

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
„ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”

МЕТОДИЧНІ ВКАЗІВКИ
“ЕЛЕМЕНТИ ІНФОРМАТИКИ”
до практичних занять з курсу “Інформатика”
для студентів спеціальності мікро- та наноелектроніка

Затверджено
редакційно-видавничою
радою університету,
протокол № ____ від ____

Харків НТУ “ХПІ” 2008

Методичні вказівки “Елементи інформатики” до самостійної роботи з курсу “Інформатика” для студентів спеціальності мікро- та наноелектроніка / Укл. В. І. Шкалето, Г. С. Хрипунов, О. П. Черних. – Харків: НТУ “ХП”, 2008. – 48 с.

Укладачі: В.І. Шкалето

Г.С.Хрипунов

О.П. Черних

Рецензент

Кафедра фізичного матеріалознавства для електроніки та геліоенергетики

Лабораторна робота № 3.1
ДВІЙКОВА І ШІСТНАДЦЯТКОВА СИСТЕМИ ЧИСЛЕННЯ.
ПЕРЕВЕДЕННЯ З ОДНІЄЇ СИСТЕМИ ЧИСЛЕННЯ В ІНШУ

Кількість годин на виконання – 4 години.

1. ОБЛАДНАННЯ

- 1.1. Комп'ютер.
- 1.2. Повчальні програми “ИнфоТ1.exe” і “Преобразование2_10_16.exe”.
- 1.3. Програма «Блокнот».

2. МЕТА РОБОТИ

- 2.1. Навчитися працювати з двійковими, вісімковими і шістнадцятковими числами.
- 2.2. Вивчити можливі способи представлення негативних двійкових чисел і особливості їх арифметики.
- 2.3. Вивчити основи вісімкової і шістнадцяткової арифметики.

3. ЗАГАЛЬНІ ВІДОМОСТІ

3.1. Інформація і інформатика

Під **інформацією** розумітимемо віддзеркалення матеріального світу у формі знаків і сигналів: креслень, фотографій, текстів і т.д. У широкому сенсі слова **інформація** – це будь-які сигнали, які можуть сприйматися деяким об'єктом.

Інформатика – наука про закони і методи накопичення, обробки і представлення інформації.

3.2. Основні відомості про уявлення і перетворення інформації в комп'ютері

Комп'ютер – універсальний засіб обробки інформації. Універсальність комп'ютера як засоби обробки інформації виявляється двояко: по відношенню до інформації і по відношенню до методів обробки.

3.2.1. Універсальність комп'ютера по відношенню до оброблюваної інформації

Ця універсальність пов'язана з можливістю її двійкового кодування.

Іншими словами, за допомогою комп'ютера можна обробляти все, що вдається представити у вигляді послідовності нулів і одиниць. Не дивлячись на таку «простоту» коду, в якому беруть участь всього дві цифри, з його допомогою можна представити інформацію різних типів – числову, символну, графічну і т.д. Деякі способи кодування показані на рис. 3.1а – 3.1в.

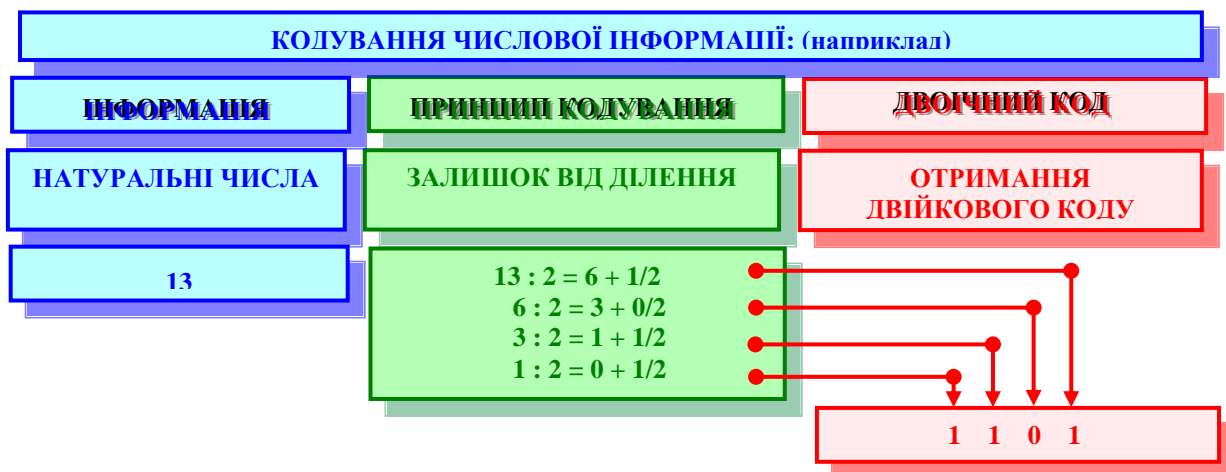


Рисунок 3.1а - Спосіб кодування числової інформації

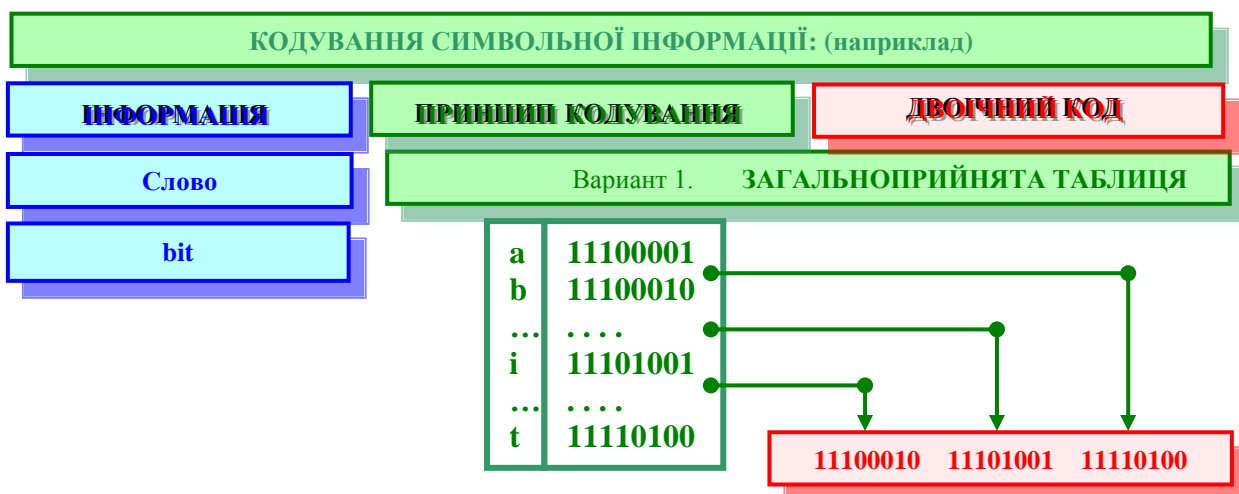


Рисунок 1.1 б - Спосіб кодування символної інформації

Використання двійкового коду дозволяє конструювати комп'ютер з великої кількості однотипних елементів, кожний з яких має всього два стійкі стани. Одне з них відповідає цифрі 0, інше – цифрі 1.

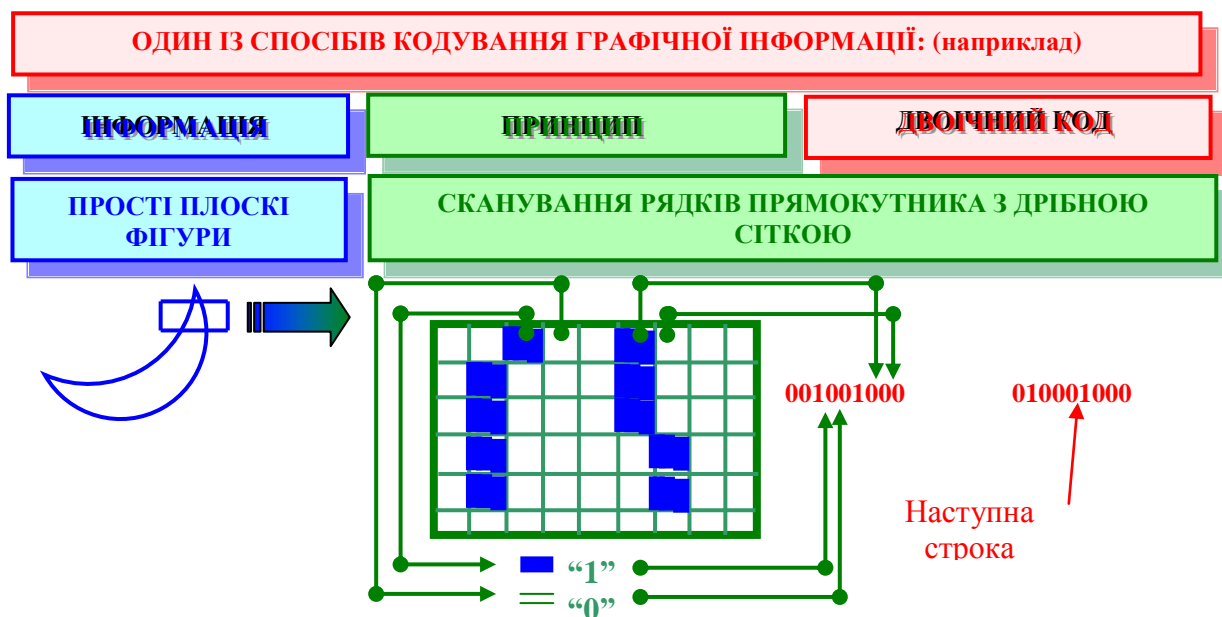


Рисунок 1.1 в - Один із способів кодування графічної інформації

3.2.2. Комп'ютер як універсальний засіб обробки інформації

Універсальність комп'ютера по відношенню до методів обробки інформації. Цю універсальність можна пояснити, використовуючи поняття виконавця. У загальному випадку під *виконавцем* розуміється пристрій (наприклад, комп'ютер), який може формально виконувати певний набір операцій, не «вникаючи» в їх сенс. В даному випадку йдеться про операції перетворення інформації. У зв'язку з двійковим кодуванням інформації її перетворення комп'ютером, як виконавцем, кінець кінцем, зводиться до операцій з двійковими розрядами. Технічно це означає, що комп'ютер повинен за деякими правилами змінювати стан тих пристроїв, які «зберігають» оброблювану інформацію.

Реально комп'ютер сприймає від користувача не елементарні команди, а інформацію про «більші» операції (у вигляді складання або віднімання чисел), а потім автоматично перетворює її на послідовність елементарних команд (це перетворення знову ж таки здійснюється спеціальною послідовністю елементарних команд). Така технологія уповільнює обробку, але значно полегшує «спілкування» користувача з комп'ютером.

3.3. Інформація

Інформатика - наука про закони і методи накопичення, обробки і представлення інформації. Інформація стає для виробництва таким же необхідним ресурсом як верстати, паливо, метал і т.д. Об'єми інформації, які доводиться переробляти суспільству, зростають. Звідси необхідність в розвитку засобів зв'язку і обчислювальної техніки для автоматизації праці працівників інформаційних професій.

Під інформацією розумітимемо віддзеркалення предметного світу у формі знаків і символів: креслень, фотографій, тексту і т.д. У широкому сенсі слова інформація це будь-які сигнали, які можуть сприйматися деяким об'єктом. Обмінюватися інформацією можуть не тільки люди, але і тварини, і механічні пристрої. Для передачі інформації використовуються сигнали: звукові, електричні, електромагнітні, механічні, світлові. Для представлення інформації використовують мови: жестів і міміки, малюнків і креслень, розмовні мови, мова математики, азбука Морзе, мова музики, мова двійкових чисел і т.д.

Особливе значення має **алфавітний спосіб представлення інформації** - пропозиції будуються з певного набору знаків:

- ❖ букви алфавіту;
- ❖ крапка і тире в азбуці Морзе;
- ❖ 1 і 0 в двійковому числі;
- ❖ цифри від 0 до 9 в десятковому числі.

При використанні алфавітного способу представлення інформації вона завжди передається дискретним сигналом – послідовністю окремих букв, цифр або знаків.

Дискретні сигнали надійніше передаються по каналах зв'язку. Інформацію, представлену алфавітним способом зручніше обробляти із застосуванням обчислювальної техніки.

У природі носієм інформації часто є безперервний сигнал, наприклад, музика або фотографія. Для автоматичної обробки безперервних сигналів

застосовують **дискретизацію**. В цьому випадку, наприклад, безперервний звуковий сигнал перетворюється на послідовність чисел. Через рівні, дуже маленькі проміжки часу вимірюється звуковий тиск. Передаючи числові значення звукового тиску, одержуємо дискретний сигнал, який тим краще, чим менше проміжки часу між вимірюваннями.

3.3.1. Кількість інформації

Кількість інформації не залежить від важливості її змісту, а визначається тільки кількістю знаків, передаваних по зв'язку дискретним сигналом.

Одиниця вимірювання кількості інформації називається біт - це інформація, яку несе відповідь на просте питання, коли можна відповісти тільки «та» чи ні».

Кількість можливих варіантів відповіді на питання називається невизначеністю. Інформація в 1 біт зменшує невизначеність в 2 рази.

Записавши послідовність відповідей у вигляді нулів і одиниць можна одержати двійковий код кожного числа. Кожен символ такого коду це 1 біт інформації.

У обчислювальній техніці інформація передається і приймається порціями не менше 8 біт (8 паралельних дротів). Тому вводиться одиниця вимірювання кількості інформації **1 байт = 8 біт**.

