Федеральное государственное образовательное бюджетное учреждение

высшего образования

**«Финансовый университет при Правительстве Российской Федерации»**

**(Финансовый университет)**

Факультет прикладной математики и информационных технологий

**Департамент анализа данных, принятия решений и финансовых технологий**

**Контрольная работа №1 по дисциплине**

**«Организация вычислительных систем»**

Вариант №6

**Выполнил:**

Студент группы ПИ19-4

Деменчук Георгий Максимович

**Преподаватель:**

Профессор д/н

Корнеев Николай Владимирович

**Москва 2019**

Содержание

[Задание №1 3](#_Toc22834387)

[Условие 3](#_Toc22834388)

[Решение 3](#_Toc22834389)

[Задание №2 5](#_Toc22834390)

[Условие 5](#_Toc22834391)

[Решение 5](#_Toc22834392)

[Задание №3 14](#_Toc22834393)

[Условие 14](#_Toc22834394)

[Решение 14](#_Toc22834395)

[Задание №4 18](#_Toc22834396)

[Условие 18](#_Toc22834397)

[Решение 18](#_Toc22834398)

[Список литературы 19](#_Toc22834399)

# Задание №1

## Условие

Преобразуете следующие двоичные числа, представленные в дополнительном коде, в десятичные.

a)

b)

c)

d)

## Решение

1. Число имеет старшую 1, поэтому оно должно быть отрицательным. Чтобы найти его модуль, инвертируем все биты:

После чего добавляем 1 в инвертированное выражение:

Далее переводим выражение из 2-ной системы счисления в 10:

Отсюда ответ:

1. Число имеет старшую 1, поэтому оно должно быть отрицательным. Чтобы найти его модуль, инвертируем все биты:

После чего добавляем 1 в инвертированное выражение:

Далее переводим выражение из 2-ной системы счисления в 10:

Отсюда ответ:

1. Число имеет старшую 0, поэтому оно должно быть положительным. В связи с этим отсутствует необходимость инвертирования исходного выражения и прибавления 1. Переводим выражение из 2-ной системы счисления в 10:

Отсюда ответ:

1. Число имеет старшую 1, поэтому оно должно быть отрицательным. Чтобы найти его модуль, инвертируем все биты:

После чего добавляем 1 в инвертированное выражение:

Далее переводим выражение из 2-ной системы счисления в 10:

Отсюда ответ:

# Задание №2

## Условие

Преобразуйте следующие десятичные числа в 6-разрядные двоичные числа, представленные в дополнительном коде, и вычтите одно число из другого:

a) −

b) −

c) −

d) −

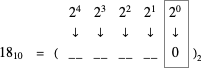
Укажите, если разность переполняет 6-битный регистр.

## Решение

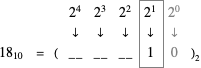
1. Переводим исходные числа из десятичной системы счисления в двоичную без учёта знака с помощью деления на основание новой системы счисления.

Работаем с числом :

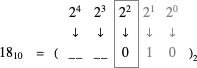
с остатком 0



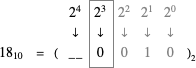
с остатком 1



с остатком 0



с остатком 0



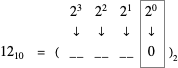
с остатком 1



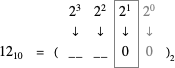
Получаем , добавляем 0 слева в связи с тем, что используем 6 разрядов -

Работаем с числом :

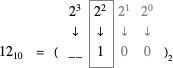
с остатком 0



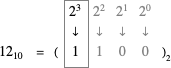
с остатком 0



с остатком 1



с остатком 1



Получаем , добавляем 0 слева в связи с тем, что используем 6 разрядов -

Формируем дополнительный код для отрицательных значений, в конкретном случае для :

Инвертируем все биты:

Добавляем 1 в инвертированное выражение:

Складываем полученные результаты:

Т.к. используется только 6 регистров, то получаем

Выполняем проверку на переполнение регистра. Переводим ответ из двоичной системы счисления в десятичную:

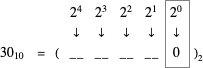
Т.к. в регистре 6 разрядов, то он может хранить значения от до т.е. от до , а именно от -32 до 31.

