**Лабораторная работа №6**

**Изучение логического элемента сумматор с использованием программы Logisim**

**Цель работы: Изучить принцип действия сумматора, построить различные модели сумматоров в программе Logisim**

**Теоретическая часть:**

**Полный одноразрядный сумматор.**

**Связь между двоичной арифметикой и алгеброй логики позволяет реализовать логические схемы основных элементов процессора и памяти компьютера.**

**Сумматор - это устройство, предназначенное для сложения двоичных чисел.**

**Рассмотрим сначала более простое устройство – полусумматор.**

**Построим таблицу истинности для устройства реализующего арифметическую операцию сложения. Операция «+» бинарная, поэтому полусумматор должен иметь два входа (A и B). В результате сложения двух одноразрядных двоичных чисел может получиться двухразрядное число (с переносом в следующий разряд). Значит, устройство должно иметь два выхода (P - перенос в следующий разряд, S - результат, остающийся в текущем разряде).**

|  |  |  |  |
| --- | --- | --- | --- |
| **A** | **B** | **P** | **S** |
| **0** | **0** | **0** | **0** |
| **0** | **1** | **0** | **1** |
| **1** | **0** | **0** | **1** |
| **1** | **1** | **1** | **0** |

**По данной таблице истинности построим СДНФ (см.**[**алгоритм построения СДНФ**](http://informatics-lesson.ru/logic/disjunctive-conjunctive-forms.php)**):**

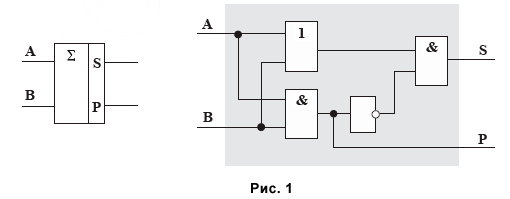
**Для переноса в старший разряд: P = A ∧ B**

**Для текущего разряда: S = ¬ A ∧ B ∨ A ∧ ¬ B**

**Преобразуем логическую формулу для S:  
(¬ A • B) + (A • ¬ B) = (¬ A • A) + ( ¬ A • B) + (A • ¬ B) + (¬ B • B) =  
= ¬ A • (A + B) + ¬ B • (A + B) = (A + B) • ¬ (A • B)**

**С учетом формулы для переноса имеем:  
S = (A + B) • ¬ (A • B) = (A + B) • ¬ P**

**Таким образом, полусумматор можно построить, используя четыре**[**простейших логических элемента**](http://informatics-lesson.ru/logic/logic-elements.php)**: два конъюнктора, дизъюнктор и инвертор (см. рис.1, слева показано условное обозначение полусумматора):**

****

**Итак, получено устройство, реализующее суммирование одноразрядных двоичных чисел без учета переноса из младшего разряда.**

**Для реализации полного одноразрядного сумматора необходимо учесть перенос из младшего разряда (P0). Поэтому сумматор должен иметь три входа. Построим таблицу истинности для устройства с учетом третьего входа:**

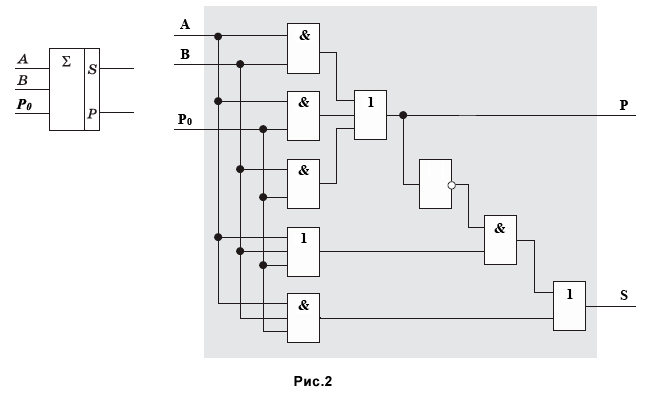
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **A** | **B** | **P0** | **P** | **S** |
| **0** | **0** | **0** | **0** | **0** |
| **0** | **0** | **1** | **0** | **1** |
| **0** | **1** | **0** | **0** | **1** |
| **0** | **1** | **1** | **1** | **0** |
| **1** | **0** | **0** | **0** | **1** |
| **1** | **0** | **1** | **1** | **0** |
| **1** | **1** | **0** | **1** | **0** |
| **1** | **1** | **1** | **1** | **1** |

