МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Московский технический университет связи и информатики»

(МТУСИ)

Кафедра «Математическая кибернетика и информационные технологии»

(МКиИТ)

## ЛАБОРАТОРНАЯ РАБОТА № 2

# по дисциплине

# «Основы кибернетики и вычислительной техники»

## Выполнила:

## Студент группы: УБВТ2403

## ФИО:Арзуманян А.К.

## Москва, 2024

## Компьютерная арифметика.

## Цель: научиться строить и анализировать сумматоры, «вычитатор» и «умножатор» в программе Logisim. Исследовать восьмибитное число, используя сдвиги.

## Сумматор двух двухбитных чисел:

## Задания:

## 1.Сумматор двух двухбитных чисел:

## Составить таблицу истинности.

## Минимизировать с помощью карты Карно.

## Построить в Logisim.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| a | b | C0 | S | C1 |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

## Для Sum (S):

## 

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| C0 \ ab | 00 | 01 | 10 | 11 |
| 0 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 |

## Для CarryOut (C):

## 

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| C0 \ ab | 00 | 01 | 10 | 11 |
| 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 |

## Построение сумматоров:

## Полусумматор: использовать элементы И и Исключающее ИЛИ, обозначить входы (A, B) и выходы (Sum, CarryOut)

## 

## Полный сумматор: использовать два полусумматора и элемент ИЛИ, указать CarryInput, Sum, CarryOut.

## 

## 8-битный сумматор: соединить полные сумматоры, указать CarryInput, CarryOut.

## 

## 

## Инвертор:

## Использовать Исключающие ИЛИ для подачи на них входного значения и значения инвертирования

## 

## «Вычитатор»:

## Подать входные значения A и B.

## Использовать инвертор, 8-битный сумматор, контакт для вычитания и Исключающее ИЛИ

## 

## «Умножатор» двух четырехбитных чисел:

## Использовать частичные произведения и 4-битные сумматоры.

## Реализовать схему умножения.

## 

## Исследование числа посредством сдвигов:

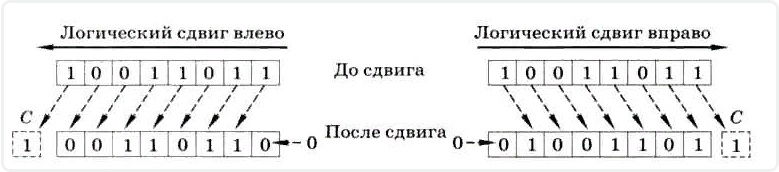
## Изучить логический, арифметический и циклический сдвиги.

## Исследовать двоичное число 10000001 .

## Представить результаты в беззнаковом и знаковом десятичном виде.

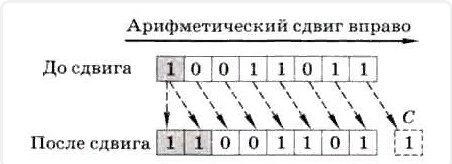
## Теория сдвиги:

Сдвиги незаменимы тогда, когда требуется проделать ту или иную обработку каждого бита, входящего в число. Сдвиги двоичного числа позволяют быстро умножить или разделить число на степени двойки: 2, 4, 8 и т. д.



**Логический сдвиг влево на 1 разряд увеличивает целое положительное число вдвое, а сдвиг вправо делит на 2 нацело.**

Сдвиг вправо отрицательных чисел, позволяет их делить. При этом старший разряд надо заполнять не нулём, а единицей! **Это** **арифметический сдвиг**.



## Перевод в десятичную систему 10000001

## 2^7 +2^0 = 128 + 2 + 1= 128 + 1= 129

## Логический сдвиг

## *Логический сдвиг двигает все биты числа влево или вправо, заполняя пустые позиции нулями.*

## *Левый логический сдвиг на 1:*

## Исходное: 10000001 (129)

## Результат: 00000010 (2 в десятичной системе).

## *Правый логический сдвиг на 1:*

## Исходное: 10000001 (129)

## Результат: 01000000 (64 в десятичной системе).

## Арифметический сдвиг

## *Арифметический сдвиг тоже может двигать биты влево или вправо, но при правом сдвиге старший бит, который отвечает за знак, копируется.*

## Левый арифметический сдвиг на 1:

## Результат: 00000010 (2 в десятичной системе).

## 

## Правый арифметический сдвиг на 1:

## Если рассматривать число как беззнаковое - результат: 01000000 (64).

## Если рассматривать как знаковое, то результат также будет 01000000 (64), так как это положительное число.

## Циклический сдвиг

## *Циклический сдвиг сдвигает биты числа и переносит "выдвинутые" биты в другую сторону.*

## Циклический сдвиг влево на 1:

## Исходное: 10000001 (129)

## Результат: 00000011 (3 в десятичной системе).

## Циклический сдвиг вправо на 1:

## Исходное: 10000001 (129)

## Результат: 11000000 (192 в десятичной системе).

## Результаты:

## Левый логический сдвиг: 2 (б/з)

## Правый логический сдвиг: 64 (б/з)

## Левый арифметический сдвиг: 2 (б/з)

## Правый арифметический сдвиг: 64 (б/з, 64 для знакового тоже)

## Циклический сдвиг влево: 3 (б/з)

## Циклический сдвиг вправо: 192 (б/з)

## Беззнаковый десятичный вид: 129

## Знаковый десятичный вид: 129