**Исследование принципа работы мультиплексора**

1 Введение

Мультиплексором называют коммутатор сигналов с нескольких входов на один выход. Назначение мультиплексоров – коммутировать в нужном порядке входную информацию, поступающую с нескольких шин, на одну выходную. Мультиплексор имеет несколько информационных входов, адресные входы, вход для подачи стробирующего сигнала и один выход.

2 Синтез мультиплексора

Основу мультиплексора составляет линейный дешифратор, выходы которого объединены по ИЛИ. Логическая функция, которую реализует мультиплексор:



Собственно мультиплексор выполнен на четырех логических ячейках И, выполняющих роль дешифратора, и одной ячейки ИЛИ, логически суммирующей выходные сигнала дешифратора.

Адрес выбранного направления определяется управляющими сигналами. Синхросигнал Е управляет работой ячеек в дешифраторе. При Е=0 работа дешифратора запрещена, на выходе F устанавливается *лог* 0. При Е=1 ячейки дешифратора активизированы. Информация проходит на выход с того входа, двоичный код которого указан на адресных входах.

В данной работе цифровой код поочередно перебирает все комбинации двоичных переменных на адресных входах, состояние на выходе последовательно повторяет состояние всех его информационных входов.

Для последовательного перебора всех комбинаций переменных на адресных входах используются JK-триггеры, работающие в режиме переключения (на входы J и K подается логическая 1), т.е. в качестве счетчиков.

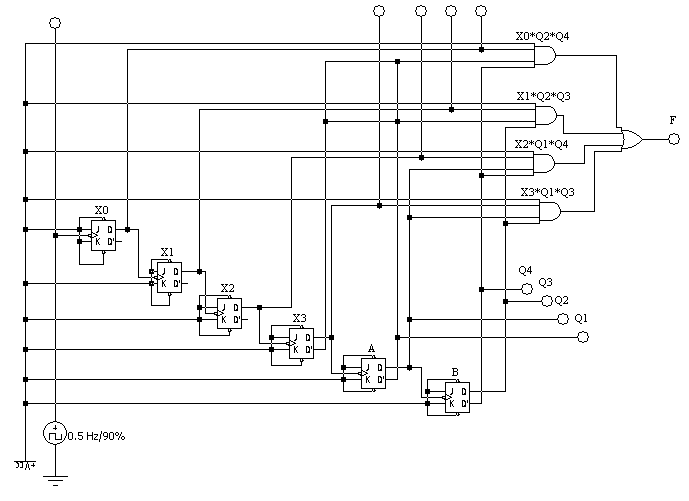


Рисунок 2.1 – Мультиплексор

В данном случае роль адресных входов А и В играют выходы триггеров Q1 и Q3, а А и В – Q2 и Q4 соответственно.

C

X0

X1

X2

X3

Q1

Q3

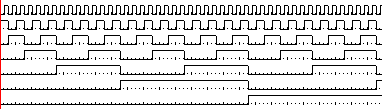


Рисунок 2.2 – Показания логического анализатора

3 Заключение

Мультиплексоры удобно использовать для реализации логических функций, записанных непосредственно в СДНФ. Любую булеву функцию двух переменных можно реализовать с помощью четырехканального мультиплексора.

**5. Сумматор**

**Цель работы**

Целью настоящей работы является научить студентов самостоятельно проводить анализ проектируемых схем. А так же создание сумматора ± А ± В, который будет показывать сумму двух чисел на семисегментных индикаторах.

**1.1. Теоретические предпосылки**

**Кодирование отрицательных чисел**

Для выполнения арифметических операций двоичные числа кодируются специальными *машинными кодами: прямыми, дополнительными и обратными,* позволяющими заменить операции *вычитания* операциями *суммирования,* что упрощает построение арифметическо-логических устройств.

**Модифицированные коды**

В отличие от обычных машинных кодов в модифицированных кодах под знак числа отводится два разряда: плюс изображается *двумя нулями,* а минус -*двумя единицами.* Это весьма удобно для выявления переполнения разрядной сетки, которое может получиться при сложении чисел с одинаковыми знаками.

*Пример.* Числа А = +0.10101 и В = - 0.1010 представить в обратном и дополнительном кодах

Aмпр= 00.10101, Амобр = 00.10101, Амдоп = 00.10101;

Вмпр=П.10101, Вмобр= 11.01010, Вмдоп= 11.01011.