Існують більші одиниці вимірювання кількості інформації:

$$1 \text{ К (кілобайт)} = 1024 \text{ байт} = 2^{10} \text{ байт}$$

$$1 \text{ М (мегабайт)} = 1024 \text{ К} = 2^{10} \text{ К}$$

$$1 \text{ Г (гігобайт)} = 1024 \text{ М} = 2^{10} \text{ М}$$

3.4. Позиційні системи числення

Розглянемо такий принцип кодування чисел, коли код визначається використовуваною для кодування системою числення, тобто способом запису і найменування чисел. Загальноприйнято кодування із застосуванням *позиційних систем числення*.

У позиційних системах числення одна і та ж цифра має різне значення залежно від того, яке місце (позицію) вона займає в послідовності цифр, що

зображають число. Так, наприклад, в загальноприйнятій десятковій системі числення в послідовності цифр 777,77 одна і та ж цифра 7 представляє сотні, десятки, одиниці, десяті і соті долі одиниці:

$$777,7 = 7 \cdot 100 + 7 \cdot 10 + 7 \cdot 1 + 7 \cdot \frac{1}{10} + 7 \cdot \frac{1}{100} = 7 \cdot 10^2 + 7 \cdot 10^1 + 7 \cdot 10^0 + 7 \cdot 10^{-1} + 7 \cdot 10^{-2}$$

Тут число 10 є основою системи числення, визначуване кількістю цифр, що використовуються в ній (0, 1, 2., 9). Легко відмітити, що в записі, числа значення кожної цифри більше значення сусідньої цифри справа в число раз, рівне основі системи. Це є загальним правилом для чисел, що представляються в позиційних системах числення. Позиції цифр в записі числа називаються **розрядами числа**. Якщо в загальному випадку основа позиційної системи числення позначити через q , то будь-яке n -розрядне число можна представити в розгорненому вигляді таким чином:

$$a_{n-1}a_{n-2}\dots a_0a_{-1}a_{-2}\dots = a_{n-1}q^{n-1} + a_{n-2}q^{n-2} + \dots + a_0q^0 + a_{-1}q^{-1} + a_{-2}q^{-2} + \dots,$$

де як коефіцієнти $a_{n-1}, a_{n-2}, \dots, a_0, a_{-1}, a_{-2}$ $a_{n-1}, a_{n-2}, \dots, a_0, a_{-1}, a_{-2}$ можуть стояти будь-які із q цифр, використовуваних в системі числення.

3.4.1. Двійкова, вісімкова і шістнадцяткова системи числення

Для двійкової системи ($q = 2$) – **0,1**; для вісімкової системи ($q = 8$) – **0, 1, 2, 3, 4, 5, 6, 7**; для шістнадцяткової системи ($q = 16$) – **0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F** (A відповідає 10, B – 11, C – 12, D – 13, E – 14, F – 15). Нижче приведені записи десяткового числа $175 = 1 \cdot 10^2 + 7 \cdot 10^1 + 5 \cdot 10^0$ відповідно в шістнадцятковій, вісімковій і двійковій системах числення, а також розгорнене зображення в десятковій системі і звичайний десятковий запис:

$$1) AF_{16} = 10 \cdot 16^1 + 15 \cdot 16^0 = 175_{10};$$

$$2) 257_8 = 2 \cdot 8^2 + 5 \cdot 8^1 + 7 \cdot 8^0 = 175_{10};$$

$$3) 10101111_2 = 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 175_{10}.$$

Для передачі, обробки і зберігання інформації в сучасній техніці застосовують двійкове кодування: кожному знаку ставлять у відповідність число, і це число записують в двійковій системі числення. Числа в двійковій

системі числення записуються за допомогою двох цифр: 0 і 1. Чим ліво стоїть цифра в числі, тим більше її вага.

Слід зазначити, що кількість цифр в двійковому числі рівна кількості інформації, яка відповідає цьому числу.

3.4.2. Переведення чисел з однієї системи числення в іншу

Для переведення числа з двійкової системи в десяткову необхідно кожен цифру двійкового числа помножити на її вагу і результати скласти:

$$\begin{aligned} 10101111_{(2)} &= 1 \cdot 2^7 + 0 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = \\ &= 128 + 32 + 8 + 4 + 2 + 1 = 175_{(10)}. \end{aligned}$$

Для переведення числа з десяткової системи в двійкову використовують метод ділення на 2 із залишком. Десяткове число послідовно ділиться на 2, доки не вийде 0. Залишки від кожного ділення виписуються в зворотному порядку, що схематично показано на рис. 3.2. Аналогічно здійснюється переведення числа з десяткової системи числення у вісімкову (ділення на 8) і в шістнадцяткову (ділення на 16).

Перехід від запису чисел в двійковій системі до вісімкової або шістнадцяткової і назад проводиться за дуже простими правилами (рис. 3.3 і 3.4).

Для переходу від двійкової до вісімкової (або шістнадцяткової) системи необхідно рухаючись, справа наліво, розбити двійкове число груп по три (чотири) розряди відповідно, доповнюючи при необхідності ліву крайню групу нулями. Потім кожен групу з трьох (чотирьох) розрядів замінити відповідною вісімковою (шістнадцятковою) цифрою.

Для переведення вісімкового (шістнадцяткового) числа в двійкове досить замінити кожен цифру вказаного числа відповідним трьохрозрядним або чотирирозрядним двійковим числом. При цьому непотрібні нулі зліва відкидаються.

Запис чисел і кодів у вісімковій або шістнадцятковій системах містить менше цифр. Ця перевага дозволяє полегшити програмісту написання програм в кодах машинних команд.

Число:		залишок від ділення на 2	Число:		залишок від ділення на 2
43		1	22		0
21		1	11		1
10		0	5		1
5		1	2		0
2		0	1		1
1		1	0		
0					
		двоичное число: 101011			двоичное число: 10110

Рисунок. 3.2 - Переведення з десяткової системи числення в двійкову послідовним діленням на 2

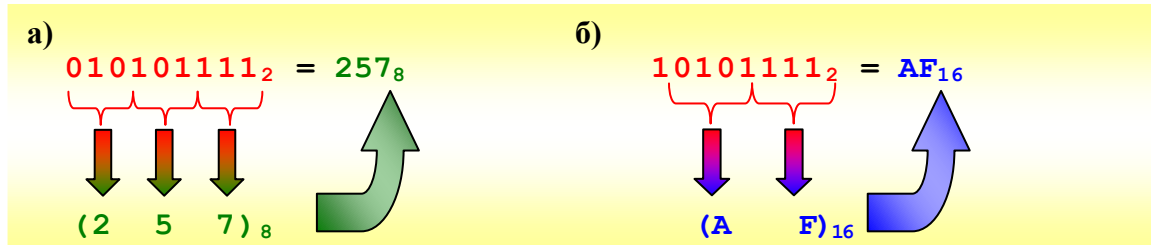


Рисунок. 3.3 - Переведення двійкового числа у вісімкове (а) і шістнадцятиричне (б)

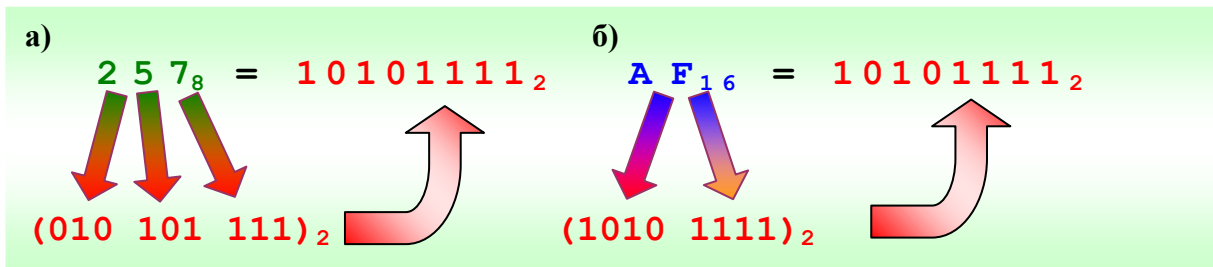


Рисунок. 3.4 - Перевод восьмеричного (а) і шістнадцятиричного (б) чисел в двоичні числа

3.4.3. Арифметичні операції з двійковими числами

З метою спрощення реалізації арифметичних операцій в комп'ютері для представлення двійкового числа використовуються прямий, зворотний і додатковий двійкові коди.

Прямий код використовується для представлення цілого двійкового числа у вигляді:

$$X_{\text{пр}} = \alpha a_{n-1} a_{n-2} \dots a_1 a_0,$$

де α – знаковий розряд (0 – для позитивного числа, 1 – для негативного числа); a_n – двійкові розряди числа (0 або 1).

Числа $X = -1101101$; $Y = 1101101$ в прямому коді мають вигляд:

$$X_{\text{пр}} = 11101101;$$

$$Y_{\text{пр}} = 01101101$$

Зворотний код використовується тільки для представлення негативних чисел і утворюється шляхом постановки в знаковий розряд одиниці і заміни у всіх інших розрядах числа одиниць нулями, а нулів

одинацями. Наприклад, число $X = -1101101$ в зворотному коді буде представлено у вигляді $X_{обр} = 10010010$. Зворотний код, якщо його розглядати як число, є доповненням модуля вихідного числа без знаку, поміщеного в розрядну сітку:

$$X + X_{обр} = 1101101 + 10010010 = 11111111.$$

Додатковий код числа отримують із зворотного шляхом збільшення одиниці до молодшого розряду. Наприклад, для числа $X = -01010101$

$$X_{обр} = 10101010; \quad X_{доп} = 10101011$$

Зворотний і додатковий коди позитивного числа співпадають з представленням самого числа. Для переходу від додаткового коду до модуля числа виконуються інверсія і додавання одиниці в тій же послідовності.

При складанні і відніманні чисел вони звичайно представляються залежно від типу арифметико-логічного пристрою (АЛП) процесора в зворотному або додатковому коді.

Якщо в пристрої числа, що запам'ятовує, представлені в іншому коді, то заздалегідь перед дією складання, алгебри, в АЛП здійснюється їх переведення в необхідний код. Після виконання даної операції результат, якщо він записується в оперативну пам'ять, знову переводиться в код, в якому представлені числа, що зберігаються в пристрої, що запам'ятовує.

Таблиця складання:

$0 + 0 = 0$
 $1 + 0 = 1$
 $1 + 1 = 10$ – перенесення в наступний розряд

Таблиця множення в двійковій системі:

$0 * 0 = 0$
 $1 * 0 = 0$
 $1 * 1 = 1$

Множення зводиться до операцій зсуву і складання.

Приклад:

$\begin{array}{r} 1011 \\ \underline{1101} \\ 1011 \\ 0000 \\ 1011 \\ \underline{1011} \\ 10001111 \end{array}$	} складання краще виконувати послідовно	} $\begin{array}{r} 1011 \\ + \\ \underline{101100} \\ 110111 \\ + \\ \underline{1011000} \\ 10001111 \end{array}$
---	--	--

При операціях з числами в додатковому коді самий старший розряд є носієм знаку числа: 0 - позитивне число, 1 – негативне. Перенесення із знакового розряду ігнорується.

Складання чисел в зворотних кодах виконується таким чином: зворотні коди разом із знаковими розрядами складаються як двійкові числа, і у разі виникнення одиниці перенесення із знакового розряду слід додати її до молодшого розряду числа. При цьому сума завжди виходить в зворотному коді.

Операція віднімання замінюється складанням:

Різниця $Z = X - Y$, де $X = 1101110$; $Y = 1001100$.

$Z = X_{обр} + (-Y_{обр}) = 01101110 + 10110011$.

Тоді різниця рівна:

$$\begin{array}{r}
 01101110 \\
 + \underline{10110011} \\
 100100001 \\
 \underline{1} \rightarrow \text{перенесення} \\
 00100010 \text{ із знакового розряду}
 \end{array}
 \qquad
 \begin{array}{l}
 Z_{обр} = 00100010 \\
 Z_{пр} = 00100010
 \end{array}$$

3.5. Представлення символної і графічної інформації

Символьна (алфавітно-цифрова) інформація в комп'ютері представляється за допомогою восьмирозрядних двійкових кодів. Повне число кодових комбінацій нулів і одиниць визначається завдовжки (розрядністю) коду і складає $2^8=256$. Кожному символу (цифрі, літері, знаку) ставиться у відповідність єдиний код з числа кодових комбінацій. За допомогою восьмирозрядного коду можна закодувати рядкові і прописні літери латинського алфавіту, літери російського алфавіту, цифри, розділові знаки, знаки математичних операцій і деякі спеціальні символи.

Існує багато різних систем кодування на базі восьмирозрядного двійкового коду. Стандартної прийняте як кодування символів ASCII (American Standard Code for Information Interchange – Американський стандартний код для обміну інформацією), приведене в таблиці 1.1а,б.

Тексти, складені з цих символів і введені в комп'ютер, зберігатимуться в його пам'яті у вигляді послідовності кодових комбінацій. Наприклад, тексту

$A=2.1+BC$ в пам'яті комп'ютера відповідатиме запис: **01000001 00111101 00110010 00101110 00110001 00101011 01000010 01000011.**

Про представлення графічної інформації в комп'ютері мовилося раніше, див. розділ 3.2.

Таблиця 1.1а - Регулярна таблиця ASCII (символьні коди 0 - 127).

