZENIAMI

Urządzenia automatyki cyfrowej pracują często w trudnych warunkach technicznych, pod działaniem silnych zakłóceń. Dla uniknięcia skutków zniekształcenia sygnału pod wpływem zakłóceń, zwłaszcza przy przekazywaniu informacji długimi przewodami lub drogą radiową, stosuje się specjalne zabezpieczenia, w tym również zabezpieczenia kodowe. Przykłady częściej stosowalnych kodów o zwiększonej odporności na zakłócenia zestawiono w tabl. 1-3. Są to kody dwójkowo-dziesiętne, lecz zasady ich tworzenia można przenieść na dowolne inne kody dwójkowe.

Najprostszą metodą zwiększenia odporności kodu jest *kontrola parzystości)* polegająca na dodaniu do w bitów informacyjnych jednego bitu. kontrolnego o takiej wartości, by liczba jedynek była parzysta

Ta blica 1-3

*K"dy* ***dtvójkowe*** *dziesiętne z* ***sabespttczemami***

| Kod | VII | VIII | IX | X | XI |
|---|---|---|---|---|---|
| Bity | *EDCBA* | *KIHGFEDCBA* | ***EDCBA*** | *GFEDCBA* | ***GFBDC BA*** |
| Wagi | **S42I0** | ')K76.i-kT210 | | 5043210 | S6420 10 |
| 0 | 0000(1 | O00O0000OI | 01)011 | 0100001 | 01)001 01 |
| 1 | 00011 | oooooonoio | 00101 | Ot 00010 | 00001 10 |
| 2 | 00101 | 0000000100 | 01001 | 0100100 | 0O010 01 |
| J | **00110** | 0000001001) | 111001 | 0101000 | 00010 10 |
| **4** | **01001** | oooooioooo | 00110 | 0110000 | O0100 Ol |
| **S** | OlOIf) | 0000100000 | 01010 | 1000001 | 00I00 10 |
| ▷ | **011QQ** | oooioooona | 10010 | 1000010 | 01000 01 |
| 7 | 01111 | 0010000000 | 01100 | 1000100 | 01000 10 |
| 8 | toooi | 0100000000 | **10100** | 1001000 | 1(1000 01 |
| 'J | 10O1O | I000000000 | ii 000 | 1010000 | 10000 10 |

(w innej wersji — nieparzysta). Układ odczytujący kod sprawdza liczbę jedynek, co umożliwia wykrycie *(deiekcją)* **przypadków** ze zniekształconą jedną, cyfrą, trzema i ogólnie — nieparzystą liczbą cyfr **w** kodzie. Rozwiązanie takie często stosuje się do kontroli urządzeń zewnętrznych, współpracujących z **urządzeniami** cyfrowymi (dziurkarki, czytniki itp.). Przykładem może być kod **VII** % **tabl.** 1-3; jest to kod „8421" *z* bitem kontroli parzystości.

Drugą ważną grupą kodów z zabezpieczeniem są kody ze ***stałym*** *indeksem,* tzn, ze stalą liczbą jedynek w zapisie liczby. Liczba możliwych rozmieszczeń /; jedynek na *n* pozycjach (liczba kombinacji) wynosi[1]

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \quad \text{dla} \quad 0 < k \leq n$$

Oczywiście: $\binom{n}{1} = n$; $\binom{n}{n} = 1$; $\binom{n}{k} = \binom{n}{n-k}$, więc np. $\binom{7}{5} = \binom{7}{2}$. Wartości $\binom{n}{k}$ dla $n \leq 10$ podaje tabela:

---

[1] Obok $\binom{n}{k}$ jest również stosowany zapis $C_n^k$.

| k \ n | 10 | 9 | S | 7 | 6 | 5 | 4 | 3 |
|---|---|---|---|---|---|---|---|---|
| **2** | 45 | 36 | 28 | 21 | 15 | 10 | 6 | 3 |
| **3** | 120 | K4 | 56 | 35 | 20 | 1(1 | 4 | |
| 4 | 210 | 126 | 71) | 35 | 15 | 5 | | |
| 5 | 252 | 126 | 56 | 21 | 6 | | | |

Kontrola znanej liczby jedynek przy dckodowaniu umożliwia detekcję wszystkich błędów nieparzystych (jak przy kontroli parzystości) oraz wielu błędów parzystych.