***Сложение чисел в модифицированном обратном коде.*** Сложение осуществляется по правилам двоичной арифметики. Отличие состоит лишь в том, что единицу переноса из *старшего знакового* разряда (если она появляется) необходимо прибавить *к младшему разряду* суммы *{циклический перенос).*

*Пример.* Сложить в модифицированном обратном коде двоичные числа А и В при условии: А > 0, В < 0, (А + В) < 0. Переведем А и В в обратные модифицированные коды и произведем их сложение.

A=+1010 Апр=0.1010 Амобр =00.1010

В=-1410 Впр=1.1110 Вмобр =11.0001

А+В=-410 (А+В) мобр=11.1001

переведем результат в прямой код (А+В) мпр=11.0100

Сочетание двух единиц в знаковых разрядах показывает, что знак суммы отрицательный: (А+В)пр = 11.01002 = - 410.

*Сложение чисел в модифицированном дополнительном коде* осуществляется по правилам двоичной арифметики. Единица переноса, возникающая в старшем знаковом разряде суммы, *отбрасывается.* Знаковым разрядом числа является второй слева от запятой разряд; первый разряд служит для анализа переполнения разрядной сетки.

*Пример.* Сложить в модифицированном дополнительном коде двоичные числа А и В при условии: А > О, В < О, (А + В) < 0. Переведем А и В в дополнительные модифицированные коды и произведем их сложение.

A=+1110 Апр=0.1011 Амдоп =00.1011

В=-1510 Впр=1.1111 Вмдоп =11.0001

А+В=-410 (А+В) мдоп=11.1101

переведем результат в прямой код (А+В)пр=11.0100

Получилось число в дополнительном коде. Переведем его в прямой код, для чего инвертируем каждый разряд числа и к младшему разряду прибавим 1. Сочетание двух единиц в знаковых разрядах показывает, что знак суммы отрицательный: (А+В)пр = 11.01002 = - 410.

1. **Методика работы**

Схема реализована в программе Electronics Workbench Multisim 8.

Для проектирования восьмиразрядного сумматора используются стандартные элементы: SWITCH SPDT из вкладки Basic, которые были соединены и добавлены в новый созданный элемент A\_B (рис. 2.1 а); FULL-ADDER из вкладки Misc Digital, которые были последовательно соединены и созданы новые элементы S (рис. 2.1 в,г,д,ж); EOR2 из вкладки Misc Digital, которые были соединены и созданы новые элементы XOR; семисегментные DCD\_HEX\_DIG\_RED из вкладки Indicators для индикации чисел; символьные ALPHA\_NUMERIC\_COM\_A\_RED из вкладки Indicators для индикации знаков: «=», «+», «-»; индикаторы PROBE\_DIG\_BLUE из вкладки Indicators; стандартная шина BUS для соединения элементов, а так же 12 В источник питания.

На рисунке 2.1 представлена разработанная схема сумматора:

