**1 Функції бінарної логіки**

*Функції бінарної логіки* - це функції, які приймають один або більше бінарних входів (тобто входи, які можуть мати лише два можливих значення: 0 або 1) і повертають одне бінарне значення відповідно до певного правила. Ці функції є основою для побудови логічних схем, які використовуються у цифрових пристроях, таких як комп'ютери, мікроконтролери та інші електронні пристрої.

Найпоширеніші функції бінарної логіки включають:

1. AND (логічне І): Повертає 1, якщо всі вхідні значення рівні 1, в іншому випадку повертає 0.

2. OR (логічне АБО): Повертає 1, якщо хоча б одне з вхідних значень дорівнює 1.

3. NOT (логічне НІ): Повертає зворотне значення свого вхідного біта. Якщо вхід 1, то вихід буде 0, і навпаки.

4. XOR (виключне АБО): Повертає 1, якщо кількість вхідних значень, що дорівнюють 1, є непарною.

5. NAND (виключне АБО): Це AND-операція, за якою стоїть NOT. Повертає 0, якщо всі вхідні значення рівні 1, і 1 у всіх інших випадках.

6. NOR (виключне АБО): Це OR-операція, за якою стоїть NOT. Повертає 0, якщо хоча б одне з вхідних значень дорівнює 1, і 1 у всіх інших випадках.

Ці функції можуть бути комбіновані для створення складних логічних операцій і схем, що відображають різноманітні логічні операції, які використовуються у цифровій електроніці та обчислювальних пристроях.

**2. Подання даних на рівні машин**

**2.1 Позиційні системи числення. Двійкова, вісімкова, шістнадцяткова системи числення. Беззнаковий код цілих чисел. Доповнювальний код цілих чисел. Основні арифметичні операції над цілими числами в беззнаковому та доповнювальному кодах.**

*Двійкова система числення* - це позиційна система числення, у якій кожне число представлене відповідною комбінацією двох цифр: 0 і 1. Кожен розряд у двійковому числі має певну вагу, яка збільшується від правого до лівого розряду. Наприклад, число "1010" у двійковій системі означає (1×23)+ (0×22) + (1×21) + (0×20), що дорівнює 10 в десятковій системі.

*Вісімкова система числення*, також відома як октальна система, є позиційною системою числення, в якій кожне число представлене відповідною комбінацією вісім цифр: 0, 1, 2, 3, 4, 5, 6, та 7. Кожен розряд у вісімковому числі має певну вагу, яка збільшується від правого до лівого розряду. Наприклад, число "52" у вісімковій системі означає (5×81) + (2×80), що дорівнює 42 в десятковій системі.

*Десяткова система числення* - це позиційна система числення, у якій кожне число представлене відповідною комбінацією десяти цифр: 0, 1, 2, 3, 4, 5, 6, 7, 8 та 9. Кожен розряд у десятковому числі має певну вагу, яка збільшується від правого до лівого розряду. Наприклад, число "365" у десятковій системі означає (3×102) + (6×101) + (5×100), що дорівнює 365.

*Шістнадцяткова система числення*, також відома як шістнадцяткова або шістнадцяткова система, є позиційною системою числення, у якій кожне число представлене відповідною комбінацією шістнадцяти цифр: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Кожен розряд у шістнадцятковому числі має певну вагу, яка збільшується від правого до лівого розряду. Цифри A, B, C, D, E та F представляють відповідно 10, 11, 12, 13, 14 та 15 в десятковій системі. Таким чином, число "1F" у шістнадцятковій системі означає (1×161) + (15×160), що дорівнює 31 в десятковій системі.

**2.2 Принципи зображення дійсних чисел в пам’яті програми у форматі з плаваючою комою. Переваги**

**та недоліки форматів чисел з плаваючою комою. Основні арифметичні операції над дійсними**

**числами у форматі з плаваючою комою та їхні проблеми.**

*Представлення з плаваючою комою* — це метод, який використовується в обчисленнях для представлення дійсних чисел (як раціональних, так і ірраціональних) із фіксованою кількістю цифр, що дозволяє представляти широкий діапазон значень із розумним рівнем точності. Ось основні принципи представлення дійсних чисел у форматі з плаваючою комою в програмній пам’яті:

1. Знаковий біт: крайній лівий біт зазвичай представляє знак числа. 0 для додатних чисел і 1 для від’ємних чисел.

