**Лабораторне заняття 1**

**Основи обчислювальної та мікропроцесорної техніки**

**Мета:** ознайомитися з основами обчислювальної техніки, а саме: системами числення, булевою алгеброю, синтезом комбінаційних схем та автоматів з пам'яттю. Отримати практичні навички.

Зміст роботи

**Завдання 1:** Згідно з вказаним викладачем варіантом переведіть десяткове число А10 (таблиця 1.1) у двійкову та шістнадцяткову системи числення



У двійкову систему:

Ціла частина:

291(10) = 100100011(2)

|  |  |
| --- | --- |
| Число | Частка(біт) |
| 291/2 | 1 |
| 145/2 | 1 |
| 72/2 | 0 |
| 36/2 | 0 |
| 18/2 | 0 |
| 9/2 | 1 |
| 4/2 | 0 |
| 2/2 | 0 |
| 1 |  |

Дробова частина:

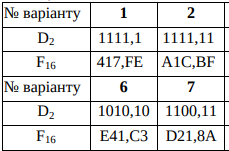
0,73(10)=0.1011101(2)

|  |  |
| --- | --- |
| 0 | 73 |
|  | \*2 |
| 1 | 46 |
|  | \*2 |
| 0 | 92 |
|  | \*2 |
| 1 | 84 |
|  | \*2 |
| 1 | 68 |
|  | \*2 |
| 1 | 36 |
|  | \*2 |
| 0 | 72 |
|  | \*2 |
| 1 | 44 |

У шістнадцяткову систему счислення:

123.BAE1

**Завдання 2**: Згідно з вказаним викладачем варіантом переведіть двійкове число D2 та шістнадцяткове число F16 (таблиця 1.2) в десяткову систему числення.



1. Переведення числа з двійкової у десяткому систему числення:

1100 => =8+4+0+0=12

0,11 => =0.5+0.25=0.75

1100,011=12.075

1. Переведення числа з шістнадцяткової у десяткому систему числення

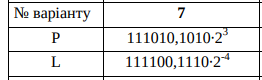
D21 =>

0.8A =

Завдання 3: Згідно з вказаним викладачем варіантом виконайте наступні дії над числами P2 і L2, що задані в формі з плаваючою комою (таблиця 1.3):

а) додавання;

б) віднімання



Для початку переведемо наші числа в нормальну форму, для цього виконаємо відповідне перетворення чисел. В результаті отримаємо наступне.

|  |  |  |
| --- | --- | --- |
| № варіанту | 7 | |
| P | 111010,1010∙23 | 111010101,0 |
| L | 111100,1110∙2-4 | 11,11001110 |

**Операція додавання.** Маючи відповідне представлення чисел в двійковій формі можемо провести операцію побітового додавання. Суть виконання даної опреації досить проста, під час виконання додавання варто пам’ятати декілька правил на основі яких і виконується додвання, під час додавання двох нулів в результаті отримаємо нуль, під час додавання нуля та одиниці в результат записується 1, під час додавання двох одиниць в результат записується нуль, але в старший розряд переноситься значення 1. Вище описані операції додавання зображені в таблиці №3.2.

Розпочнімо процес виконаня операції додавання над нашими числами P та L. Результат виконання операції додавання зобразимо у вигляді таблиці №3.3.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| + | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | , | 0 |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  | 1 | 1 | , | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |  |
| = | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |  | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |  |

Виконаємо операцію віднімання на числами P та L. Виконання даної операції у нашому випадку може бути виконано у прямому коді, так як представлені числа мають однаковий знак числа. У випадку, коли числа мають протилежні знаки, операція віднімання не може бути виконано у прямому коді. Тоді така операція зводиться до виконання операції додавання над числами у прямому та доповняльному коді. Виконаємо операція віднімання над нашими числами. В такому випадку отримаємо наступний результат.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| - | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | , | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |
|  |  |  |  |  |  |  |  | 1 | 1 | , | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |  |
| = | 1 | **1** | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | , | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 |

Як можемо побачити, операція віднімання виконана успішно, але в такому випадку отриманий результат віднімання не матиме достатньо точність для виконання деяких інших операцій. Тому така операція віднімання також можемо бути виконанна іншими способами, котрі забезпечують вищого рівня точності.

**Завдання 4**: Згідно з заданим викладачем варіантом запишіть досконалу диз'юнктивну нормальну форму (ДДНФ) для логічної функції F чотирьох змінних, яка задана таблицею істинності (таблиця 1.4).

Використовуючи карти Карно запишіть вираз для мінімальної форми і побудуйте функціональну електричну схему (перевірити роботу у Electronic Workbench(Multisim).