переполнения регистра не происходит.

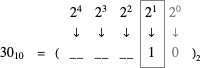
1. Переводим исходные числа из десятичной системы счисления в двоичную без учёта знака с помощью деления на основание новой системы счисления.

Работаем с числом :

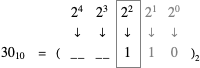
с остатком 0



с остатком 1



с остатком 1



с остатком 1



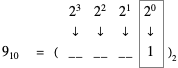
с остатком 1



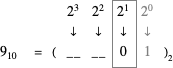
Получаем , добавляем 0 слева в связи с тем, что используем 6 разрядов -

Работаем с числом :

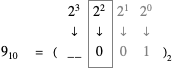
с остатком 1



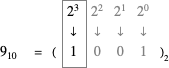
с остатком 0



с остатком 0



с остатком 1



Получаем , добавляем 0 слева в связи с тем, что используем 6 разрядов -

Формируем дополнительный код для отрицательных значений, в конкретном случае для :

Инвертируем все биты:

Добавляем 1 в инвертированное выражение:

Складываем полученные результаты:

Т.к. используется только 6 регистров, то получаем

Выполняем проверку на переполнение регистра. Переводим ответ из двоичной системы счисления в десятичную:

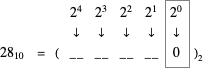
Т.к. в регистре 6 разрядов, то он может хранить значения от до т.е. от до , а именно от -32 до 31.

переполнения регистра не происходит.

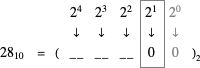
1. Переводим исходные числа из десятичной системы счисления в двоичную без учёта знака с помощью деления на основание новой системы счисления.

Работаем с числом :

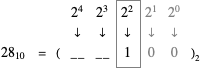
с остатком 0



с остатком 0



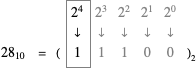
с остатком 1



с остатком 1



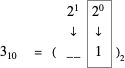
с остатком 1



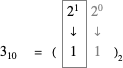
Получаем , добавляем 0 слева в связи с тем, что используем 6 разрядов -

Работаем с числом :

с остатком 1



с остатком 1



Получаем , добавляем 0 слева в связи с тем, что используем 6 разрядов -

Формируем дополнительный код для отрицательных значений.

Для :

Инвертируем все биты:

Добавляем 1 в инвертированное выражение:

Для :

Инвертируем все биты:

Добавляем 1 в инвертированное выражение:

Складываем полученные результаты:

Т.к. используется только 6 регистров, то получаем

Выполняем проверку на переполнение регистра. В отличии от предыдущих результатов, нынешний является отрицательным, поэтому необходимо осуществить инверсию и прибавление 1.

Инвертируем полученный ответ:

Прибавляем 1 к инвертированному результату:

Переводим из двоичной системы счисления в десятичную:

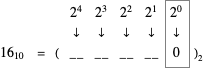
Т.к. в регистре 6 разрядов, то он может хранить значения от до т.е. от до , а именно от -32 до 31.

переполнения регистра не происходит.

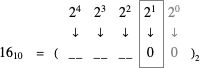
1. Переводим исходные числа из десятичной системы счисления в двоичную без учёта знака с помощью деления на основание новой системы счисления.

Работаем с числом :

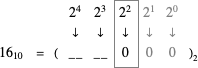
с остатком 0



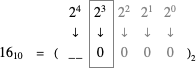
с остатком 0



с остатком 0



с остатком 0



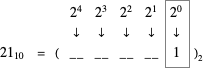
с остатком 1



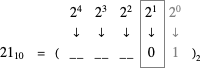
Получаем , добавляем 0 слева в связи с тем, что используем 6 разрядов -

Работаем с числом :

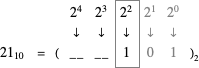
с остатком 1



с остатком 0



с остатком 1



с остатком 0



с остатком 1



Получаем , добавляем 0 слева в связи с тем, что используем 6 разрядов -

Формируем дополнительный код для отрицательных значений.