Wśród kodów **ze** stałym indeksem **szczególne** znaczenie ma kod $\binom{M}{1}$ — ,,/ 3 »", traktowany jako kod **pierwotny,** baza dla wszystkich innych kodów. Wynika to z faktu, że stanowi on najprostszy strukturalnie zapis informacji i jest dogodną postacią przejściową **między** sygnałami wielowartościowymi a bardziej ekonomicznymi postaciami dwójkowymi. Na przykład rozkazy sterowania ruchu jakiegoś narzędzia; „w lewo", „w prawo" „stop" - - najdogodniej jest zakodować wstępnie jako 100, 010 i 001 (trzy przyciski), a dopiero przy ewentualnym przetwarzaniu tej informacji zastosować krótsze wyrażenia (np. 10, *01,* 11). Ta rola kodu ,,1 z «" sprawia, że proces jego zamiany w inny kod jest często nazywany po prostu *kodowaniem,* a konwersja innego kodu w „1 **z** w" — *dekodowaniem.* Kod „ I z 10" przedstawiono wtabl. 1-3 (VIII). Niekiedy jest stosowana modyfikacja tego kodu, pozbawiona bitu *A.*

Innymi częściej stosowanymi kodami ze stałym indeksem są kody ,,2 z 5" należące do grupy dwójkowo-dziesiętnych $\binom{5}{2} = 10$. Jedną z możliwych postaci tego kodu przedstawiono jako IX w tabl. 1-3. Możliwe są takie zapisy kodu *„2* z 5", by stał się on quasi-pozycyjnym; np. stosując wagi 74210 lub 63210 można poprawnie przedstawić wszystkie pozycje z wyjątkiem 0 (suma **wag** 0 wynosi $11_{10}$ V? pierwszym przypadku **i** $3_{10}$ — w drugim).

Realizacja techniczna zliczania **w** kodzie ,,1 ⁄. »*' jest prosta, ale kosztowna ze względu na dużą rozwlekłość tego kodu. Pewne skrócenia są **możliwe** przy łączeniu kodów; zamiast ,,1 z w" można zastosować „1 Z K, " ora? „1 z H, " tak, by $\binom{n}{1} = \binom{a,}{1} + \binom{i}{1}$ czyli $n = i_1 i_2$. Na przykład; kod dwójkowo-dziesiętny można zbudować jako $\binom{2}{1}$ i $\binom{5}{1}$ lub $\binom{5}{1}$, i $\binom{2}{1}$). **Pierwszy** nosi **nazwę** *kodu* ***biqinarnego*** **(X), drugi** — kodu *guibinantego* (XI). Są to szczególne przypadki kodów „2 z 7".

Wszystkie kody ,,*k* z *n*" są **łatwe** do **dekodowania, zwłaszcza przy** małych wartościach *k.*

Kody z kontrolą parzystości i statym indeksem są najprostszymi kodami z zabezpieczeniem przed przekłamaniem. W przypadkach gdy prawdopodobieństwo zakłóceń jest duże, a przenoszona informacja szczególnie cenna — stosuje się znacznie bardziej rozbudowane kody z detekcją lub korekcją błędu. Zagadnienia te mają obszerną, specjalną literaturę.

### 1.3.4. KODY URZĄDZEŃ ZEWNĘTRZNYCH

Bardzo ważną czynnością układów cyfrowych jest pobieranie informacji wejściowych od urządzeń zewnętrznych (przełączniki, czytniki taśm, czujniki, przetworniki) i przekazywanie przetworzonej informacji innym urządzeniom zewnętrznym (wskaźniki, przekaźniki, drukarki, przetworniki itp.). Ta wymiana informacji powinna się odbywać za pomocą kodów możliwie najbardziej korzystnych zarówno dla układu