000 (nul)	016 ► (dle)	032	048 0	064 @	080 P	096 `	112 p
001 ☉ (soh)	017 ◀ (dc1)	033 !	049 1	065 A	081 Q	097 a	113 q
002 ● (stx)	018 ↓ (dc2)	034 "	050 2	066 B	082 R	098 b	114 r
003 ♥ (etx)	019 !! (dc3)	035 #	051 3	067 C	083 S	099 c	115 s
004 ♦ (eot)	020 ¶ (dc4)	036 \$	052 4	068 D	084 T	100 d	116 t
005 ♣ (enq)	021 § (nak)	037 %	053 5	069 E	085 U	101 e	117 u
006 ♠ (ack)	022 = (syn)	038 &	054 6	070 F	086 V	102 f	118 v
007 ● (bel)	023 ↓ (etb)	039 '	055 7	071 G	087 W	103 g	119 w
008 ▣ (bs)	024 ↑ (can)	040 (056 8	072 H	088 X	104 h	120 x
009 (tab)	025 ↓ (em)	041)	057 9	073 I	089 Y	105 i	121 y
010 (lf)	026 → (eof)	042 *	058 :	074 J	090 Z	106 j	122 z
011 ♂ (vt)	027 ← (esc)	043 +	059 ;	075 K	091 [107 k	123 {
012 ♀ (np)	028 ⊥ (fs)	044 ,	060 <	076 L	092 \	108 l	124
013 (cr)	029 ↔ (gs)	045 -	061 =	077 M	093]	109 m	125 }
014 🎵 (so)	030 ▲ (rs)	046 .	062 >	078 N	094 ^	110 n	126 ~
015 ☼ (si)	031 ▼ (us)	047 /	063 ?	079 O	095 _	111 o	127 △

Таблиця 1.1б - Розширена Таблиця ASCII (символьні коди 128 - 255)

128 A	144 P	160 a	176 ☐	192 L	208 ll	224 p	240 Ě
129 Б	145 C	161 б	177 ☐	193 L	209 ll	225 c	241 ě
130 В	146 T	162 в	178 ☐	194 T	210 ll	226 t	242 €
131 Г	147 У	163 г	179	195 T	211 ll	227 y	243 €
132 Д	148 Ф	164 д	180	196 —	212 ll	228 ф	244 Ě
133 Е	149 X	165 е	181	197 †	213 ll	229 x	245 ě
134 Ж	150 Ц	166 ж	182	198 ll	214 ll	230 ц	246 Ě
135 З	151 Ч	167 з	183 ll	199 ll	215 ll	231 ч	247 ě
136 И	152 Ш	168 и	184 ll	200 ll	216 ll	232 ш	248 °
137 Й	153 Щ	169 й	185 ll	201 ll	217 ll	233 щ	249 •
138 К	154 Ъ	170 к	186 ll	202 ll	218 ll	234 ъ	250 •
139 Л	155 Ы	171 л	187 ll	203 ll	219 ll	235 ы	251 v
140 М	156 Ь	172 м	188 ll	204 ll	220 ll	236 ь	252 №
141 Н	157 Э	173 н	189 ll	205 =	221 ll	237 э	253 □
142 О	158 Ю	174 о	190 ll	206 ll	222 ll	238 ю	254 ■
143 П	159 Я	175 п	191 ll	207 ll	223 ll	239 я	255

4. ХІД РОБОТИ

- 4.1. Вивчити методичні вказівки (МВ) до лабораторної роботи.
- 4.2. Відповісти на питання в кінці МВ.

4.3. Включити комп'ютер.

4.4. Запустити на виконання повчальну програму «ИнфоТ1.exe». Ознайомитися з даними, запропонованими в програмі.

4.5. Запустити на виконання навчальну програму «Преобразование2_10_16.exe». Вивчити метод переведення чисел з однієї системи числення в іншу.

4.6. Виконати завдання, спільне для всіх варіантів.

4.7. Виконати завдання свого варіанту.

4.8. Оформити звіт.

5. ЗМІСТ ЗВІТУ

5.1. Мета роботи.

5.2. Результати виконання завдань лабораторної роботи.

5.3. Відповіді на контрольні питання.

6. ЗАВДАННЯ ДЛЯ ВИКОНАННЯ РОБОТИ

За допомогою програми «Блокнот» виконаєте наступні завдання:

6.1. Спільні завдання для всіх:

1. Представте дату свого народження в десятковому вигляді без роздільників в наступному порядку: день місяць рік (Наприклад: 03071986). Одержане число переведіть в двійкову, вісімкову і шістнадцяткову систему числення.

2. Запишіть своє прізвище у вигляді послідовності кодів ASCII (див. таблицю 1.1б) в десятковій, двійковій, вісімковій і шістнадцятковій формі.

3. Зобразіть у вигляді пікселів першу і третю літери свого прізвища (Чорні пікселі зображайте у вигляді літери «Ж» в знакомісті розміром 8*8 пікселів). Представте цю графічну інформацію в двійковому, вісімковому, шістнадцятковому і десятковому вигляді.

6.2. Варіанти завдань для виконання лабораторної роботи

6.2.1. Варіанти завдання 4

Переведіть наступні числа в двійкову, вісімкову і шістнадцяткову систему числення:

Варіант 1	130;	36;	125
Варіант 2	97;	83;	118
Варіант 2	89;	172;	222
Варіант 4	79;	222;	40
Варіант 5	141;	182;	180
Варіант 6	70;	178;	119
Варіант 7	205;	148;	234
Варіант 8	133;	238;	247
Варіант 9	190;	125;	29
Варіант 10	30;	206;	124
Варіант 11	60;	100;	29
Варіант 12	57;	112;	127
Варіант 13	190;	245;	152
Варіант 14	81;	228;	207
Варіант 15	39;	93;	54
Варіант 16	242;	145;	175
Варіант 17	238;	145;	186
Варіант 18	230;	172;	73
Варіант 19	130;	110;	38
Варіант 20	223;	171;	215

6.2.2. Варіанти завдання 5

Переведіть слідуєчі числа в двійкову систему числення, представте кожне з них в прямому, зворотному і додатковому кодах:

Варіант 1	-43;	-62;	-39
Варіант 2	-72;	-36;	-100
Варіант 2	-103;	-75;	-39
Варіант 4	-108;	-114;	-50
Варіант 5	-71;	-111;	-38
Варіант 6	-43;	-77;	-59
Варіант 7	-54;	-121;	-118
Варіант 8	-81;	-105;	-78
Варіант 9	-92;	-84;	-89
Варіант 10	-77;	-88;	-107
Варіант 11	-86;	-30;	-115
Варіант 12	-111;	-37;	-110
Варіант 13	-56;	-100;	-109
Варіант 14	-56;	-89;	-39
Варіант 15	-56;	-42;	-109
Варіант 16	-103;	-85;	-96
Варіант 17	-95;	-39;	-60
Варіант 18	-123;	-48;	-29
Варіант 19	-47;	-64;	-99
Варіант 20	-92;	-76;	-120

6.2.3. Варіанти завдання 6

Переведіть слідуєчі числа в двійкову, вісімкову і шістнадцяткову систему числення і виконайте в цих системах числення вказані дії:

Варіант 1	97 + -55;	117 - -51;	-15 * 10
Варіант 2	33 + 74;	38 - -42;	11 * -6
Варіант 2	61 + 110;	117 - -126;	13 * -14
Варіант 4	100 + 80;	40 - -102;	-4 * 7
Варіант 5	88 + -105;	113 - -120;	2 * 8
Варіант 6	-114 + -115;	-106 - -72;	6 * 6
Варіант 7	-89 + 53;	71 - -59;	7 * -17
Варіант 8	-46 + 73;	86 - -52;	-11 * -16
Варіант 9	90 + -70;	-50 - -28;	12 * 16
Варіант 10	104 + -104;	-63 - -91;	-7 * 3
Варіант 11	50 + -49;	-102 - 112;	-16 * -14
Варіант 12	84 + -115;	-72 - 72;	5 * -10
Варіант 13	-26 + -83;	-48 - -77;	7 * -8
Варіант 14	62 + 126;	-123 - -103;	-12 * -17
Варіант 15	81 + -31;	92 - -122;	-4 * -8
Варіант 16	-97 + -119;	-34 - 87;	-4 * -17
Варіант 17	-107 + 57;	93 - -47;	-16 * -17
Варіант 18	107 + 87;	122 - 67;	-8 * 3
Варіант 19	36 + -29;	116 - -43;	-6 * -17
Варіант 20	92 + -99;	-58 - -112;	-15 * -9

7. КОНТРОЛЬНІ ПИТАННЯ

7.1. Дайте визначення інформації, інформатиці і одиниці вимірювання інформації.

7.2. Опишіть один із способів кодування числової, символної і графічної інформації.

7.3. Опишіть алфавітний спосіб кодування інформації.

7.4. Що таке дискретизація? Для чого вона застосовується?

7.5. Опишіть принцип побудови позиційної системи числення.

7.6. Опишіть з погляду позиційної системи числення відмінності між двійковою, вісімковою і шістнадцятковою системами числення.

7.7. Як можна переходити із однієї системи числення в іншу?

7.8. Що таке метод послідовного ділення?

7.9. Для чого застосовуються прямий, зворотний і додатковий коди в двійковій арифметиці?

7.10. Опишіть одне з можливих застосувань таблиці ASCII кодів.

Лабораторна робота № 3.2

ПРЕДСТАВЛЕННЯ НЕГАТИВНИХ ДВІЙКОВИХ ЧИСЕЛ В КОМП'ЮТЕРІ. ДВІЙКОВА АРИФМЕТИКА

Кількість годинника на виконання – 2 години.

1. ОБЛАДНАННЯ

1.1. Комп'ютер.

1.2. Програма операційної системи Windows: «Блокнот».

2. МЕТА РОБОТИ

2.1. Навчитися працювати з негативними двійковими числами.

2.2. Вивчити різні представлення двійкових чисел.

2.3. Вивчити особливості арифметики двійкових чисел в різних представленнях.

3.1. Арифметичні операції з двійковими числами. Представлення

З метою спрощення реалізації арифметичних операцій в комп'ютері для представлення двійкового числа використовуються прямий, зворотний і додатковий двійкові коди.

Прямий код використовується для представлення цілого двійкового числа у вигляді

$$X_p = \alpha a_{n-1} a_{n-2} \dots a_1 a_0,$$

де α – знаковий розряд (0 – для позитивного числа, 1 – для негативного числа); a_n – двійкові розряди числа (0 або 1).

Числа $X = -1101101$; $Y = 1101101$ в прямому коді мають вигляд:

$$X_{пр} = 11101101;$$

$$Y_{пр} = 01101101.$$

Зворотний код використовується для представлення негативних чисел і утворюється шляхом постановки в знаковий розряд одиниці і заміни у всіх інших розрядах числа одиниць нулями, а нулів одиницями. Наприклад, число

$$X = -1101101$$

у зворотному коді буде представлено у вигляді

$$X_{обр} = 10010010.$$

Зворотний код, якщо його розглядати як число, є доповненням модуля вихідного числа без знаку, поміщеного в розрядну сітку:

$$X + X_{\text{обр}} = 1101101 + 10010010 = 1111111$$

Додатковий код числа отримують із зворотного шляхом збільшення одиниці до молодшого розряду. Наприклад, для числа $X = -01010101$

$$X_{\text{обр}} = 10101010; \quad X_{\text{дод}} = 10101011$$

Зворотний і додатковий коди позитивного числа співпадають з наданням самого числа.

Для переходу від додаткового коду до модуля числа виконуються інверсія і додавання одиниці в тій же послідовності.

При складанні і відніманні чисел вони звичайно представляються залежно від типу арифметико-логічного пристрою (АЛП) в зворотному або додатковому коді. Якщо в пристрої числа, що запам'ятовує, представлені в іншому коді, то заздалегідь перед дією складання, алгебри, в АЛП здійснюється їх переведення в необхідний код. Після виконання даної операції результат, якщо він записується в оперативну пам'ять, знову переводиться в код, в якому представлені числа, що зберігаються в пристрої, що запам'ятовує.

Таблиця складання	Таблиця множення
$0 + 0 = 0$	$0 * 0 = 0$
$1 + 0 = 1$	$1 * 0 = 0$
$1 + 1 = 10$ – перенесення в наступний розряд	$1 * 1 = 1$

3.1.1. Двійкове множення

Множення зводиться до операцій зсуву і складання (складання краще виконувати послідовно із зсувом).

Приклад:

Стандартне складання:	Складання із зсувом:
1011	1011
* 1101	(зсув)
1011	+ 101100 складання
0000	110111
1011	(зсув)
<u>1011</u>	+ 1011000 складання
10001111	10001111

3.1.2. Операції в прямому коді

Підрахунок суми $Z = X + Y$, де:

$$1. X = 1001101; \quad Y = 0011101.$$

$$\begin{array}{r} 01001101 \text{ (X)} \\ + \qquad \qquad \text{складання} \\ \underline{00011101} \text{ (Y)} \\ 01101010 \end{array}$$

$$2. X = -1011101; \quad Y = 0011101.$$

$$\begin{array}{r} 00011101 \text{ (Y)} \\ - \qquad \qquad \text{віднімання} \\ \underline{01011101} \text{ (-X)} \\ 11000000 \end{array}$$

$$3. X = 1010101; \quad Y = -1011101.$$

$$\begin{array}{r} 01010101 \text{ (X)} \\ - \qquad \qquad \text{віднімання} \\ \underline{01011101} \text{ (-Y)} \\ 10001000 \end{array}$$

3.1.3. Операції в зворотному коді

Складання чисел в зворотних кодах виконується таким чином: зворотні коди разом із знаковими розрядами складаються як двійкові числа, і у разі виникнення одиниці перенесення із знакового розряду слід додати її до молодшого розряду числа. При цьому сума завжди виходить в зворотному коді.

Сума $Z = X + Y$, де:

- $X = -1011101; X_{\text{обр}} = 10100010$
 $Y = 0011101 = Y_{\text{обр}}$
 $10100010 \text{ (X}_{\text{обр}})$
 $+ \underline{00011101} \text{ (Y}_{\text{обр}})$
 $10111111 \text{ (Z}_{\text{обр}}) = 11000000 \text{ (Z}_{\text{пр}})$
- $X = 1010101 = X_{\text{обр}}$
 $Y = -1011101; Y_{\text{обр}} = 10100010.$
 $01010101 \text{ (X}_{\text{обр}})$
 $+ \underline{10100010} \text{ (Y}_{\text{обр}})$
 $11110111 \text{ (Z}_{\text{обр}}) = 10001000 \text{ (Z}_{\text{пр}})$
- $X = 1110101 = X_{\text{обр}}$
 $Y = -1011101; Y_{\text{обр}} = 10100010.$
 $01110101 \text{ (X}_{\text{обр}})$
 $+ \underline{10100010} \text{ (Y}_{\text{обр}})$
 $00010111 \text{ (перенесення)}$
 $+ \underline{\quad\quad\quad 1}$
 $00011000 \text{ (Z}_{\text{обр}}) = 00011000 \text{ (Z}_{\text{пр}})$

3.1.4. Операції в додатковому коді

При операціях з числами в додатковому коді самий старший розряд є носієм знаку числа: 0 – позитивне число, 1 – негативне. Перенесення із знакового розряду ігнорується.