Для :

Инвертируем все биты:

Добавляем 1 в инвертированное выражение:

Для :

Инвертируем все биты:

Добавляем 1 в инвертированное выражение:

Складываем полученные результаты:

Т.к. используется только 6 регистров, то получаем

Выполняем проверку на переполнение регистра.

Инвертируем полученный ответ:

Прибавляем 1 к инвертированному результату:

Переводим из двоичной системы счисления в десятичную:

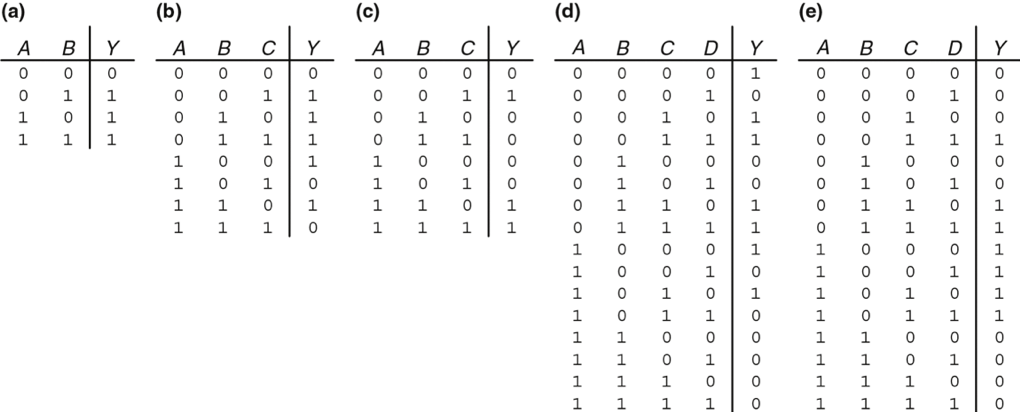
Т.к. в регистре 6 разрядов, то он может хранить значения от до т.е. от до , а именно от -32 до 31.

не входит в тот диапазон, происходит переполнение регистра!

# Задание №3

## Условие

Запишите логическое выражение в совершенной дизъюнктивной нормальной̆ форме для всех таблиц истинности, приведенных ниже и минимизируйте их.



## Решение

1. Строим СДНФ по таблице истинности:

Минимизируем СДНФ путём применения булевых теорем для нескольких переменных.

Применение теоремы склеивания:

Применение теоремы поглощения:

Таким образом,

1. Строим СДНФ по таблице истинности:

Минимизируем СДНФ путём применения карт Карно:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 0 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |

Получаем выражение C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 0 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |

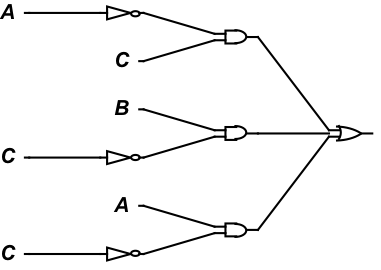
Получаем выражение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 0 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |

Получаем выражение

Путём суммирования всех выражений формируем минимальное ДНФ:

Представление полученного выражения в виде логической схемы:



1. Строим СДНФ по таблице истинности:

Минимизируем СДНФ путём применения карт Карно:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |

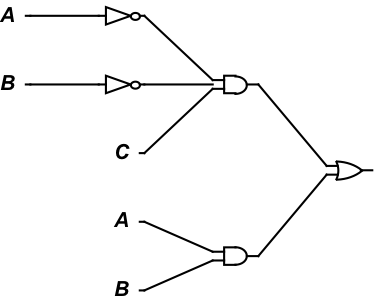
Получаем выражение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |

Получаем выражение

Путём суммирования всех выражений формируем минимальное ДНФ:

Представление полученного выражения в виде логической схемы:



1. Строим СДНФ по таблице истинности:

Минимизируем СДНФ путём применения карт Карно:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 00 | 1 | 0 | 1 | 1 |
| 01 | 0 | 0 | 1 | 1 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 1 | 0 | 0 | 1 |