2. Експонента: наступний набір бітів представляє експоненту. Цей показник зазвичай зміщений, щоб врахувати як додатні, так і від’ємні показники степеня. Зміщення дозволяє представити експоненту за допомогою форми доповнення до двох, де певне значення (зміщення) віднімається або додається до фактичного значення експоненти. Загальні зміщення включають 127 для одинарної точності (32-біт) і 1023 для подвійної точності (64-біт).

3. Дріб (значуще або мантиса): решта бітів представляють дробову частину числа. Цей дріб є нормалізованим, тобто початковий біт завжди дорівнює 1 (за винятком особливих випадків, таких як денормализовані числа або нуль). Оскільки початковий біт завжди дорівнює 1, він не зберігається явно, а зберігається лише дробова частина.

4. Особливі випадки: існують спеціальні випадки, які слід враховувати, наприклад позитивний і негативний нуль, денормализовані числа (де експонента має мінімальне значення), нескінченність (переповнення) і NaN (не число, використовується для представляють невизначені або помилкові умови).

5. Точність і діапазон: кількість бітів, призначених для експоненти та частки, визначає точність і діапазон чисел, які можна представити. Одинарна точність зазвичай використовує 32 біти (1 знаковий біт, 8 експонентних бітів і 23 дробові біти), тоді як подвійна точність зазвичай використовує 64 біти (1 знаковий біт, 11 експонентних бітів і 52 дробові біти).

6. Нормалізація: числа з плаваючою комою зазвичай зберігаються в нормалізованій формі, тобто початковий біт дробу завжди дорівнює 1. Це дозволяє виконувати ефективні арифметичні операції та спрощує порівняння між числами з плаваючою комою.

7. Помилка округлення: через кінцеву точність представлення числа з плаваючою комою можуть виникати помилки округлення під час виконання арифметичних операцій, особливо з числами, які неможливо точно представити у двійковому форматі з плаваючою комою.

8. Стандарт IEEE: стандарт IEEE 754 широко використовується для представлення чисел з плаваючою комою у двійкових форматах. Він визначає формати для одинарної точності (float), подвійної точності (double) і розширеної точності (long double), а також правила для арифметичних операцій, режимів округлення та обробки особливих випадків.

Формат чисел з плаваючою комою, такі як числа з плаваючою точкою (floats) у більшості мов програмування, мають свої переваги та недоліки.

Переваги:

1. Гнучкість у представленні дійсних чисел: Цей формат дозволяє представляти дуже великі або дуже малі числа, які можуть бути важко або неможливо представити з точністю за допомогою цілих чисел.

2. Широке використання у наукових обчисленнях та інженерії: Формат чисел з плаваючою комою дозволяє точно моделювати фізичні реальності, такі як дробові числа, результати вимірювань тощо.

3. Ефективність при зберіганні та обробці даних: У багатьох випадках числа з плаваючою комою займають менше місця у пам'яті порівняно з деякими іншими форматами, такими як рядки або десяткові числа.

Недоліки:

1. Невідомість точного значення: У деяких випадках виконання арифметичних операцій з числами з плаваючою комою може призвести до втрати точності або до виникнення помилок округлення.

2. Порівняння точності: Порівняння двох чисел з плаваючою комою на рівність може призвести до непередбачуваних результатів через обмежену точність зберігання.

3. Проблеми з відображенням: Деякі дрібові числа не можуть бути точно представлені у форматі чисел з плаваючою комою через їхню десяткову або бінарну природу.

Основні арифметичні операції над дійсними числами у форматі з плаваючою комою включають додавання, віднімання, множення та ділення. Проте ці операції можуть призводити до деяких проблем через обмежену точність та представлення чисел у цьому форматі.

1. Помилки округлення: У результаті обмеженості кількості бітів, використаних для представлення чисел з плаваючою комою, можуть виникати помилки округлення, особливо при великих розмірах або маленьких значеннях.