Сума $Z = X + Y$, де:

1) $X = -1011101$; $X_{\text{обр}} = 10100010$; $X_{\text{дол}} = 10100011$

$Y = 0011101 = Y_{\text{дол}}$

10100011 ($X_{\text{дол}}$)

+ 00011101 ($Y_{\text{дол}}$)

11000000 ($Z_{\text{дол}}$) = 11000000 ($Z_{\text{пр}}$)

2) $X = 1010101 = X_{\text{дол}}$;

$Y = -1011101$; $Y_{\text{обр}} = 10100010$; $Y_{\text{дол}} = 10100011$.

01010101 ($X_{\text{дол}}$)

+ 10100011 ($Y_{\text{дол}}$)

11111000 ($Z_{\text{дол}}$)

10000111 (інверсія)

+ $\underline{\quad\quad\quad} 1$

= 10001000 ($Z_{\text{пр}}$)

3) $X = 1110101 = X_{\text{обр}}$;

$Y = -1011101$; $Y_{\text{обр}} = 10100010$; $Y_{\text{дол}} = 10100011$.

01110101 ($X_{\text{дол}}$)

+ 10100011 ($Y_{\text{дол}}$)

00011000 (перенесення в знаковому розряді ігнорується)

00011000 ($Z_{\text{дол}}$) = 00011000 ($Z_{\text{пр}}$)

3.1.5. Віднімання в двійковій арифметиці

Операція віднімання $Z = X - Y$ замінюється складанням:

$$Z = X + (-Y)$$

Знайдемо $Z = X - Y$, де $X = 1101110$; $Y = 1001100$.

1. У зворотному коді:

$-Y_{\text{обр}} = 10110011$

01101110 ($X_{\text{обр}}$)

+ 10110011 ($Y_{\text{обр}}$)

00100001 (перенесення)

+ $\underline{\quad\quad\quad} 1$

00100010 ($Z_{\text{обр}}$) = 00100010 ($Z_{\text{пр}}$)

2. У додатковому коді:

$-Y_{\text{дол}} = 10110100$.

01101110 ($X_{\text{дол}}$)

+ 10110100 ($Y_{\text{дол}}$) (перенесення в знаковому розряді ігнорується)

00100010 ($Z_{\text{дол}}$) = 00100010 ($Z_{\text{пр}}$)

4. ХІД РОБОТИ

4.1. Вивчити методичні вказівки (МВ) до лабораторної роботи.

4.2. Відповісти на питання в кінці МВ.

4.3. Включити комп'ютер.

4.4. Запустити на виконання програму «Информатика_Ведение.exe».

Ознайомитися з арифметикою двійкових чисел.

4.5. Запустити на виконання програму операційної системи Windows: «Блокнот».

4.6. Виконати завдання з розділу 6 сьогодення МВ для свого варіанту.

4.7. Скласти звіт по лабораторній роботі.

5. ЗМІСТ ЗВІТУ

5.1. Мета роботи.

5.2. Результати виконання завдань лабораторної роботи.

5.3. Відповіді на контрольні питання.

6. ЗАВДАННЯ ДЛЯ ВИКОНАННЯ РОБОТИ

За допомогою програми «Блокнот» виконаєте наступні завдання:

6.1. Варіанти завдань для виконання лабораторної роботи

6.1.1. Варіанти завдання 1

Виконайте множення двійкових чисел і перевірте результат, перейшовши в десяткову систему числення.

Варіант 1.	1101 * 1001	Варіант 11.	1001 * 1000
Варіант 2.	1000 * 0111	Варіант 12.	1001 * 1010
Варіант 3.	1010 * 1000	Варіант 13.	1100 * 1101
Варіант 4.	1110 * 0101	Варіант 14.	1001 * 1100
Варіант 5.	0101 * 1100	Варіант 15.	0111 * 1000
Варіант 6.	1110 * 1100	Варіант 16.	1010 * 1001
Варіант 7.	1101 * 1000	Варіант 17.	0110 * 1101
Варіант 8.	0110 * 1011	Варіант 18.	0110 * 1110
Варіант 9.	1110 * 0111	Варіант 19.	1101 * 1000
Варіант 10.	1000 * 1110	Варіант 20.	1011 * 1101

6.1.2. Варіанти завдання 2

Виконайте складання двійкових чисел в прямому коді і перевірте результат, перейшовши в десяткову систему числення.

Підрахунок суми $Z = X + Y$, де:

Варіант 1	1) X = 00111111; Y = 00101111	2) X = - 01111000; Y = 01101011	3) X = 01000101; Y = - 01010010
Варіант 2	1) X = 00101111; Y = 00111000	2) X = - 01010110; Y = 01001011	3) X = 01001100; Y = - 10000000
Варіант 2	1) X = 00010100; Y = 00100100	2) X = - 01111000; Y = 00101101	3) X = 00110001; Y = - 00111011
Варіант 4	1) X = 01011011; Y = 00010010	2) X = - 01110110; Y = 01110000	3) X = 00011101; Y = - 00110111
Варіант 5	1) X = 00110101; Y = 00010010	2) X = - 01101100; Y = 01100001	3) X = 00110110; Y = - 01101010

Варіант 6	1) X = 01001100; Y = 00010010	2) X = - 01101110; Y = 01010100	3) X = 00110110; Y = - 01110011
Варіант 7	1) X = 00110101; Y = 00101001	2) X = - 01101010; Y = 01101001	3) X = 01011001; Y = - 01100101
Варіант 8	1) X = 00111110; Y = 00011101	2) X = - 00110101; Y = 00101000	3) X = 00011111; Y = - 01001011
Варіант 9	1) X = 01000110; Y = 00010100	2) X = - 01100001; Y = 01001101	3) X = 00110001; Y = - 01101000
Варіант 10	1) X = 00111011; Y = 00010111	2) X = - 01110001; Y = 01001100	3) X = 00101100; Y = - 01000111
Варіант 11	1) X = 00011100; Y = 00111111	2) X = - 01001101; Y = 00110010	3) X = 01000011; Y = - 01000100
Варіант 12	1) X = 01010010; Y = 00100000	2) X = - 01100101; Y = 00111110	3) X = 00100000; Y = - 01100110
Варіант 13	1) X = 01011011; Y = 00001111	2) X = - 01111110; Y = 01001001	3) X = 00110001; Y = - 01000110
Варіант 14	1) X = 00100111; Y = 00111001	2) X = - 01011010; Y = 01011000	3) X = 00011101; Y = - 00110001
Варіант 15	1) X = 00100101; Y = 00110111	2) X = - 01100000; Y = 00110100	3) X = 00110100; Y = - 01101111
Варіант 16	1) X = 01001100; Y = 00100111	2) X = - 01110100; Y = 00110000	3) X = 00101101; Y = - 01010110
Варіант 17	1) X = 00011011; Y = 00010101	2) X = - 01101110; Y = 00100101	3) X = 00011101; Y = - 01011101
Варіант 18	1) X = 01100010; Y = 00010110	2) X = - 01110010; Y = 00110011	3) X = 01100000; Y = - 01100111
Варіант 19	1) X = 00010001; Y = 00111111	2) X = - 01111101; Y = 01110011	3) X = 00111010; Y = - 01010100
Варіант 20	1) X = 00110001; Y = 00101010	2) X = - 01101101; Y = 01000000	3) X = 00011010; Y = - 01101110

6.1.3. Варіанти завдання 3

Виконайте складання двійкових чисел в зворотному коді і перевірте результат, перейшовши в десяткову систему числення.

Підрахунок суми $Z = X + Y$, де:

Варіант 1	1) X = - 01010100; Y = 01011100	2) X = 01100110; Y = - 00101101	3) X = 01100100; Y = - 01101100
Варіант 2	1) X = - 00111001; Y = 01100110	2) X = 01110110; Y = - 01010001	3) X = 01101111; Y = - 01110111
Варіант 2	1) X = - 00011100; Y = 00100000	2) X = 01000101; Y = - 00111101	3) X = 00011011; Y = - 01010011
Варіант 4	1) X = - 00100111; Y = 01001000	2) X = 01100111; Y = - 00101011	3) X = 01010111; Y = - 01101100
Варіант 5	1) X = - 00100000; Y = 00101001	2) X = 01100001; Y = - 01010111	3) X = 01001100; Y = - 01111100
Варіант 6	1) X = - 01001111; Y = 01101011	2) X = 01101111; Y = - 00100100	3) X = 01101101; Y = - 01110010
Варіант 7	1) X = - 00011001; Y = 00110000	2) X = 01100000; Y = - 00101101	3) X = 00110000; Y = - 00110111
Варіант 8	1) X = - 00101110; Y = 01011111	2) X = 01111011; Y = - 01010001	3) X = 00100100; Y = - 10000000
Варіант 9	1) X = - 01101101; Y = 01101110	2) X = 01010011; Y = - 01010001	3) X = 00101101; Y = - 01100001
Варіант 10	1) X = - 01100110; Y = 01111100	2) X = 01101001; Y = - 00101110	3) X = 01010101; Y = - 01100011
Варіант 11	1) X = - 00001111; Y = 01010001	2) X = 01101010; Y = - 00100011	3) X = 01001101; Y = - 01110100
Варіант 12	1) X = - 00111100; Y = 00111101	2) X = 01011001; Y = - 00111100	3) X = 01001011; Y = - 01010111
Варіант 13	1) X = - 00110111; Y = 01110000	2) X = 01110111; Y = - 01010010	3) X = 00011111; Y = - 01101000
Варіант 14	1) X = - 01001101; Y = 01010000	2) X = 01000011; Y = - 00111010	3) X = 01100011; Y = - 01101010
Варіант 15	1) X = - 00110111; Y = 01110100	2) X = 01110111; Y = - 00111101	3) X = 00100100; Y = - 01101111
Варіант 16	1) X = - 00110001; Y = 01000000	2) X = 01001011; Y = - 01001010	3) X = 00100100; Y = - 01010100
Варіант 17	1) X = - 00010110; Y = 01001001	2) X = 01100010; Y = - 01001011	3) X = 01010101; Y = - 01111110
Варіант 18	1) X = - 01010000; Y = 01110100	2) X = 01101000; Y = - 00100111	3) X = 00111110; Y = - 01111010
Варіант 19	1) X = - 01101000; Y = 01111110	2) X = 01111101; Y = - 00110010	3) X = 00101100; Y = - 01100111
Варіант 20	1) X = - 00111101; Y = 01110110	2) X = 01100101; Y = - 01001000	3) X = 01001011; Y = - 01111010

6.1.4. Варіанти завдання 4

Виконайте складання двійкових чисел в додатковому коді і перевірте результат, перейшовши в десяткову систему числення.

Підрахунок суми $Z = X + Y$, де:

Варіант 1	1) X = - 00101011; Y = 01010111	2) X = 01101010; Y = - 01000001	3) X = 00011111; Y = - 01101111
Варіант 2	1) X = - 01100011; Y = 01101111	2) X = 01001100; Y = - 00101100	3) X = 00110111; Y = - 01001100
Варіант 2	1) X = - 00111000; Y = 01001111	2) X = 01110001; Y = - 01100000	3) X = 01001111; Y = - 01010011
Варіант 4	1) X = - 01000011; Y = 01101011	2) X = 01111101; Y = - 01000111	3) X = 01101000; Y = - 01111101
Варіант 5	1) X = - 00101110; Y = 01010110	2) X = 01011000; Y = - 00110011	3) X = 00101011; Y = - 01010101
Варіант 6	1) X = - 01110100; Y = 01111101	2) X = 01001110; Y = - 00110010	3) X = 01010101; Y = - 01011110
Варіант 7	1) X = - 00100100; Y = 00111010	2) X = 01010100; Y = - 00101011	3) X = 00011110; Y = - 01011110

Варіант 8	1) X = - 01011001; Y = 01100100	2) X = 01101001; Y = - 01000111	3) X = 00111011; Y = - 01011100
Варіант 9	1) X = - 00011111; Y = 00111010	2) X = 01000110; Y = - 00101100	3) X = 00011100; Y = - 01010101
Варіант 10	1) X = - 00011001; Y = 01011011	2) X = 00111101; Y = - 00111100	3) X = 00110000; Y = - 00110101
Варіант 11	1) X = - 00001111; Y = 00010010	2) X = 01001001; Y = - 00111110	3) X = 01001001; Y = - 01101011
Варіант 12	1) X = - 00010101; Y = 00011001	2) X = 01110101; Y = - 01000000	3) X = 00111001; Y = - 01010000
Варіант 13	1) X = - 00011010; Y = 00101010	2) X = 01100100; Y = - 01001011	3) X = 00100100; Y = - 01000110
Варіант 14	1) X = - 00011111; Y = 01010011	2) X = 01001100; Y = - 00111011	3) X = 01001000; Y = - 01110000
Варіант 15	1) X = - 00101010; Y = 00110110	2) X = 01110101; Y = - 01001100	3) X = 00101100; Y = - 00111001
Варіант 16	1) X = - 00010001; Y = 01100001	2) X = 01001111; Y = - 01000101	3) X = 00110001; Y = - 01111000
Варіант 17	1) X = - 01011110; Y = 01110100	2) X = 01111001; Y = - 01000000	3) X = 00011011; Y = - 00100011
Варіант 18	1) X = - 00100000; Y = 01001000	2) X = 01111010; Y = - 00101100	3) X = 01011000; Y = - 01011010
Варіант 19	1) X = - 01111100; Y = 01111101	2) X = 01110101; Y = - 01011011	3) X = 00011101; Y = - 00110111
Варіант 20	1) X = - 00011101; Y = 01111101	2) X = 01111001; Y = - 01001111	3) X = 00111101; Y = - 01101010

6.1.5. Варіанти завдання 5

Виконайте віднімання двійкових чисел в прямому і додатковому кодах і перевірте результат, перейшовши в десяткову систему числення.