**Построим СДНФ для выхода P (перенос в старший разряд):  
P =(¬ A ∧ B ∧ P0) ∨ (A ∧ ¬ B ∧ P0) ∨ (A ∧ B ∧ ¬ P0) ∨ (A ∧ B ∧ P0)  
Преобразуем:  
1) (A ∧ B ∧ ¬ P0) ∨ (A ∧ B ∧ P0) = (A ∧ B) ∧ (¬ P0 ∨ P0) = A ∧ B  
Имеем, P = (¬ A ∧ B ∧ P0) ∨ (A ∧ ¬ B ∧ P0) ∨ (A ∧ B)  
2) (¬ A ∧ B ∧ P0) ∨ (A ∧ B) = B ∧(¬ A ∧ P0 ∨ A) = B ∧ (¬ A ∨ A ) ∧ (P0 ∨ A) =  
= B ∧ (P0 ∨ A) = (B ∧ P0) ∨ (A ∧ B)  
Имеем, P = (A ∧ ¬ B ∧ P0) ∨ (B ∧ P0) ∨ (A ∧ B)  
3) (A ∧ B) ∨ (A ∧ ¬ B ∧ P0) = A ∧ (B ∨ ¬ B ∧ P0) = A ∧ (B ∨ ¬ B)(B ∨ P0) =  
= A ∧ (B ∨ P0) = (A ∧ B) ∨ (A ∧ P0)  
Таким образом, для переноса в старший разряд получили:  
P = A ∧ B ∨ A ∧ P0 ∨ B ∧ P0**

**Проанализируем таблицу истинности для выхода S. Значение S отлично от нуля в том случае, если единица поступает ровно на один вход (при этом на двух других входах фиксируется ноль), или на все три входа сразу, т. е.:  
S = ¬ (A ∧ B ∨ A ∧ P0 ∨ B ∧ P0) ∧ (A ∨ B ∨ P0) ∨ (A ∧ B ∧ P0)**

**С учетом формулы для переноса в старший разряд, имеем:  
S = ¬ P ∧ (A ∨ B ∨ P0) ∨ (A ∧ B ∧ P0)**

**Таким образом, одноразрядный двоичный сумматор можно реализовать с помощью следующей схемы (см. рис. 2, слева показано условное обозначение сумматора), которая соответствует полученным логическим формулам (1) и (2).**

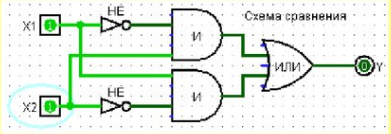
****

**Заметим, что логические функции P и S можно выразить с помощью других формул. В таком случае для одноразрядного двоичного сумматора потребуется другая логическая схема.**

**Выполнение работы:**

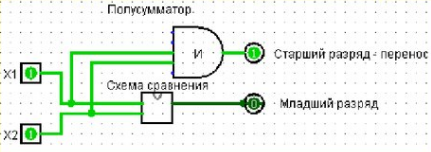
**Начнем моделирование со схемы полусумматора.**

**Построить в Logisim Схему сравнения**

****

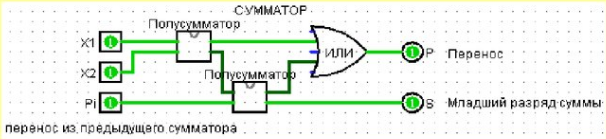
**С помощью команды Проект - Анализировать схему получить таблицу истинности Схемы сравнения.**

**С помощью инструмента «Добавть схему» добавить схему «Полусумматор»**

****

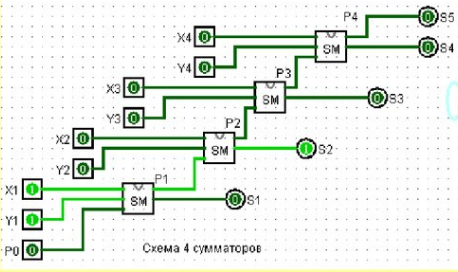
**С помощью команды Проект - Анализировать схему получить таблицу истинности Полусумматора.**

**Используя «Полусумматор» построить схему «Полного сумматора»**

****

**С помощью команды Проект - Анализировать схему получить таблицу истинности Сумматора.**

**Построить схему из 4х сумматоров, которые позволяют складывать 2 четырехразрядных числа**

****

**Здесь, соответственно X1 и Y1 - слагаемые первого сумматора, X2 и Y2 - второго и т.д. S1-S5 - младший разряд суммы, Р1-Р4 - перенос, старший разряд соответствующего сумматора, Р0 всегда равно 0, так как в первом сумматоре всегда складываются двоичные числа и переноса там нет. На изображенном в схеме примере складываются 0001 +0001 = 00010**

**X4 X3 X2 X1**

**+ Y4 Y3 Y2 Y1**

**S5 S4 S3 S2 S1**

**Проверить на другом числе.**