Реалізувати функцію логічного виразу на будь-якій мові програмування: асемблер, С, С ++, С # тощо. Вхідним параметром функції є однобайтова змінна.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x1 | x2 | x3 | x4 | f |
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
|  |  |  |  |  |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

- 0000

0010

x2 - 0100

x2x3 -0110

x1x4 -1001

x1x2x4 - 1101

x1x2x3 - 1110

x1x2x3x4 - 1111

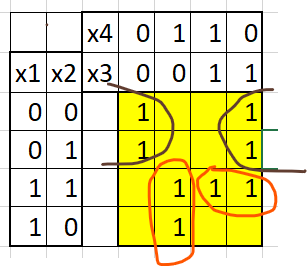
Записуємо досконалу диз`юнктивну нормальну форму (ДДНФ), за допомогою логічного додавання двійкових наборів на яких функція приймає значення, що дорівнює 1:

F36(x1x2x3x4)дднф = ++x2 +x2x3+x1x4 +x1x2x4 +x1x2x3+x1x2x3x4

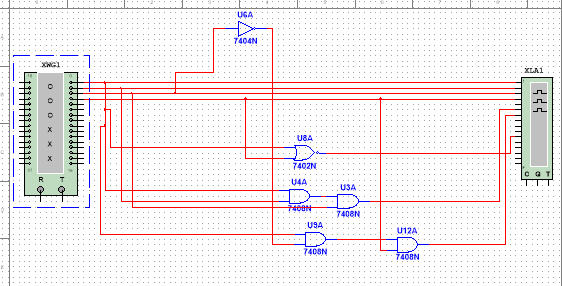
¬x1¬x2¬x3¬x4 ∨ ¬x1¬x2x3¬x4 ∨ ¬x1x2¬x3¬x4 ∨ ¬x1x2x3¬x4 ∨ x1¬x2¬x3x4 ∨ x1x2¬x3x4 ∨ x1x2x3¬x4 ∨ x1x2x3x4

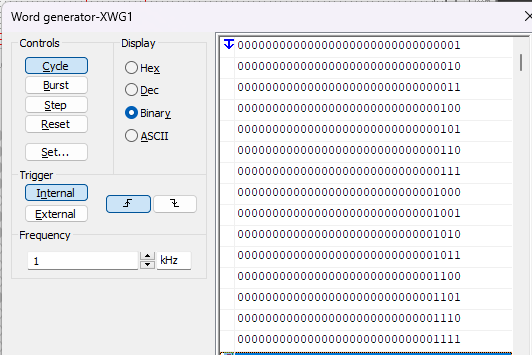
Мінімізуємо ДДНФ:

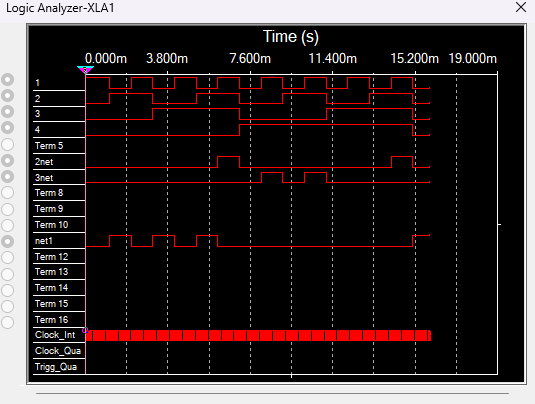
F36(x1x2x3x4)мдднф = ++x2 +x2x3+x1x4 +x1x2x4 +x1x2x3+x1x2x3x4 = x2 *+ +* x1x4 =

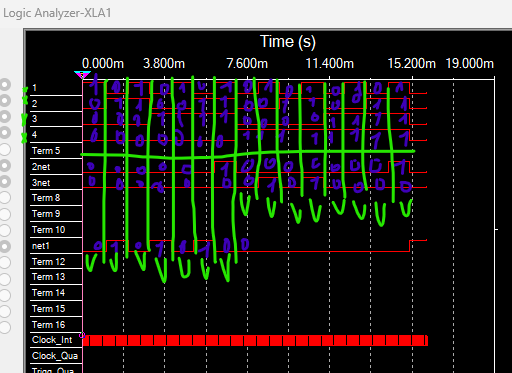


x1x2x3x4мдднф =

**

**

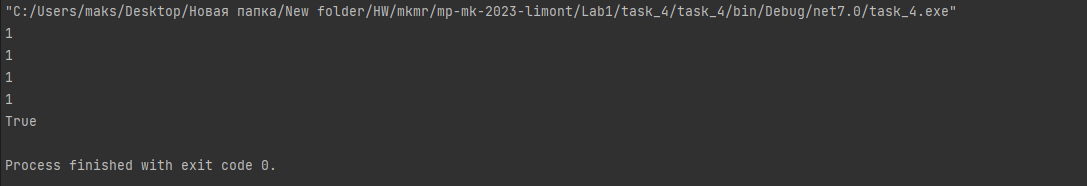
**

**

*Программа реалізована на c#*

int x1 = int.Parse(Console.ReadLine());  
int x2 =int.Parse(Console.ReadLine());  
int x3 =int.Parse(Console.ReadLine());  
int x4 = int.Parse(Console.ReadLine());  
  
if (x1 == x4 && x1 == 0)  
{  
 Console.WriteLine(true);  
}  
if (x1 == x2 && x2==x3 && x1 == 1)  
{  
 Console.WriteLine(true);  
}  
if (x1 == x4 && x3==0 && x1 == 1)  
{  
 Console.WriteLine(true);  
}

Результат програми:

**

**Висновки:** В ході виконання лабораторної роботи було закріплено навички та вміння з основами роботи обчислювальної техніки. Під час виконання лабораторної роботи було проведено роботу з двійковими числами, продемонстровано процес їх перетворення, додавання та вднімання. Проведено відповідні операції побудови логічних функцій на основі таблиці істинності, а також їх подальша мінімізація на основі карт Карно та методу поглинання. Також в ході виконання роботи було побудовано логічні функції для забезпечення роботи лічильника, робота якого зображена за допомогою графа.