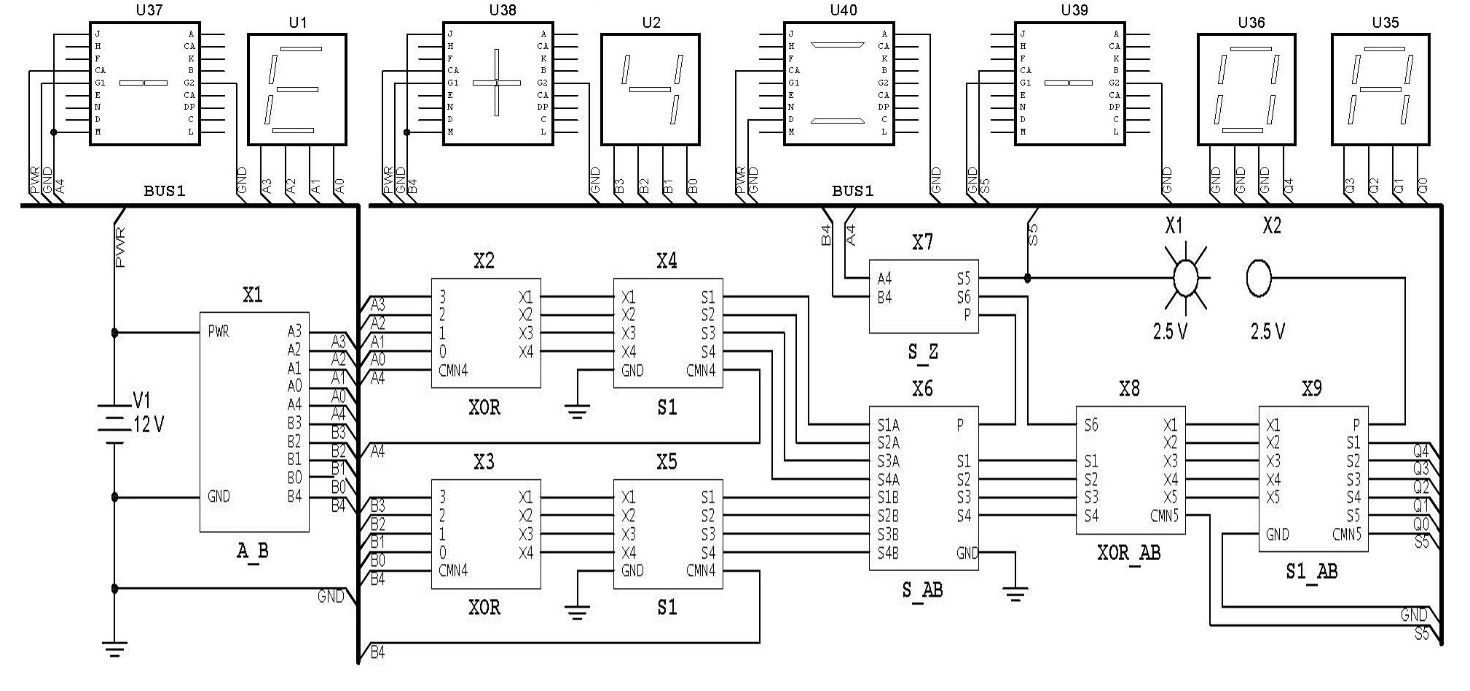
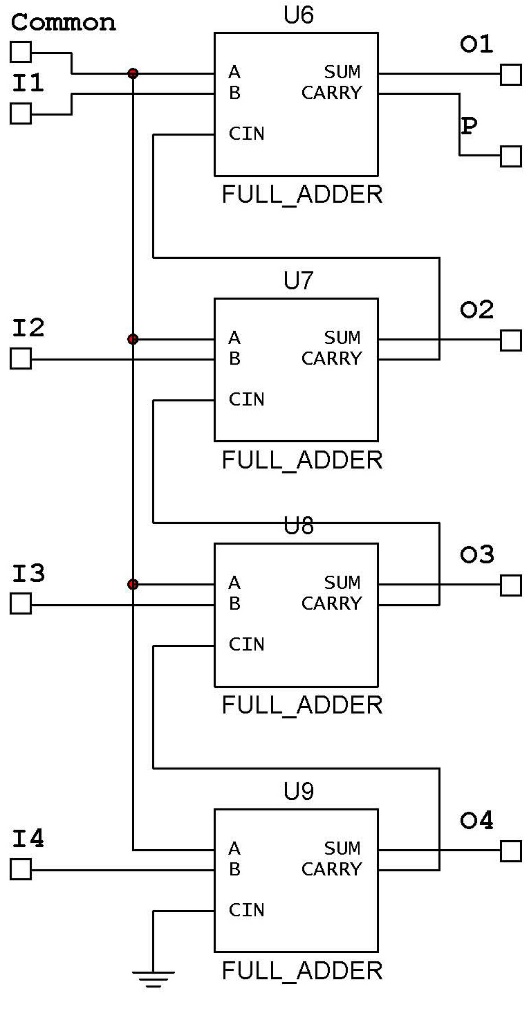
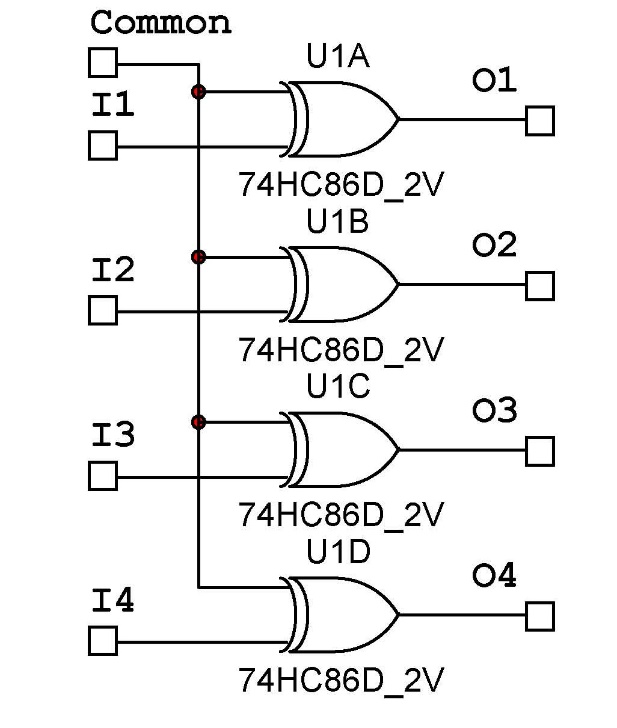
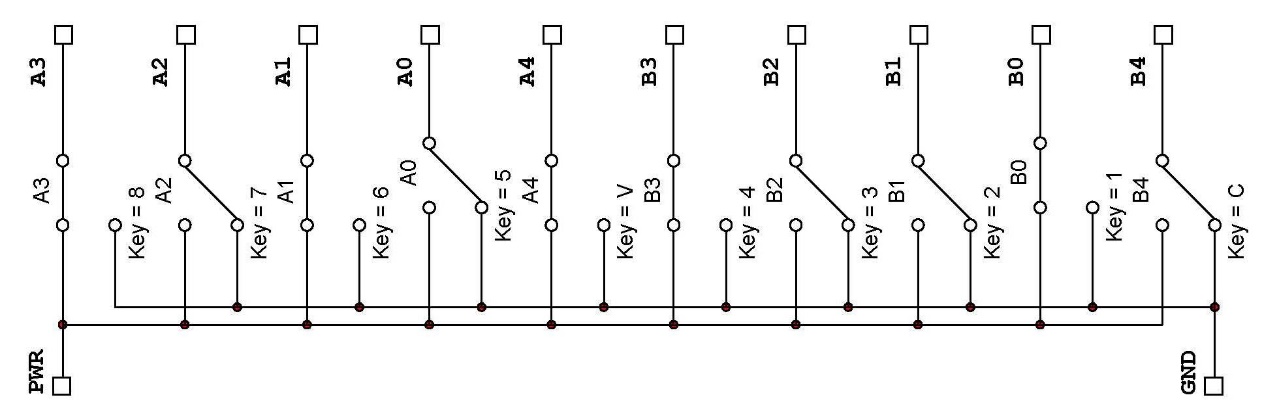