Tablica 1-4

*Kotły* ***tfaoójketBthd&iesictut*** »r~mfccł! *zewnętrznych*

| Kod | XII | **XIII** | XIV |
|---|---|---|---|
| **Bity** | *DCBA* | ***DCBA*** | *DCBA* |
| **0** | 0000 | 0010 | 0000 |
| 1 | 0001 | **0110** | **8001** |
| **2** | 0011 | **0111** | oou |
| **3** | **0010** | **0101** | **0010** |
| 4 | nno | **0100** | ono |
| **5** | 0111 | **1100** | 1110 |
| **6** | 0101 | **1101** | 1010 |
| 7 | 0100 | 1111 | lon |
| 9 | 11(10 | **1110** | **1001** |
| 9 | 1101 | **1010** | **1000** |

T ab Ii c a 1-5

*Kod **dfllekópis</icy** nr* i

| Nr | Bity JKLMN | Litery | Cyfry Znaki | Xr | Bity JK!,MX | Litery | Cyfry Zeaki |
|---|---|---|---|---|---|---|---|
| 1 | 11000 | A | — | 16 | 01101 | P | II |
| 2 | [0011 | B | > | 17 | 11101 | **n** | 1 |
| .1 | (11110 | C | *i* | *li* | 010111 | R | 4 |
| 4 | 100(0 | D | **kto tam** | (<> | (0100 | S | ' |
| 5 | 10000 | E | *3* | 20 | IX1001 | T | 5 |
| 6 | nuto | F | wolny | 21 | 1110(1 | L | 7 |
| 7 | 01011 | G | wolny | 22 | 01111 | V | = |
| **S** | 991')i | H | wolny | 2.1 | 11001 | W | 2 |
| 0 | 01100 | [ | 8 | 24 | io.ni | X | / |
| 10 | 11010 | J | dzwonek | 25 | 10101 | V | 6 |
| | | | | 2fi | [1)1)111 | Z i | + |
| 11 | I11I0 | K | ( | 27 | 00010 | Powrót wózka | |
| 12 | 01001 | L | ) | *2H* | **fHOOfl** | Zmiana wierazs | |
| 13 | 00111 | M | . | 2<) | **11111** | Litery | |
| 14 | 0011(J | X | , i | 30 | (1011 | Cyfry | |
| 1S | (10011 | 0 | *9* | 31 | 00100 | Udst^P | |

cyfrowego juk i urządzeń zewnętrznych. Ze względu na dużą liczbę możliwych urządzeń i specyficznych wymagań, kodów takich może być wiele; niektóre z nich opisano wyżej, kilka innych przedstawiono w tabli 1-4 i 1-5.

Wartości przesunięć liniowych i kątowych mrtgą być bezpośrednio zamieniane na kod dwójkowy za pomocą liniałów i tarcz kodowych (rozdz. 6). Dla uniknięcia dużych błędów przy odczytywaniu wartości przesunięć wymaga się, by wyrażenia kodowe dotyczące sąsiednich położeń różniły się jednym tylko bitem. Nie spełnia tego wymagania naturalny kod dwójkowy, gdyż np. przy przejściu od położenia 3 do położenia 4 (od 011 do 100) aż trzy bity zmieniają swe wartości. Kod zachowujący jednostkowe przejścia można zbudować wieloma sposobami. Najczęściej stosowany, ze względu na prostą strukturę i łatwość dekodowania, jest *kod Graya,* zwany też *refleksyjnym* ze względu na sposób konstruowania wyrażeń.

Kod Graya jednobitowy ma oczywiście postać jednoznaczną. Aby rozszerzyć go do dwóch bitów, należy jeszcze raz przepisać początkowe wyrażenie, ale w odwrotnej kolejności, a następnie początkowe wyrażenia opisać dodatkowym bitem 0, natomiast powtórzone —- bitem 1

| *A* | | *BA* | | *CBA* |
|---|---|---|---|---|
| 0 | 0 | 00 | 00 | 000 |
| 1 | 1 | 01 | 01 | 001 |
| | 1 | 11 | u | 011 |
| | 0 | 10 | 10 | 010 |
| | | | 10 | 110 |
| | | | 11 | 111 |
| | | | 01 | 101 |
| | | | 00 | 100 |