2. Втрата точності: Під час виконання послідовних арифметичних операцій може відбуватися втрата точності через накопичення помилок округлення та обмежену кількість значущих бітів.

3. Проблеми з порівнянням: Порівняння двох чисел з плаваючою комою на рівність може бути проблематичним через недостатню точність представлення.

4. Переповнення та денормалізація: Великі чи дуже малі числа можуть призвести до переповнення або денормалізації, що також може впливати на точність обчислень.

**3 Пристрої введення-виведення. Поняття шини комп'ютера**

**"Пристрої введення-виведення" (ПВВ) - це апаратні засоби, що дозволяють комп'ютеру обмінюватися даними з зовнішніми пристроями, такими як клавіатура, миша, монітор, принтер тощо.**

*Шина комп'ютера* - це спосіб, за допомогою якого ці пристрої зв'язуються з центральним процесором і іншими частинами системи. Вона виступає як шлях для передачі даних і сигналів між різними компонентами комп'ютера. Таким чином, шина відіграє ключову роль у забезпеченні спільної роботи всіх пристроїв в системі.

**4 Функціональна організація обчислювальних систем**

**4.1 Структура комп'ютера, класична архітектура фон Неймана, гарвардська архітектура.**

Структура комп'ютера включає в себе різні компоненти, які спільно працюють для виконання обчислень і обробки інформації. Класична архітектура фон Неймана та гарвардська архітектура - це два основних підходи до організації цих компонентів.

1. Класична архітектура фон Неймана: Ця архітектура базується на ідеях, висунутих відомим математиком та комп'ютерним вченим Джоном фон Нейманом. У цій архітектурі пам'ять і процесор розміщені у спільному блоку пам'яті. Інструкції і дані зберігаються у спільній пам'яті, а процесор зчитує їх послідовно для виконання операцій.

2. Гарвардська архітектура: У цій архітектурі пам'ять для інструкцій і пам'ять для даних зберігаються в окремих блоках. Це означає, що процесор може одночасно зчитувати інструкції і дані з різних джерел, що може підвищити швидкодію обчислень.

Ці дві архітектури мають свої переваги і недоліки, і використовуються у різних типах комп'ютерів залежно від конкретних потреб і вимог застосування.

**4.2 Ієрархічний принцип побудови пам'яті – регістрова, кеш, оперативна пам'ять, зовнішня пам'ять. CPU.**

*Ієрархічний принцип побудови пам'яті* - це організація пам'яті комп'ютера у вигляді декількох рівнів з різною швидкістю та розмірами для оптимізації доступу до даних. Основні рівні ієрархії пам'яті включають:

1. Регістри: Найшвидший тип пам'яті, який знаходиться безпосередньо в середині процесора. Використовується для тимчасового зберігання даних, необхідних для миттєвого доступу.

2. Кеш-пам'ять: Невеликий обсяг швидкодії пам'яті, розташований на чіпі процесора або недалеко від нього. Використовується для збереження копій часто використовуваних даних та інструкцій для прискорення доступу до них.

3. Оперативна пам'ять (ОЗП): Основний обсяг пам'яті, який використовується для зберігання даних і програм, що виконуються в даний момент. Це місце для тимчасового зберігання інформації, поки вона активно використовується.

4. Зовнішня пам'ять: Це великі обсяги пам'яті, які знаходяться поза межами процесора і використовуються для довгострокового зберігання даних, таких як жорсткі диски, SSD-накопичувачі, флеш-накопичувачі тощо.

Центральний процесор (CPU) використовує ці рівні пам'яті для виконання різних операцій і обробки даних. Ієрархічна організація пам'яті дозволяє підвищити швидкість доступу до даних та оптимізувати продуктивність системи.

**4.3 Периферійні пристрої.**

*Периферійні пристрої* - це пристрої, що приєднані до комп'ютера, але не входять до його основної системи. Вони використовуються для введення, виведення та обробки інформації. Сюди входять такі пристрої, як клавіатура, миша, монітор, принтер, сканер, флеш-накопичувачі, зовнішні жорсткі диски та інші. Периферійні пристрої розширюють функціональні можливості комп'ютера та забезпечують взаємодію користувача з системою.