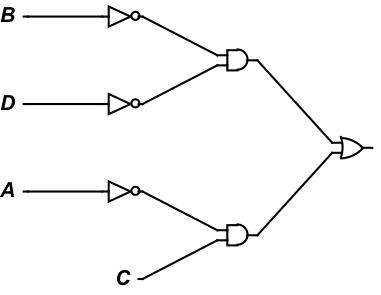
Получаем выражение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 00 | 1 | 0 | 1 | 1 |
| 01 | 0 | 0 | 1 | 1 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 1 | 0 | 0 | 1 |

Получаем выражение

Путём суммирования всех выражений формируем минимальное ДНФ:

Представление полученного выражения в виде логической схемы:



1. Строим СДНФ по таблице истинности:

Минимизируем СДНФ путём применения карт Карно:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 1 | 0 |
| 01 | 0 | 0 | 1 | 1 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 1 | 1 | 1 | 1 |

Получаем выражение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 1 | 0 |
| 01 | 0 | 0 | 1 | 1 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 1 | 1 | 1 | 1 |

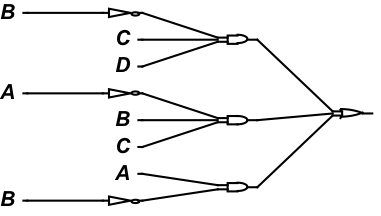
Получаем выражение

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 1 | 0 |
| 01 | 0 | 0 | 1 | 1 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 1 | 1 | 1 | 1 |

Получаем выражение

Путём суммирования всех выражений формируем минимальное ДНФ:

Представление полученного выражения в виде логической схемы:



# Задание №4

## Условие

Вы проектируете конечный автомат для отслеживания настроение четырёх студентов, работающих в лаборатории по проектированию цифровых схем. У студентов может быть следующие настроения: СЧАСТЛИВЫЙ (если схема работает), ГРУСТНЫЙ (если схема сгорела), ЗАНЯТЫЙ (работает над схемой), ЗАГРУЖЕННЫЙ (думает над схемой), СПЯЩИЙ (спит на рабочем месте). Сколько состояний будет у вашего автомата? Какое минимальное количество битов состояний необходимо для кодирования состояния автомата?

## Решение

Всего существует 5 состояний (счастливый, грустный, занятый, загруженный, спящий) и 4 студента, следовательно, всего у конечного автомата = 625 состояний. Находим подходящее минимальное количество для кодирования всех состояний конечного автомата:

< 625 <

512 < 625 < 1024

Следовательно, минимально требуется 10 бит для кодирования всех состояний автомата.

# Список литературы

1. Харрис, Д.М. Цифровая схемотехника и архитектура компьютера / Д.М. Харрис, С.Л. Харрис ; пер. с англ. Imagination Technologies. - Москва : ДМК Пресс, 2018. - 792 с. - ISBN 978-5-97060-570-7. - Текст : электронный. - URL: http://znanium.com/catalog/product/1032279
2. Схемотехника ЭВМ: Учебное пособие / Лехин С.Н. - СПб:БХВ-Петербург, 2010. - 663 с. ISBN 978-5-9775-0353-2 - Режим доступа: http://znanium.com/catalog/product/350620
3. Пуховский, В.Н. Электротехника, электроника и схемотехника. Модуль «Цифровая схемотехника» : учеб. пособие / В.Н. Пуховский, М.Ю. Поленов ; Южный федеральный университет. - Ростов-на-Дону ; Таганрог : Издательство Южного федерального университета, 2018. - 163 с. - ISBN 978-5-9275-3079-3. - Режим доступа: http://znanium.com/catalog/product/1039797
4. Архитектура компьютеров: Учебник / Буза М.К. - Мн.:Вышэйшая школа, 2015. - 414 с.: ISBN 978-985-06-2652-3 - Режим доступа: http://znanium.com/catalog/product/1011033
5. Архитектура компьютера / Догадин Н.Б., - 3-е изд., (эл.) - М.:БИНОМ. ЛЗ, 2015. - 274 с.: ISBN 978-5-9963-2638-9 - Режим доступа: http://znanium.com/catalog/product/539585