

WIA2.

Tadeusz Puźniakowski

PJATK

Moje dane

- Tadeusz Puźniakowski
- <https://szuflandia.pjwstk.edu.pl:9000> – strona z materiałami
- [tadeusz.puzniakowski \(na\) pjatk.edu.pl](mailto:tadeusz.puzniakowski@pjak.edu.pl)

Uwagi

- Wykład jest przygotowaniem do ćwiczeń
- Zasady zaliczenia omówię na ćwiczeniach
- Sprawy organizacyjne

UWAGA

Poniższe kwestie przedstawię w praktyce na laboratoriach. Teraz czas na rzeczy ciekawsze (przeskakujemy na następny wykład)

Pozycyjne Systemy Liczbowe

Pozycyjny system liczbowy

Metoda zapisu liczby za pomocą ciągu cyfr. Każda pozycja w takim ciągu ma przyporządkowaną wagę. Najczęściej wagi są kolejnymi potęgami liczby b . Liczba b jest nazywana bazą systemu liczbowego.

- Liczbę $a_3a_2a_1a_0$ interpretujemy jako $a_0 \times b^0 + a_1 \times b^1 + a_2 \times b^2 + \dots$
- Dla nas naturalym pozycyjnym systemem liczbowym jest system dziesiętny (o podstawie 10).
- $a_0 \times 10^0 + a_1 \times 10^1 + a_2 \times 10^2 + \dots$
- Na przykład: $27 = 7 \times 1 + 2 \times 10 + 0 \times 100 + \dots$
- Komputery operują na systemie dwójkowym.

Oznaczenia Systemu Liczbowego – Ogólnie

- Jeśli na indeksie dolnym, w nawiasie, po liczbie jest jakaś inna liczba, to oznacza ona podstawę systemu liczbowego.
- Na przykład $12422_{(5)}$ oznacza liczbę w systemie pozycyjnym o podstawie 5
- Czasami będę stosował po prostu nawiasy, jeśli będzie to jasne w danym kontekście.

Konwencje Zapisu Liczb (stosowane na zajęciach)

- Konwencje literałów liczbowych zgodne z językiem C i Asemblera w wersji Intel
- System dziesiętny – po prostu
- System dwójkowy – mała litera b po prawej stronie (konwencja ASM)
- System ósemkowy – liczbę poprzedzamy cyfrą 0, na przykład 0771
- System szesnastkowy – liczbę poprzedzamy znakiem 0x, albo jak w Asemblerze ostatnim znakiem jest litera h, na przykład 16h, 0x16
- Oczywiście także notacja z liczbami w nawiasach.

Bit informacji

Wartości bitu

Bit przyjmuje wartość 0 albo 1, fałsz lub prawdę.

Trochę historii

Słowo bit zostało pierwszy raz użyte przez Claude'a Shannona w 1948 roku. Stwierdził on wtedy że ciągami zer i jedynek da się opisać dźwięk i obraz. Shannon stworzył także podwaliny pod informatykę wykazując że za pomocą układów logicznych działających na bitach da się wykonywać dowolne operacje arytmetyczne.

Przeliczanie na System Dziesiętny

Opis słowny

Ta operacja jest dość prosta, wystarczy mnożyć kolejne cyfry przez kolejne potęgi podstawy systemu liczbowego.

Przykład na żywo

- $62h = ? (10)$
- $1011(2) = ? (10)$

Przeliczanie Na Wybrany System Liczbowy

- Dzielimy przez podstawę liczbową oraz wyciągamy resztę.
- Resztę zapisujemy, a wynik używamy do kolejnej iteracji.
- Kończymy gdy dojdziemy do wyniku 0.
- Rezultat spisujemy od końca.
- Przykład na żywo: Przeliczenie liczby 101 zapisanej w systemie dziesiętnym na system dwójkowy

Przeliczanie Między Systemami Opartymi o Potęgę dwójki

- Najwygodniej to przeliczyć na system dwójkowy, a następnie na docelowy system liczbowy.
- Przykład na żywo: 0x0A1 na system ósemkowy
- Przykład na żywo: 0x0A1 na system binarny

Arytmetyka – dodawanie, odejmowanie

Jak to zrobić?

Operacje wykonujemy analogicznie do tego, jak to robimy dla systemu dziesiętnego.

Przykład na żywo

- $101101_2 + 11101011_2 = ?$
- $101101_2 - 1011_2 = ?$

Arytmetyka – mnożenie

Jak to zrobić?

Podobnie jak w systemie dziesiętnym, w słupku.

Przykład na żywo

- $101101_2 * 1101011_2 = ?$

Mała uwaga

System szesnastkowy i dwójkowy

Jeśli na koniec tego semestru, ktoś nie będzie sobie z tym radził, to nie ma szans u mnie na zaliczenie. Proszę mieć to na uwadze.

Sposoby zapisu liczb ze znakiem

- Zapis liczby ujemnej w pamięci komputera jest sensowny dla jakiejś ustalonej ilości bitów! Zwykle jest to 8, 16, 32, 64
- Kod ZM – znak-moduł
- Kod U1 – uzupełnienia do jedności.
- Kod U2 – uzupełnienie do 2

Najbardziej znaczący bit

UWAGA

W każdym sposobie zapisu liczby ujemnej, 1 na najbardziej znaczącej pozycji oznacza że jest to liczba ujemna.!

UWAGA

Zawsze należy określić na ilu bitach przechowujemy liczbę, bez tego zapis nie ma sensu.

Kod U1

- Bardzo prosty. Zakres $\langle -2^{n-1} + 1, 2^{n-1} - 1 \rangle$
- Liczby ujemne tworzy się na zasadzie odwrócenia każdego bitu liczby
- Dopisujemy tyle zer ile trzeba, aby wypełnić zadany rozmiar (na przykład 8, jeśli bajt)
- uwaga: czy istnieje liczba dająca się zapisać na 2 sposoby?
- Przykład na żywo:
- Przykład: Przelicz -67 na bajt w kodzie U1

Kod U2

- Najpopularniejszy w układach cyfrowych. Zakres $\langle -2^{n-1}, 2^{n-1} - 1 \rangle$
- Operacje dodawania i odejmowania tak jak przy zwykłych liczbach binarnych
- Liczbę ujemną otrzymuje się przez odjęcie danej liczby od dwukrotnej wagi najbardziej znaczącego bitu.
- Przykład: Przelicz -67 na bajt w kodzie U2 stosując zasadę „odwróć i dodaj 1”

Kod U2 – przykład

- Za pomocą kodu U2 wykonaj $13 - 5$ na liczbach ośmiobitowych
- Najpierw należy przerobić całe wyrażenie na $13 + (-5)$
- Teraz już wystarczy przeliczyć -5 na kod U2
- Wykonujemy dodawanie i otrzymujemy wynik. Zawsze należy ustalić na ilu bitach przeprowadzamy operację.

Kod U2 – przykład

- Jak widać, wystarczy nam dodawanie

Źródła

- intel.com
- wikipedia.org
- Moja wiedza