Підрахунок різниці $Z = X - Y$, де

Варіант 1	1) X = 00110100; Y = 01000001	2) X = 01001001; Y = 00111101
Варіант 2	1) X = 00111111; Y = 01010111	2) X = 01011010; Y = 00111100
Варіант 2	1) X = 00111001; Y = 01101001	2) X = 01110000; Y = 01100000
Варіант 4	1) X = 01001011; Y = 01110101	2) X = 00110001; Y = 00110000
Варіант 5	1) X = 01001010; Y = 01100011	2) X = 01110100; Y = 01100010
Варіант 6	1) X = 00100000; Y = 01110011	2) X = 01111101; Y = 01011001
Варіант 7	1) X = 01011111; Y = 01110011	2) X = 01110100; Y = 01000101
Варіант 8	1) X = 01001110; Y = 01100100	2) X = 01100110; Y = 01100001
Варіант 9	1) X = 00101001; Y = 01000010	2) X = 01001111; Y = 00110110
Варіант 10	1) X = 00111000; Y = 01010100	2) X = 00110101; Y = 00100111
Варіант 11	1) X = 00100111; Y = 01010000	2) X = 01001100; Y = 00111101
Варіант 12	1) X = 00110101; Y = 01000010	2) X = 01000101; Y = 00101100
Варіант 13	1) X = 01011001; Y = 01100000	2) X = 01110111; Y = 01001010
Варіант 14	1) X = 00011111; Y = 01010010	2) X = 01110111; Y = 00101100
Варіант 15	1) X = 00101011; Y = 00101111	2) X = 01010011; Y = 01001111
Варіант 16	1) X = 00011011; Y = 00101111	2) X = 01111111; Y = 01111010
Варіант 17	1) X = 01001011; Y = 01011010	2) X = 01111111; Y = 01111001
Варіант 18	1) X = 01001110; Y = 01011001	2) X = 01110110; Y = 01110110
Варіант 19	1) X = 00110010; Y = 01011101	2) X = 01001100; Y = 00110000
Варіант 20	1) X = 01000111; Y = 01101001	2) X = 01110000; Y = 01011111

7. КОНТРОЛЬНІ ПИТАННЯ

- 7.1. Від чого залежить вид представлення двійкових чисел в комп'ютері?
- 7.2. Які представлення можуть використовуватися для двійкових чисел?
- 7.3. Опишіть позиційний спосіб представлення чисел.
- 7.4. Опишіть прямий код для представлення цілого двійкового числа.
- 7.5. Для чого використовується зворотний і додатковий коди?
- 7.6. Як з прямого коду отримати двійкове число в зворотному коді?
- 7.7. Як з прямого коду отримати двійкове число в додатковому коді?
- 7.8. Як з представлення цілого двійкового числа в зворотному коді отримати двійкове число прямому коді?
- 7.9. Як з додаткового коду отримати двійкове число в прямому коді?

Лабораторна робота № 3.3

ВИКОНАННЯ КОМАНД МІКРОПРОЦЕСОРОМ

Кількість годин на виконання – 4 години.

1. ОБЛАДНАННЯ

- 1.1. Комп'ютер.
- 1.2. Моделююча програма **prCP.exe**.

2. МЕТА РОБОТИ

- 2.1. Вивчити склад і основні принципи роботи окремих вузлів комп'ютера.
- 2.2. Вивчити склад і основні принципи роботи центрального процесора.

3. ЗАГАЛЬНІ ВІДОМОСТІ

3.1. Основні блоки персонального комп'ютера

3.1.1. Характеристики і класифікація комп'ютерів

В даний час всі комп'ютери можна розділити на 3 великих класу – великі ЕОМ, персональні комп'ютери і «ноутбуки». Ці класи відрізняються один від одного по декількох основних параметрах, що характеризують роботу комп'ютера.

До основних таких параметрів відносяться:

- ❖ швидкодія, що характеризує швидкість обробки інформації;
- ❖ об'єм пам'яті, що характеризує кількість інформації, яку здатний обробити комп'ютер протягом достатньо малого проміжку часу;
- ❖ геометричні розміри і маса самого комп'ютера.

Наприклад, «ноутбуки» – переносні комп'ютери, що уміщаються в портфель.

Великі ЕОМ (розміром з кімнату) – обробляють достатньо великі масиви інформації (наприклад, комп'ютер, обслуговуючий «Пентагон» в США).

3.1.1.1. Складові персональних ЕОМ (ПЕВМ)

Складові персональних ЕОМ (ПЕВМ) можна розглядати з трьох різних точок зору: по тому, де вони розміщуються, як вони функціонують, і як вони взаємодіють один з одним.

ПЕВМ, що фізично становлять, можна розділити на компоненти системного блоку і компоненти блоку розширення. Вся основна плата, що входить до складу будь-якої моделі IBM/PC, і блок живлення розміщуються у великому блоці, що одержав назву системного. Системний блок включає всі необхідні компоненти, що дозволяють комп'ютеру працювати без яких-небудь доповнень. Тут знаходяться:

- ❖ мікропроцесор
- ❖ перші 64 До пам'яті;
- ❖ «вбудовані» програми, записані в мікросхемах пристрою, що постійного запам'ятовує (ПЗП).

Велика частина компонентів, описаних в наступних вікнах, також знаходиться в системному блоці.

3.1.1.2. Системний блок

Системний блок поміщений в корпус, в якому також розміщені дисководи і інші пристрої зовнішньої пам'яті.

У корпусі передбачено місце, призначене для розміщення блоків розширень, а також є вільні роз'єми, призначені для підключення додаткового обладнання, яке може бути запроваджене до складу комп'ютера (PC). Блоки розширення (інтерфейси) або карти, як їх іноді називають, можуть використовуватися для обслуговування пристроїв, що підключаються до PC. Вони можуть використовуватися для двох основних цілей: для збільшення об'єму пам'яті і підключення додаткових пристроїв. Якщо обладнання уміщається на одній платі, то його можна розмістити усередині корпусу PC. Якщо ж воно не поміщається в корпус, наприклад, у випадку з дисплеєм, то всередині розміщується тільки плата управління, що з'єднується з обладнанням за допомогою кабелю, який можна підключити до роз'єму, що встановлюється в задній стінці корпусу. Кожному роз'єму розширення відповідає спеціальний отвір в задній стінці корпусу, закритий заглушкою, якщо воно не використовується.

Системний блок розроблений фірмою IBM, а блоки розширення можуть розробляти всі охочі фірми, за умови що вони дотримуватимуть основні правила, що стосуються розмірів, електричних параметрів з'єднань теплового режиму і так далі.

3.1.1.3. Тенденції розвитку комп'ютерів. Принцип відкритої архітектури

Основне призначення комп'ютерів визначає і тенденції їх подальшого розвитку. Сюди відноситься, перш за все, **підвищення швидкодії, збільшення об'єму пам'яті і мініатюризація всіх його компонентів.**

Принципова відмінність нинішніх комп'ютерів від попереднього покоління полягає у використанні 64-розрядного мікропроцесора або навіть багатопроцесорних схем. В даний час відомо не менше 500 фірм-виготівників і кожна випускає декілька моделей.

Можливість добудовування системи незалежно виробленими пристроями разом з широкою доступністю інформації про принципи сполучення компонентів і складає відвертість архітектури.

Сучасна операційна система «Windows» або будь-яка інша, вживана на РС може підтримувати роботу програм на адресному просторі зверху 1Гб, причому пам'ять використовується самої OS і прикладними програмами користувача, а також служить буфером для прискорення доступу до дискової пам'яті (так звана процедура **кешированія**), до принтера (процес **спулінга**) або може бути використана як **віртуальний** («електронного») **диск**, що суттєво прискорює роботу ЕОМ.

3.1.1.4. Призначення і взаємодія основних функціональних пристроїв комп'ютера.

Системний блок і центральний процесор

Системний блок має модульну структуру. Його головним елементом є центральний процесор (ЦП), що виконує всі основні арифметичні і логічні операції і що управляє синхронною роботою всієї решти блоків ЕОМ. Згідно принципам роботи ЕОМ, сформульованим в 1945 р. американським

математиком Джоном фон Нейманом, **ЦП** (командний пункт комп'ютера) повинен складатися з двох частин:

❖ Пристрій управління (**ПУ**) сприймає команди програм і організовує їх виконання

❖ арифметико-логічний пристрій (**АЛП**) призначений тільки для обчислень.

В даний час в більшості ЕОМ використовуються нащадки Pentium – мікропроцесора (**ЦП**) (64-х розрядні мікропроцесори 5-го покоління. Раніше використовувалися процесори 8088 і 80286 – 16-ти, і 80386 – 32-х розрядні). Швидкість роботи **ЦП** прямо пропорційна тактовій частоті відповідного мікропроцесора, яка складає для Intel 8088 – 4,7 – 15, для 80286 – 6 – 20, для 80386 – 12 – 33 МГц, і до 4 ГГц для сучасних мікропроцесорів.

Персональні комп'ютери фірми ІВМ на базі процесорів 8088 одержали назву ІВМ РС/ХТ (eXtended Technology), а на базі процесорів 80286 і 80386 ІВМ РС/АТ (Advanced Technology). Сучасні однопроцесорні ЕОМ містять один **ЦП** (процесор даних), крім того, є ще 5-6 службових процесорів, **що** забезпечують роботу ЕОМ, процесор операцій з плаваючою комою; процесор каналів (для організації введення – виведення), графічний процесор, процесор – транслятор мов і т.д. Всі ці процесори по швидкодії порівняні з процесором даних. У сучасних багатопроцесорних комп'ютерах відбувається паралельна обробка даних на декількох **ЦП**, що приводить до збільшення швидкодії комп'ютера при тій же тактовій частоті процесорів.

Коли дані передаються усередині комп'ютерної системи, вони проходять по загальному каналу, до якого мають доступ всі компоненти системи. Цей шлях одержав назву **системний канал**, в який входять: *шини даних, управління і адреси*.

Найважливіша характеристика ЕОМ - об'єм його оперативної пам'яті. Правильніше говорити про три її характеристики: яка пам'ять встановлена на даній ЕОМ, яку пам'ять може адресувати **ЦП** і який об'єм пам'яті підтримує відповідна операційна система ЕОМ (не актуально).

Комп'ютер як універсальний виконавець для обробки різної інформації є сукупністю елементів, кожний з яких може знаходитися в одному з двох стійких станів. Технічна реалізація цих елементів може бути різною і тут не розглядається. Але структурно їх об'єднують в різноманітні пристрої, що виконують певні функції. Функції таких пристроїв і їх взаємодію представляє описувана нижче структурна схема комп'ютера, яка приведена на рис. 3.5, а також архітектурна схема персональної ЕОМ IBM PC/AT, приведена на рис. 3.6.

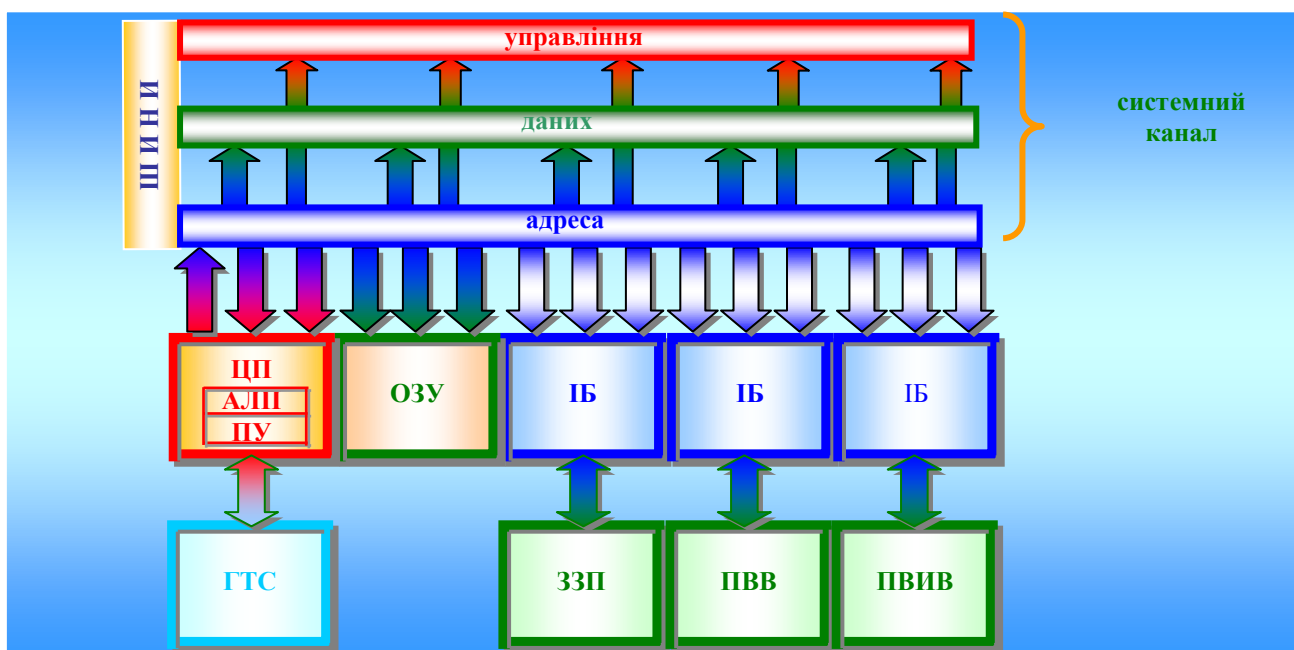


Рисунок. 3.5 - Узагальнена структурна схема комп'ютера:
 ЦП – центральний процесор; АЛП – арифметико-логічний пристрій; ПУ – пристрій управління;
 ОЗУ – оперативна пам'ять; ІБ – інтерфейсний блок; ЗЗП – що зовнішнє запам'ятовує пристрій;
 ПВВ, ПВИВ – пристрої введення і виведення

3.1.1.5. Основні блоки персонального комп'ютера.