Rozszerzenia do 3,4, ... bitów robi się podobnie: powtarzając i dopisując dodatkowy bit. Dziesięć pierwszych wartości 4-bitowego kodu Graya (XII) podano w tabl. 1-4. Nie jest to dobra metoda opisywania dziesięciu cyfr, gdyż przejściu z 9 na 10 towarzyszy zmiana aż czterech bitów. Jako kod dwójkowo-dziesiętny jest raczej stosowany zmodyfikowany *kod Graya + 3* (XIII), czyli kod Graya przesunięty o 3 pozycje. Dla zachowania jednostkowego przejścia przy zmianie dziesiątek, setek itd. (tzn. z 9 na 10, z 19 na 20, z 199 na 200 itp.) przyjmuje się zasadę zmiany kolejności cyfr w nieparzystych dekadach, np.:

| | |
|---|---|
| 9—0010 | 1010 |
| 10—0110 | 1010 |
| ... | |
| 19—0110 | 0010 |
| 20—0111 | 0010 |

Czterobitowe wyrażenie kodowe ma więc wartość *N* lub 9—*N* w zależności od tego, czy wartość starszej dekady jest parzysta czy nieparzysta. Dla kodu XIII obliczenie 9—A^ jest bardzo proste, gdyż polega na zanegowaniu bitu *T*>. Taką samą właściwość ma kod XIV, zwany *kodem Wattsa.*

Do drukowania wyników przetwarzania cyfrowego w automatyce często stosuje się maszyny elektryczne do pisania i dalekopisy. W tabl. 1-5 przedstawiono standardowy kod dalekopisowy nr 2. W odróżnieniu od rozważanych wyżej, umożliwia on zapis nie tylko cyfr, lecz także liter i znaków (jest tzw. *kodem alfanumerycznym).* Większość kombinacji kodowych może mieć dwa różne znaczenia; o wyborze właściwego decyduje to, który z symboli „Litery", „Cyfry" występował ostatni w ciągu sygnałów. Zastępując dla uproszczenia wyrażenia kodowe ich numerami z przecinkiem, można np. nazwę

F I A T 12 5

zakodować

29,6,9,1,20,31,30,17,23,20

## 1.4. DZIAŁANIA ARYTMETYCZNE

### 1.4.1. DODAWANIE I ODEJMOWANIE DWÓJKOWE

W układach cyfrowych automatyki występuje często konieczność wykonywania operacji arytmetycznych na sygnałach, przedstawionych w postaci liczb dwójkowych lub dwójkowo-dziesiętnych. Zazwyczaj nie są to operacje bardzo złożone, a zakresy zmian zarówno argumentów jak i wyników działań można z góry wyznaczyć. Z tego też powodu, w przypadku występowania liczb mieszanych (całkowitych i ułamkowych) położenie przecinka jest znane i niezmienne, jeśli nie w całym procesie przetwarzania, to przynajmniej w poszczególnych fazach tego prouesu. Stały przecinek nie ma wpływu na sposób zapisywania liczb i wykonywania działań, a zatem może być całkowicie pomijany w układach przetwarzających i dopiero w końcowej fazie — drukowania lub przekazywania wyników — może znów być odtwarzany.

Reguły dodawania i odejmowania cyfr dwójkowych są bardzo proste:

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| dodajna | 0 | 0 | 1 | 1 | odjemna | 0 | 0 | 1 | 1 |
| dodajnik | 0 | 1 | 0 | 1 | odjemnik | 0 | 1 | 0 | 1 |
| suma | 0 | 1 | 1 | 0 | różnica | 0 | 1 | 1 | 0 |
| przeniesienie | 0 | 0 | 0 | 1 | pożyczka | 0 | 1 | 0 | 0 |

W przypadku liczb wielocyfrowych przeniesienie jest dodawane do cyfr bardziej znaczących, a pożyczka jest odejmowana od cyfr bardziej znaczących. Pewne komplikacje powstają wówczas, gdy nie wszystkie argumenty działań są dodatnie lub gdy odjemna jest mniejsza od odjemnika. Proste rozwiązanie tych problemów uzyskuje się przez pewną