

PODSTAWY TEORII UKŁADÓW CYFROWYCH

Jednostki informacji

Opracował: Andrzej Nowak

Bibliografia: **Urządzenia techniki komputerowej**, K. Wojtuszkiewicz
<http://pl.wikipedia.org/>

Informacja jest tworem czysto abstrakcyjnym. Nie istnieje materialnie. Nie można jej dotknąć, poczuć, zobaczyć. Informację da się natomiast wyrażać za pomocą symboli, znaków, kodu. Człowiek najczęściej używa słów, pisma, gestów do przekazywania informacji. Ta sama informacja (idea) może przybierać formę symboli, które, chociaż różne, oznaczają to samo pojęcie. Skoro do wyrażania informacji możemy używać różnych symboli (słów), to dlaczego nie zastosować do tego celu bitów?

Bit jest symbolem, który może przyjmować dwie różne postacie. Jeśli chcemy zapisać go na papierze, to stosujemy symbole pomocnicze **0** i **1**.

Technicznie bit realizowany jest za pomocą dwóch różnych sygnałów.

W technice cyfrowej określa się poziomy napięcie, które odpowiadają bezpośrednio dwóm postaciom bitu:

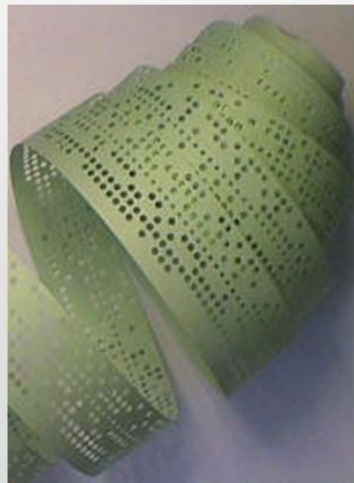
0,4 ... 0,8 V - stan **0** (oznaczany również **L** - **Low** - Niski)

2,0 ... 2,4 V - stan **1** (oznaczany również **H** - **High** - Wysoki)

Układy elektroniczne komputera reagują na te napięcia i w ten sposób przetwarzają bity. Zamiast napięć mogą to być również prądy o różnych natężeniach lub zwrotach, sygnały o dwóch rozróżnialnych częstotliwościach oraz wiele innych sposobów. Poprzestańmy więc na stwierdzeniu, iż bit to sygnał dwustanowy - jeden stan oznaczamy jako 0, a drugi jako 1. Każdy ze stanów bitu może przenosić jedną wiadomość w identyczny sposób, jak np. słówko "stół" przenosi wiadomość na temat rzeczy z płaskim blatem i zwykle czterema nogami.

Tyle tylko, że słowa naszego języka mają ustalone od stuleci znaczenia, bity natomiast możemy przystosowywać do dowolnych wiadomości w miarę potrzeb.

Jeden bit pozwoli w ten sposób przekazać jedną z dwóch wiadomości.



Taśma dziurkowana.
Tak kiedyś zapisywano dane.
Każda dziurka odpowiadała
bitowi o stanie **1**.

Brak dziurki
oznaczał stan **0**.

Przykład

Pojedyncze bity używane są do komunikacji z prostymi czujnikami, które reagują na określoną sytuację - np. gaz w chronionym pomieszczeniu, wzrost temperatury ponad wartość dopuszczalną, osiągnięcie przez ciecz w naczyniu określonego poziomu, otwarcie drzwi, przerwanie wiązki światła, itp.



Bity możemy łączyć w grupy i traktować je wspólnie jako symbol złożony. Przy takim podejściu otrzymujemy nieograniczone możliwości tworzenia słów binarnych i przypisywania im znaczeń.

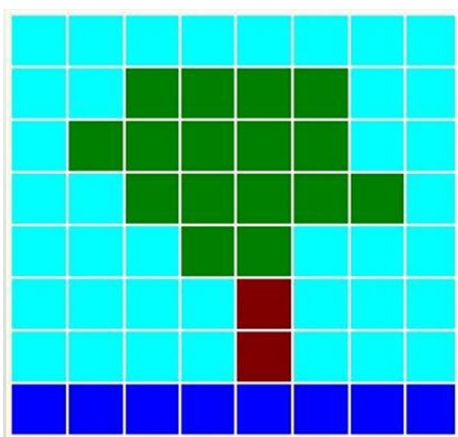
Grupa dwóch bitów oddaje nam do dyspozycji cztery różne symbole, które powstają z kombinacji stanów tworzących je bitów:

00 - symbol pierwszy
01 - symbol drugi
10 - symbol trzeci
11 - symbol czwarty





Każdemu symbolowi możemy przypisać informację o osobnym zdarzeniu. Grupa dwóch bitów może więc przekazać informację o czterech różnych zdarzeniach.

Przykład

Założmy, iż chcemy zakodować binarnie obrazek pokazany poniżej: Jest on złożony z różnokolorowych punktów, które nazywamy pikselami (z języka ang. picture element - element obrazu, punkt).



Od razu zauważamy, że punkty są tylko w czterech kolorach. Układamy tablicę kodową kolorów, w której każdemu kolorowi punktu przyporządkujemy jeden symbol dwubitowy:

	- 00
	- 01
	- 10
	- 11

Powiązaliśmy w ten sposób informację z reprezentującymi ją symbolami. Teraz wystarczy już tylko każdy piksel zastąpić symbolem dwubitowym. W tej postaci obrazek może być przechowywany w pamięci komputera, przesyłany przez sieci teleinformatyczne oraz przetwarzany.

00	00	00	00	00	00	00	00	00000000000000000000
00	00	11	11	11	11	00	00	00001111111110000
00	11	11	11	11	11	00	00	00111111111110000
00	00	11	11	11	11	11	00	00001111111111100
00	00	00	11	11	00	00	00	00000011110000000
00	00	00	00	10	00	00	00	00000000100000000
00	00	00	00	10	00	00	00	00000000100000000
01	01	01	01	01	01	01	01	010101010101010101

Zwróćmy uwagę na małą czytelność dla ludzi informacji zapisanej w systemie binarnym. Szczególnie, jeśli wszystkie bity zapiszemy w jednym ciągu:

0000000000000000000000001111111100000011111111100000001111111110000000111100000000000010000000101010101010101