ОЗУ

У **оперативній пам'яті (ОЗУ)** прийнято зберігати у вигляді двійкового (шістнадцяткового) коду як саму інформацію, так і інформацію про послідовність операцій, які необхідно здійснити для отримання за вихідними даними необхідного результату. Інформація такого роду називається програмою.

Ці операції не є елементарними командами, а є умовними позначеннями певних послідовностей елементарних команд (наприклад, прочитати один



Рисунок 3.6 - Архитектура персональної ЕВМ IBM PC/AT

або декілька байтів з ОЗУ, записати один або декілька байтів в ОЗУ, скласти два числа і т. п.).

Центральний процесор (ЦП)

Являється найскладнішим пристроєм комп'ютера. Центральний процесор виконує певний набір команд і управляє взаємодією всіх пристроїв комп'ютера. Перед виконанням кожної команди він прочитає її з **ОЗУ** і розпізнає, а потім замінює необхідними елементарними командами і виконує їх. Звичайно, перед виконанням команди ЦП прочитає необхідну інформацію з **ОЗУ** в регістри - спеціально призначені пристрої, а після виконання заносить в них результат і у разі потреби записує його в **ОЗУ**.

Пристрої введення (ПВВ), виведення (ПВИВ)

❖ **Пристрої введення (ПВВ)**. Вони необхідні для того, щоб користувач мав можливість закодувати і розмістити в ОЗУ вихідну інформацію, а також програму для її обробки. Їх може бути декілька різних типів, але такий пристрій, як клавіатура, є завжди.

❖ **Пристрої виведення (ПВИВ).** Ці пристрої призначені для декодування і виведення інформації, одержаної в результаті обробки за програмою, на спеціальний екран (дисплей), на принтер і т.д.

Що зовнішні запам'ятовують пристрої (ЗЗП)

❖ Вони забезпечують необмежено довге зберігання будь-якої інформації (даних, програм) і використовуються в тих випадках, коли інформація не поміщається в **ОЗУ** або уривається робота користувача з комп'ютером. У останньому випадку перед закінченням роботи інформація з **ОЗУ** переписується у **ЗЗП**, а перед відновленням роботи з **ЗЗП** переписується назад в **ОЗУ**.

❖ Перераховані пристрої підключаються до системного каналу за допомогою **інтерфейсних блоків (ІБ)** (або просто інтерфейсів), які погоджують швидкості циркуляції інформації в зв'язуваних каналом пристроях і здійснюють перетворення інформації (наприклад, перекодувала) до того вигляду, в якому її сприймає відповідний пристрій.

Системний канал

❖ Він є останнім елементом загальної структурної схеми і служить для передачі закодованої інформації між пристроями у вигляді електричних імпульсів одного з двох рівнів – низького або високого. Високий рівень відповідає передачі 1, а низький – 0.

❖ До складу системного каналу входить декілька **шин**, які фізично є багатожильними кабелями. По одній з шин (**шині даних**) передаються оброблювана інформація і результат; по іншій шині (**шині адреси**) – інформація про те, де розміщені або повинні бути розміщені в **ОЗУ** дані або команди процесора. Нарешті, по третій шині (**шині управління**) передаються послідовності імпульсів, що управляють. Тактова частота цих імпульсів, що в значній мірі визначає швидкість комп'ютера, задається **генератором тактових сигналів (ГТС)**.

Пристрій, що оперативне запам'ятовує (ОЗП)

Є можливість об'єднувати окремі байти в групи по 2, 4, 8 байтів або більш. Такі групи називають *осередками*, а адресою кожного осередку вважається молодша адреса вхідних в неї байтів. Звернення до осередків ОЗП здійснюється також побайтно. Процедури такого побайтного звернення управляються процесором, який за кодом команди визначає, скільки байтів потрібно записати або вважати.

В процесі обробки інформації здійснюється взаємодія ОЗП і процесора. Із ОЗП в процесор поступають команди і дані, а назад передаються для зберігання результати виконання команд. Звернення або вибір відповідного байта при прочитуванні запису здійснюється за допомогою селектора адреси (СА) – пристрою, що перетворює код адреси, що поступає по адресних шинах із процесора, в сигнал вибору байта, що адресується, як показано на рис. 3.7

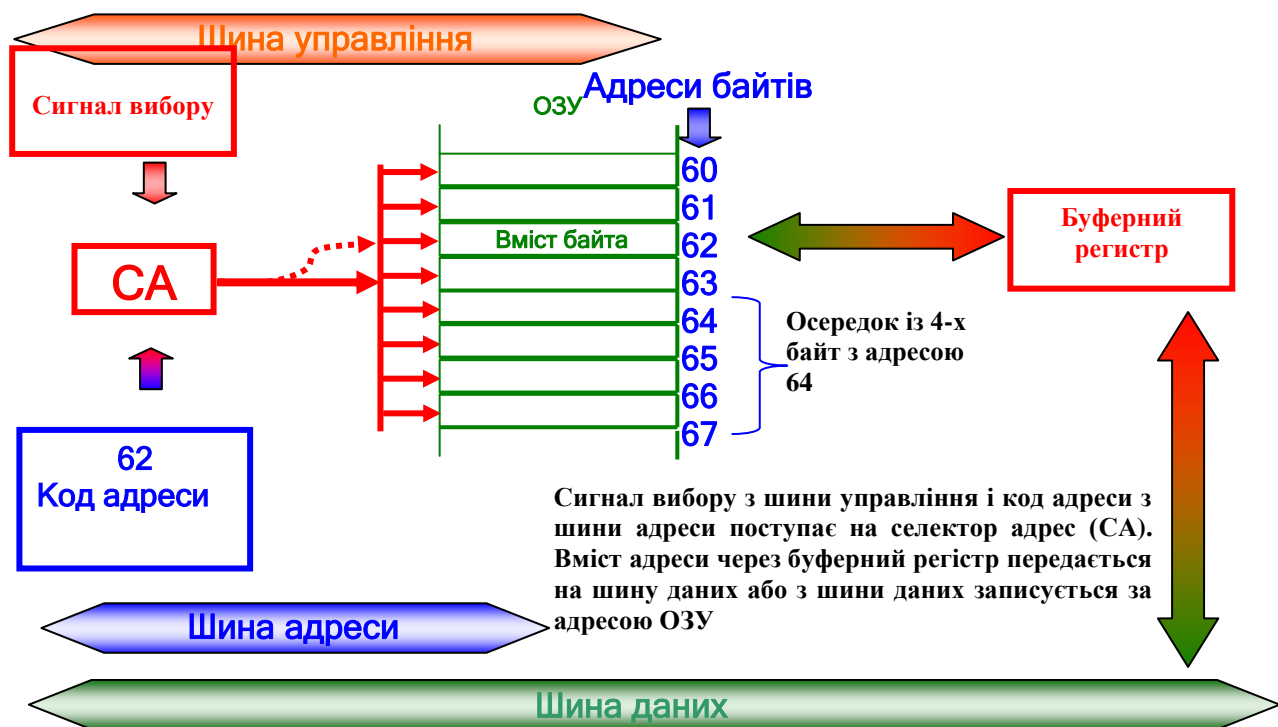


Рисунок. 3.7 - Структурна схема ОЗУ

Пристрої, що запам'ятовують

Вміст вибраного *байта* при прочитуванні переноситься в *буферний реєстр*, а потім по *шині даних* передається в *процесор*. При записі

інформація з *шини даних* передається в *буферний регістр*, а потім у вибраний *байт*. При цьому *буферний регістр* служить для узгодження часу звернення до **ОЗА** і швидкості передачі інформації по шині. Таке звернення до пам'яті називається **адресним**.

Основна пам'ять включає особливий вид пам'яті – *що постійне запам'ятовує пристрій (ПЗП)*, з якого інформацію можна тільки прочитувати. Такі пристрої містять різні стандартні програми, що забезпечують взаємодію комп'ютера з користувачем. Що постійні запам'ятовують пристрої з вбудованими в нього програмами проводяться на заводі-виготівнику апаратних засобів. Користувач не має можливості змінити вміст **ПЗП**.

3.1.2. Огляд периферійних пристроїв

Периферійні (зовнішні) пристрої використовуються для зв'язку користувача або об'єкту управління з комп'ютером. Зв'язок користувача з комп'ютером забезпечується за допомогою дисплеїв, друкуючих і зовнішніх пристроїв (**ЗЗП**), що запам'ятовують, клавіатури і т.п.

3.1.2.1. Пристрої введення і виведення інформації.

❖ **Клавіатура** призначена для введення користувачем інформації в комп'ютер. Сучасна клавіатура містить до 105-110 і більш клавiш різного призначення. Конструкція і розташування клавiш залежить від типу клавіатури і визначається виготівником, хоча більшість клавiш мають стандартне розташування.

❖ **Миша** призначена для управління роботою комп'ютера і програм користувача. У корпусі миші є кульку, тертя якої об килимок миші приводить до обертання датчиків, сигнали з яких після обробки програмою-драйвером миші приводять до переміщення покажчика по екрану дисплея. Натиснення кнопок приводить до виклику і/або виконання тих або інших команд, приписуючих програмі виконувати ті або інші дії. Сучасні оптичні миші не мають кульки і механічних частин, не вимагають чистки кульки і контактних поверхонь роликів.

❖ **Дисплей** є телевізійним монітором або панеллю на рідких кристалах, який служить для відображення інформації, що виводиться, з комп'ютера. Їх підрозділяють на символні і графічні. На екрані символного дисплея може відображатися тільки алфавітно-цифрова інформація; на екрані графічного дисплея – як символна, так і графічна інформація у вигляді схем, графіків, малюнків і креслень (в даний час всі дисплеї тільки графічні).

Як правило, дисплей використовується і при введенні з клавіатури. В цьому випадку користувач може вести візуальне спостереження на екрані дисплея за інформацією, що вводиться в комп'ютер, і при необхідності проводити її редагування.

❖ **Що друкують пристрої (принтер і плоттер).** Ці пристрої, звані принтерами, служать для виведення програм, даних і результатів обробки на широку паперову стрічку або окремі листи. Існують принтери з посимвольним виведенням і рядковим друком. Рядкові принтери мають велику швидкість, оскільки в них рядок друкується відразу повністю, а не символ за символом. У рядкових принтерах швидкість друку може досягати до 1500 рядків/мін. В даний час розрізняють принтери матричні, струменеві і лазерні; чорно-білі і кольорові

❖ Для виведення графічної інформації використовуються реєструючі пристрої – **графічні пристрої (плоттери)**. Вони перетворюють цифрову інформацію в графічну, таку, що виводиться у вигляді креслень, графіків або малюнків на папері.

3.1.2.2. Пристрої, що зовнішні запам'ятовують

Як **ЗЗП** в комп'ютерах використовуються накопичувачі на магнітних гнучких і жорстких магнітних дисках, оптичних дисках (CD/DVD) і флеш-пам'ять.

Поверхневий шар магнітних носіїв інформації утворює магнітне покриття. Запис інформації на це покриття проводиться за допомогою спеціальної головки, що створює на ньому ділянки з різним напрямом намагніченості, як це робиться, наприклад, в побутових магнітофонах.

Ділянка, намагнічена в одному напрямі, відповідає одиниці; у іншому напрямі – нулю. Запис і прочитування інформації проводяться при русі носія щодо магнітних головок.

На гнучких і жорстких магнітних дисках запис інформації проводиться уздовж концентричних кіл на їх поверхні, званих *доріжками*. Число доріжок може бути різним. Вони нумеруються, починаючи від краю до центру диска.

Поверхня диска умовно ділиться на сектори, що дозволяє запис інформації визначити по номеру сектора і номеру доріжки. Вибір потрібного запису на диску здійснюється шляхом позиціонування магнітної головки і обертання диска. Якщо, наприклад, необхідно відшукати запис на 20-й доріжці в 5-му секторі, то для цього потрібно підвести головку до доріжки з цим номером і із заданого сектора провести прочитування (запис) інформації. Цей спосіб доступу до інформації називається *прямим*.

Для збільшення ємкості **ЗП** на жорстких дисках вони об'єднуються по декілька штук. Ємкість пам'яті дискових пакетів вимірюється десятками, сотнями і тисячами гігабайт.

3.2. Мікропроцесори (МП)

Мікропроцесор здійснює управління, як роботою самого комп'ютера, так і процесом обробки інформації за допомогою комп'ютера.

Структурну схему МП розглянемо на прикладі мікропроцесора Z80. Мікропроцесор виконаний на одному кристалі кремнію у вигляді однієї інтегральної мікросхеми, в якій реалізовані всі основні функції процесора. Нижче приведена узагальнена структурна схема восьмирозрядного мікропроцесора, який обробляє інформацію послідовно по одному байту. Існують мікропроцесори, оброблювальні відразу 2, 4 або 8 байтів інформації. Їх називають відповідно 16-, 32- або 64-розрядними мікропроцесорами.

3.2.1. Структура мікропроцесора

Структура мікропроцесора приведена на рис. 3.8.

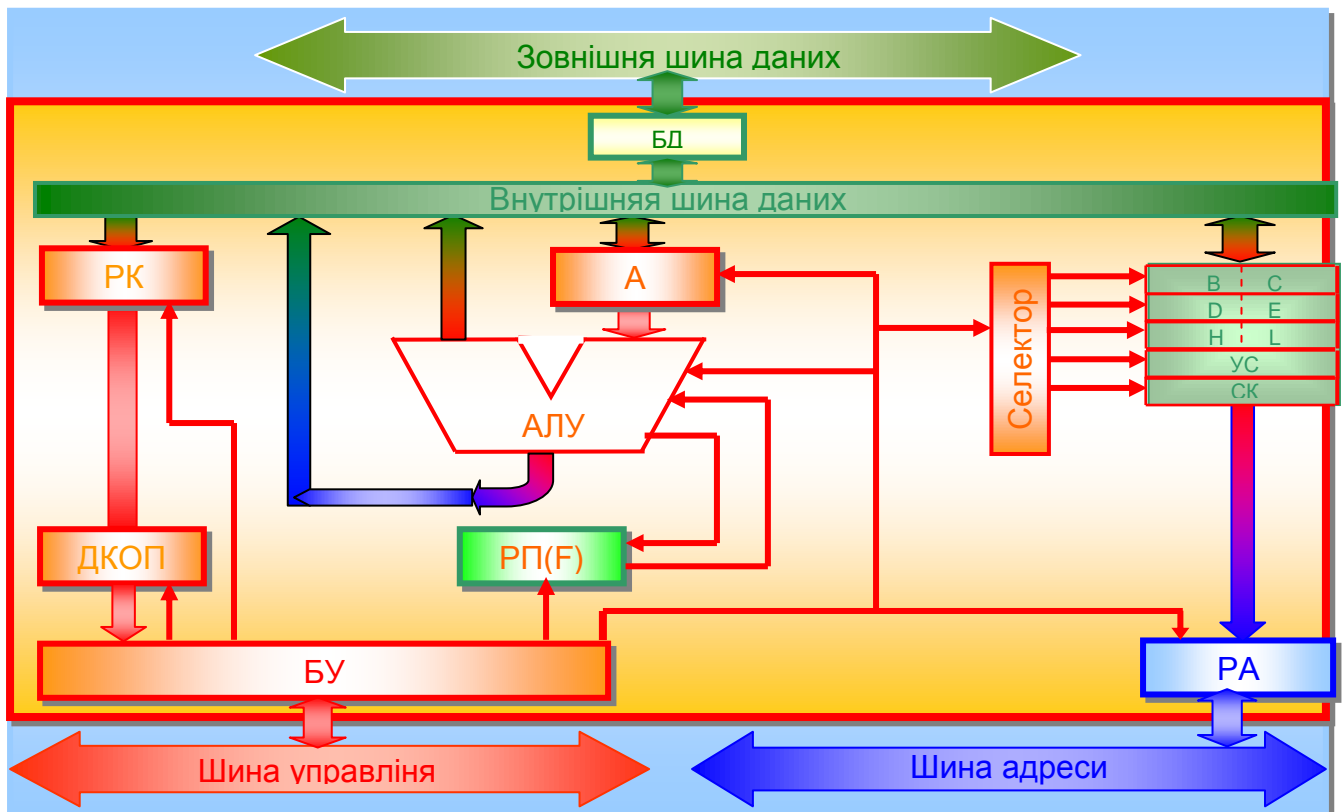


Рисунок 3.8 - Структурна схема восьмирозрядного мікропроцесора

Необхідними складовими будь-якого мікропроцесора є пристрій управління, ряд робочих регістрів і арифметико-логічний пристрій. Обмін інформацією між ними здійснюється по внутрішніх шинах. Мікропроцесори можуть включати і інші компоненти, розширяльні їх функціональні можливості.

Пристрій управління (ПУ). До основних функцій пристрою управління відносяться вибірка команд програми з основної пам'яті, їх розшифровка і виконання. Воно складається з блоків управління і наступних регістрів: лічильника команд (ЛК), регістра команд (РК), покажчика стека (ПС) і регістра ознак(РО).

Лічильник команд (ЛК), званий також програмним лічильником, служить для утворення і зберігання адрес, вказуючих на розміщення команд, вибраних з основної пам'яті. Команди мають різну довжину і можуть займати від одного до декількох байтів пам'яті. При кожній передачі байта з пам'яті в процесор блок управління збільшує вміст лічильника команд на

одиницю. Після прочитування всіх байтів команди лічильник команд міститиме адресу першого байта наступної команди. Перед виконанням програми в лічильник команд поміщається адреса першого байта першої команди програми.

3.2.1.1. Апаратні реєстри ЦП.

Реєстр команд (РК) приймає і зберігає перший байт вибіраної з пам'яті команди на час розшифровки її коду операції (**КОП**). Звичайно під код операції в команді відводиться вісім двійкових розрядів, що дозволяє реалізувати до $2^8 = 256$ різних кодів операцій. Незалежно від довжини команди код операцій завжди розміщується в першому байті команди. Розшифровка першого байта команди проводиться дешифратором коду операцій (**ДКОП**). В результаті розшифровки визначається вигляд операції і кількість байтів в команді. Під впливом цієї інформації блок управління (**БУ**) виробляє послідовність сигналів, що управляють, забезпечують виконання цієї операції.

Показчик стека (ВУС) є 16-розрядним реєстром, що служить для формування адреси стекової пам'яті.

Реєстр ознак (РО) використовується для фіксації певних ситуацій, що виникають в результаті виконання операції. Код ситуації називається *ознакою результату*. Ознака може приймати значення 0 або 1. Відзначимо деякі з можливих ознак.

- ❖ Ознака перенесення вказує на необхідність переносу одиниці при складанні двох восьмирозрядних чисел в молодший значущий розряд наступного байта.
- ❖ Ознака нуля фіксує появу нульового результату операції
- ❖ ознака знаку – знак результату операції.
- ❖ Ці та інші ознаки використовуються для зміни ходу обробки інформації.

3.2.1.2. Робочі реєстри МП.

Робочі реєстри МП служать для часового зберігання інформації. Розрізняють два види робочих реєстрів: **реєстри адреси і арифметичні реєстри**.

Реєстри адреси використовуються для адресації даних і команд в **ОЗУ**, **арифметичні реєстри** – для часового зберігання оброблюваної інформації.

Робочі реєстри часто називають *реєстрами загального призначення (РЗП)*, а спеціальний реєстр, використовуваний для зберігання результатів, одержаних в **арифметико-логічному пристрої (АЛП)**, – *акумулятором (А)*.

Реєстри загального призначення і акумулятор забезпечують найшвидший доступ до інформації, **що** зберігається в них, що позитивно позначається на швидкості виконання операцій і, отже, на швидкодії комп'ютера в цілому. Для адресації **РЗП** і **А** в коді команді відводиться три двійкові розряди. Тому можна адресуватися до $2^3 = 8$ реєстрам. Вони, як правило, позначаються літерами **B, C, D, E, H і L**, а також **A і F (PO)**. Реєстри **B і C, D і E, H і L** об'єднані попарно, що дозволяє обробляти як 1 байт, так і 2 байти інформації відразу. Через реєстрові пари звертаються до елементів основної пам'яті. У деяких типах мікропроцесорів для цих цілей використовується тільки реєстрова пара **H, L**. Тому перш ніж звернутися до потрібного елементу пам'яті, необхідно в цю реєстрову пару занести адресу цього осередку.

3.2.1.3. Арифметико-логічний пристрій (АЛП)

АЛП призначене для виконання арифметичних і логічних операцій над даними, проте саме **АЛП** здатне виконати лише обмежений набір цих операцій. Цей набір може бути різним і залежить від типу мікропроцесора. Як правило, **АЛП** може виконувати арифметичні операції складання, віднімання і порозрядного зсуву чисел, а також деякі логічні операції, що використовуються для порозрядного порівняння двох- і восьмирозрядних цифрових кодів. Інші арифметичні і логічні операції можуть бути запрограмовані на базі операцій, що виконуються **АЛП**.

Арифметико-логічний пристрій виконує вказані операції або над вмістом акумулятора і одного **РЗП**, або над вмістом акумулятора і елемента пам'яті, адреса якої міститься в одній з реєстрових пар, наприклад в **H, L**. У деяких операціях **АЛП** також використовується значення якої-небудь ознаки з реєстра ознаки. **Результат операції завжди фіксується в акумуляторі.**

3.2.1.4. Блоки МП

Для часового зберігання сформованої процесором адреси пам'яті використовується **реєстр адреси (РА)**. За цією адресою записується або прочитується байт даних або байт команди. Для передачі адреси по адресних шинах в пам'ять, його перетворення в сигнал вибору елемента пам'яті потрібний якийсь час, зване часом доступу до пам'яті. Протягом цього часу адреси пам'яті зберігаються в реєстрі адреси.

Буфер даних (БД) служить для організації обміну інформацією між внутрішньою і зовнішньою шинами даних.

3.2.2. Виконання команд мікропроцесором.

Розглянемо повний цикл роботи **ЦП** при виконанні команди.

Вибірка кожного байта команди проводиться в два етапи. На першому етапі адреса байта команди, що міститься в **ЛК**, передається в **РА**, а потім по адресних шинах в пристрій основної пам'яті, де відбувається вибір байта, що адресується. На другому етапі – передача вмісту байта, що адресується, по шинах даних в процесор і збільшення вмісту **ЛК** на одиницю. При цьому, якщо вибирається не останній байт команди, то новий вміст **ЛК** представлятиме адресу наступного байта поточної команди, а якщо вибирається останній байт, то адреса першого байта наступної команди.

При вибірці першого байта команди його вміст передається і фіксується в **РО**. Дешифратор коду операцій (**ДКОП**) по вмісту реєстра команд, що поступає на його вхід, визначає вид операції, число байтів в команді, а також спосіб адресації, використовуваний в команді. На базі цієї інформації блок управління задає необхідну послідовність дій, що управляють, приводять до вибірки наступних байтів команди або до виконання команди.

3.2.2.1. Приклад виконання команди МП

Вибірка і виконання команди, по якій вміст регістра загального призначення підсумовується з вмістом акумулятора А, а результат операції фіксується в акумуляторі. Ця команда однобайтова з прямою регістровою адресацією. У коді команди 100000012 = 8116 перші п'ять старших розрядів утворюють код операції; три останніх – код адреси регістра С. Пусть до вибірки і виконання команди вміст ЛК складає 017616, а вміст регістра С і акумулятора відповідно рівні 2F16 і 3A16.

Вибірка команди починається з передачі адреси команди 017616 в РА, а потім в пристрій пам'яті для пошуку елемента пам'яті з цією адресою. Після цього вміст осередку, що адресується, 100000012 передається і записується в РО. За кодом операції і адресної частини команди ДКОП розпізнає, що вибірка команди закінчена і над вмістом акумулятора і регістра С необхідно провести операцію підсумовування. Потім блок управління виробляє послідовність сигналів, під впливом яких вміст акумулятора 3A16 і регістра С 2F16 поступає на входи АЛП. Арифметико-логічний пристрій проводить операцію підсумовування над даними, що поступили. Результат операції $2F16 + 3A16 = 6916 = 011010012$ переноситься в акумулятор, заміщаючи в ньому колишнє значення 3A16. На цьому виконання команди закінчується. Починається вибірка наступної команди з адресою 017716, утвореним в ЛК при завершенні вибірки попередньої команди.

4. ХІД РОБОТИ

- 4.1. Вивчити методичні вказівки до лабораторної роботи.
- 4.2. Відповісти на питання в у кінці методичних вказівок.
- 4.3. Включити комп'ютер.
- 4.4. Запустити на виконання програму «**prCP.exe**».
- 4.5. Виконати завдання з розділу 6 методичних вказівок для свого варіанту.
- 4.6. Скласти звіт по лабораторній роботі. Уміти відповісти на питання, пов'язані із зміною вмісту регістрів ЦП і ОЗУ.

5. ЗМІСТ ЗВІТУ

5.1. Мета роботи.

5.2. Результати виконання завдань лабораторної роботи – файли з протоколом роботи програми для кожної команди вашого варіанту, що створюються програмою **prCP.exe** у кореневому каталозі на диску C:\.

5.3. Таблиця порівняння змін реєстрів загального призначення і службових реєстрів центрального процесора до і після виконання команди (а також зміни в ОЗУ, якщо останні мали місце, з вказівкою адрес і значень байтів у відповідних осередках ОЗУ).

5.4. Відповіді на контрольні питання.

6. ЗАВДАННЯ ДЛЯ ВИКОНАННЯ РОБОТИ

Умовні позначення в приведених нижче таблицях:

- ❖ ADC - ADDITION with CARRY - СКЛАДАННЯ з урахуванням ПЕРЕНОСУ. (Операція складання з урахуванням стану прапора переносу - прапора 3 реєстра F).
- ❖ ADD - ADDITION - СКЛАДАННЯ.
- ❖ AND - логічне множення "И".
- ❖ BIT - BIT – test. Перевірити БІТ. (Виконується перевірка: включений чи ні даний біт в реєстрі або в елементи пам'яті).
- ❖ DEC - DECREMENT - ЗМЕНШЕННЯ (на одиницю).
- ❖ DJNZ - DECREMENT and JUMP if NOT ZERO - ЗМЕНШИТИ (на одиницю) і ПЕРЕЙТИ, якщо НЕ НУЛЬ. (Зменшується на одиницю вміст реєстра BC і якщо нуль ще не досягнутий, виконується відносний перехід на задану величину байтів).
- ❖ INC - INCREMENT - ЗБІЛЬШИТИ (на одиницю).
- ❖ JP - JUMP - ПЕРЕХІД (абсолютний).
- ❖ JR - JUMP RELATIVE - ПЕРЕХІД ВІДНОСНИЙ.
- ❖ LD - LOAD - ЗАГРУЗКА. (Виконується копіювання даних з реєстра в реєстр, з реєстра в пам'ять або з пам'яті в реєстр).
- ❖ N – яке-небудь значення 1-го байта.

- ❖ NN – двобайтова величина.
- ❖ OR - логічне "АБО".
- ❖ RES - RESET - ВИКЛЮЧЕННЯ. (Вимикається заданий біт у вказаному регістрі або елементі пам'яті).
- ❖ S – Величина відносного переходу від -127_{10} до 127_{10} або від $-7F$ до $7F$.
- ❖ SBC - SUBTRACT with CARRY - ВІДНІМАННЯ з урахуванням прапора ПЕРЕНОСУ.
- ❖ SET - ВКЛЮЧИТИ (вказаний біт у вказаному регістрі або елементі пам'яті).
- ❖ SUB - SUBTRACT - ВІДНІМАННЯ.
- ❖ XOR - що логічне виключає "АБО".

6.1. Варіанти завдань для виконання лабораторної роботи

6.1.1. Варіанти завдання 1. Загрузка (пересилка) даних регістрів ЦП

Вивчити виконання двох команд запису (пересилки) даних в регістри центрального процесора, приведених в нижченаведеній таблиці 6.1:

Таблиця 6.1 - Команди запису (пересилки) даних в регістри центрального процесора

№	Мнемоніка	Код	Вплив на прапори				Кількість тактів	Примітка
			C	Z	P/V	S		
1	LD A, N	3E N	-	-	-	-	7	
2	LD B, N	06 N	-	-	-	-	7	
3	LD C, N	0E N	-	-	-	-	7	
4	LD D, N	16 N	-	-	-	-	7	
5	LD E, N	1E N	-	-	-	-	7	
6	LD H, N	26 N	-	-	-	-	7	
7	LD L, N	2E N	-	-	-	-	7	
8	LD A, (NN)	3A NN	-	-	-	-	13	
9	LD HL, (NN)	2A NN	-	-	-	-	16	
10	LD BC, (NN)	ED 4B NN	-	-	-	-	20	
11	LD DE, (NN)	ED 5B NN	-	-	-	-	20	
12	LD IX, (NN)	DD 2A NN	-	-	-	-	20	
13	LD IY, (NN)	FD 2A NN	-	-	-	-	20	
14	LD A, (HL)	7E	-	-	-	-	7	
15	LD A, (BC)	0A	-	-	-	-	7	
16	LD A, (DE)	1A	-	-	-	-	7	
17	LD H, (HL)	66	-	-	-	-	7	
18	LD L, (HL)	6E	-	-	-	-	7	
19	LD B, (HL)	46	-	-	-	-	7	
20	LD C, (HL)	4E	-	-	-	-	7	

6.1.2. Варіанти завдання 2. Пересилка (загрузка) даних в ОЗУ

Вивчити виконання двох команд запису (пересилки) даних в регістри пристрою, що оперативного запам'ятовує, приведених в нижченаведеній таблиці 6.2:

Таблиця 6.2 - Команди запису (пересилки) даних в ОЗУ

№	Мнемоніка	Код	Вплив на прапори				Кількість тактів	Примітка
			C	Z	P/V	S		
1	LD (NN), A	32 NN	-	-	-	-	13	
2	LD (IY+S), C	FD 71 S	-	-	-	-	20	
3	LD (NN), HL	22 NN	-	-	-	-	16	
4	LD (NN), BC	ED 43 NN	-	-	-	-	20	
5	LD (NN), DE	ED 53 NN	-	-	-	-	20	
6	LD (NN), IX	DD 22 NN	-	-	-	-	20	
7	LD (NN), IY	FD 22 NN	-	-	-	-	20	
8	LD (HL), A	77	-	-	-	-	7	
9	LD (BC), A	02	-	-	-	-	7	
10	LD (DE), A	12	-	-	-	-	7	
11	LD (HL), H	74	-	-	-	-	7	
12	LD (HL), L	75	-	-	-	-	7	
13	LD (HL), B	70	-	-	-	-	7	
14	LD (HL), C	71	-	-	-	-	7	
15	LD (HL), D	72	-	-	-	-	7	
16	LD (HL), E	73	-	-	-	-	7	
17	LD (HL), N	36 N	-	-	-	-	10	
18	LD (IX+S), A	DD 77 S	-	-	-	-	19	
19	LD (IX+S), H	DD 74 S	-	-	-	-	19	
20	LD (IY+S), H	FD 74 S	-	-	-	-	19	

6.1.3. Варіанти завдання 3. Переходи в програмі

Вивчити виконання команд переходу, приведених в нижченаведеній таблиці 6.3:

Таблиця 6.3 - Команди переходу

№	Мнемоніка	Код	Вплив на прапори				Кількість тактів	Примітка
			C	Z	P/V	S		
1	JP NN	C3 NN	-	-	-	-	10	
2	JP (HL)	E9	-	-	-	-	4	
3	JP (IX)	DD E9	-	-	-	-	8	
4	JP (IY)	FD E9	-	-	-	-	8	
5	JP C, NN	DA NN	-	-	-	-	10	
6	JP NC, NN	D2 NN	-	-	-	-	10	
7	JP Z, NN	CA NN	-	-	-	-	10	
8	JP NZ, NN	C2 NN	-	-	-	-	10	
9	JP P, NN	F2 NN	-	-	-	-	10	

№	Мнемоніка	Код	Вплив на прапори				Кількість тактів	Примітка
			C	Z	P/V	S		
10	JP M, NN	FA NN	-	-	-	-	10	
11	JP PE, NN	EA NN	-	-	-	-	10	
12	JP P0, NN	E2 NN	-	-	-	-	10	
13	JR C, S	38 S	-	-	-	-	7	Якщо умова не виконується
							12	Якщо умова виконується
14	JR NC, S	30 S	-	-	-	-	7	Якщо умова не виконується
							12	Якщо умова виконується
15	JR Z, S	28 S	-	-	-	-	7	Якщо умова не виконується
							12	Якщо умова виконується
16	JR NZ, S	20 S	-	-	-	-	7	Якщо умова не виконується
							12	Якщо умова виконується
17	DJNZ S	10 S	-	-	-	-	8	Якщо B=0
							13	Якщо B=0

6.1.4. Варіанти завдання 4. Операції з даними

Вивчити виконання команд арифметики, приведених в нижченаведеній таблиці 6.4:

Таблиця 6.4 - Команди арифметики (складання і т.п.)

№	Мнемоніка	Код	Вплив на прапори				Кількість тактів	Примітка
			C	Z	P/V	S		
1	ADD A, N	C6 N	*	*	V	*	7	
2	ADD A, A	87	*	*	V	*	4	
3	ADD A, H	84	*	*	V	*	4	
4	ADD A, L	85	*	*	V	*	4	
5	ADD A, B	80	*	*	V	*	4	
6	ADD A, C	81	*	*	V	*	4	
7	ADD A, D	82	*	*	V	*	4	
8	ADD A, E	83	*	*	V	*	4	
9	INC A	3C	-	*	V	*	4	
10	INC H	24	-	*	V	*	4	
11	INC L	2C	-	*	V	*	4	
12	INC B	04	-	*	V	*	4	
13	INC (HL)	34	-	*	V	*	11	
14	ADC A, N	CE N	*	*	V	*	7	
15	ADC A, A	8F	*	*	V	*	4	
16	ADC A, H	8C	*	*	V	*	4	
17	ADC A, L	8D	*	*	V	*	4	

№	Мнемоніка	Код	Вплив на прапори				Кількість тактів	Примітка
			C	Z	P/V	S		
18	ADC A, B	88	*	*	V	*	4	
19	ADC A, C	89	*	*	V	*	4	
20	ADC A, D	8A	*	*	V	*	4	

6.1.5. Варіанти завдання 5. Операції з даними

Вивчити виконання команд арифметики, приведених в нижченаведеній таблиці 6.5:

Таблиця 6.5 - Команди арифметики (віднімання і т.п.)

№	Мнемоніка	Код	Вплив на прапори				Кількість тактів	Примітка
			C	Z	P/V	S		
1	SUB N	D6 N	*	*	V	*	7	
2	SUB A	97	*	*	V	*	4	
3	SUB H	94	*	*	V	*	4	
4	SUB L	95	*	*	V	*	4	
5	SUB B	90	*	*	V	*	4	
6	SUB C	91	*	*	V	*	4	
7	SUB D	92	*	*	V	*	4	
8	SUB E	93	*	*	V	*	4	
9	SUB (HL)	96	*	*	V	*	7	
10	DEC A	3D	-	*	V	*	4	
11	DEC H	25	-	*	V	*	4	
12	DEC L	2D	-	*	V	*	4	
13	DEC B	05	-	*	V	*	4	
14	DEC C	0D	-	*	V	*	4	
15	DEC D	15	-	*	V	*	4	
16	DEC E	1D	-	*	V	*	4	
17	DEC (HL)	35	-	*	V	*	11	
18	SBC A, A	9F	*	*	V	*	4	
19	SBC A, H	9C	*	*	V	*	4	
20	SBC A, L	9D	*	*	V	*	4	

6.1.6. Варіанти завдання 6. Перевірка умов.

Вивчити виконання команд порівняння, приведених в нижченаведеній таблиці 6.6:

Таблиця 6.6 - Команди порівняння

№	Мнемоніка	Код	Вплив на прапори				Кількість тактів	Примітка
			C	Z	P/V	S		
1	AND N	E6 N	*	*	p	*	7	
2	AND A	A7	0	*	p	*	4	
3	AND H	A4	0	*	p	*	4	
4	AND L	A5	0	*	p	*	4	

№	Мнемоніка	Код	Вплив на прапори				Кількість тактів	Примітка
			C	Z	P/V	S		
5	AND B	A0	0	*	p	*	4	
6	AND C	A1	0	*	p	*	4	
7	AND D	A2	0	*	p	*	4	
8	AND E	A3	0	*	p	*	4	
9	AND (HL)	A6	0	*	p	*	7	
10	OR N	F6 N	*	*	p	*	7	
11	OR A	B7	0	*	p	*	4	
12	OR H	B4	0	*	p	*	4	
13	OR L	B5	0	*	p	*	4	
14	OR B	B0	0	*	p	*	4	
15	OR C	B1	0	*	p	*	4	
16	OR D	B2	0	*	p	*	4	
17	OR E	B3	0	*	p	*	4	
18	OR (HL)	B6	0	*	p	*	7	
19	XOR N	EE N	*	*	p	*	7	
20	XOR A	AF	0	*	p	*	4	

6.1.7. Варіанти завдання 7. Установка бітів

Вивчити виконання команд установки (перевірки) бітів, приведених в нижченаведеній таблиці 6.7:

Таблиця 6.7 - Команди установки (перевірки) бітів

№	Мнемоніка	Код	Вплив на прапори				Кількість тактів	Примітка
			C	Z	P/V	S		
1	SET 0, A	CB C7	-	-	-	-	8	
2	SET 1, H	CB C0	-	-	-	-	8	
3	SET 2, L	CB D5	-	-	-	-	8	
4	SET 3, B	CB D8	-	-	-	-	8	
5	SET 4, C	CB E1	-	-	-	-	8	
6	SET 5, D	CB EA	-	-	-	-	8	
7	SET 6, E	CB F3	-	-	-	-	8	
8	SET 7, A	CB FF	-	-	-	-	8	
9	RES 0, H	CB 84	-	-	-	-	8	
10	RES 1, L	CB 8D	-	-	-	-	8	
11	RES 2, B	CB 90	-	-	-	-	8	
12	RES 3, C	CB 99	-	-	-	-	8	
13	RES 4, D	CB A2	-	-	-	-	8	
14	RES 5, E	CB AB	-	-	-	-	8	
15	RES 6, (HL)	CB B6	-	-	-	-	15	
16	RES 7, A	CB BF	-	-	-	-	8	
17	BIT 0, H	CB 44	-	*	?	?	8	
18	BIT 1, L	CB 4D	-	*	?	?	8	
19	BIT 2, B	CB 50	-	*	?	?	8	
20	BIT 3, C	CB 59	-	*	?	?	8	

7. КОНТРОЛЬНІ ПИТАННЯ

- 7.1. Перерахуйте основні параметри, що характеризують роботу комп'ютера.
- 7.2. Що знаходиться в системному блоці?
- 7.3. Опишіть тенденції розвитку комп'ютерів.
- 7.4. Що таке принцип відкритої архітектури?
- 7.5. Перерахуйте основні блоки персонального комп'ютера.
- 7.6. Опишіть призначення Центрального процесора.
- 7.7. Перерахуйте пристрої введення/виведення в комп'ютері.
- 7.8. Перерахуйте види пам'яті в комп'ютері.
- 7.9. Що таке системний канал?
- 7.10. Опишіть структуру простого центрального процесора.
- 7.11. Перерахуйте основні апаратні регістри центрального процесора.

Навчальне видання

МЕТОДИЧНІ ВКАЗІВКИ
“Елементи інформатики”
до практичних занять з курсу “Інформатика”
для студентів спеціальності мікро- та наноелектроніка

Укладачі: ШКАЛЕТО Володимир Іванович
ХРИПУНОВ Геннадій Семенович
ЧЕРНИХ Олена Петрівна

Відповідальний за випуск *

Роботу до видання рекомендував проф. **

Редактор _____
Коректор _____

План 2008 р., поз. ____/____

Підп. до друку __. __. __. Формат 60x84 1/16. Папір офсетний.
Друк – ризографія. Гарнітура Times New Roman. Ум. друк. арк. ____.
Обл.-вид. арк. _____. Наклад ____ прим. Зам. № _____. Ціна договірна.

Видавничий центр НТУ “ХП”.

Свідоцтво про державну реєстрацію ДК №116 від 10.07.2000 р.

61002, Харків, вул. Фрунзе, 21.

Друкарня НТУ “ХП”.
61002, Харків, вул. Фрунзе, 21.