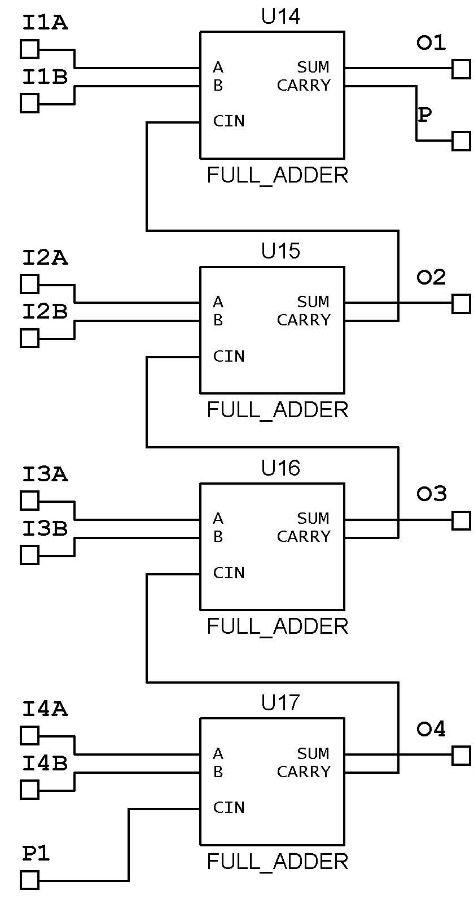
Рисунок 2.1 Разработанная схема сумматора.

Данная схема содержит несколько блоков.

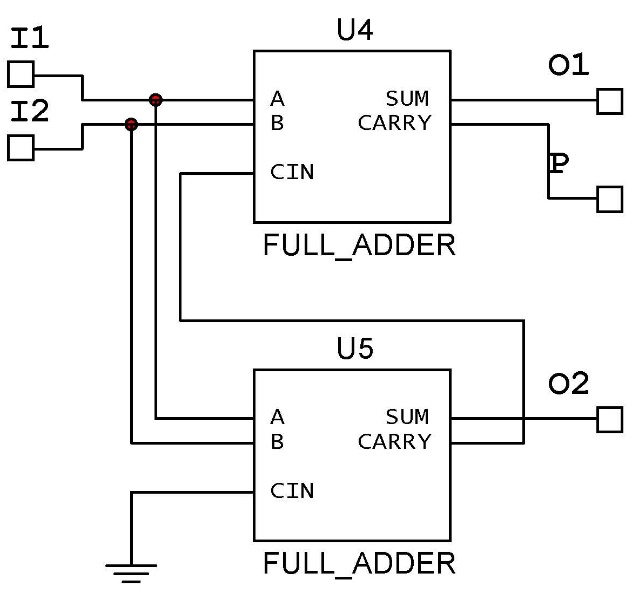
Блоки представлены на рисунке 2.2.

******

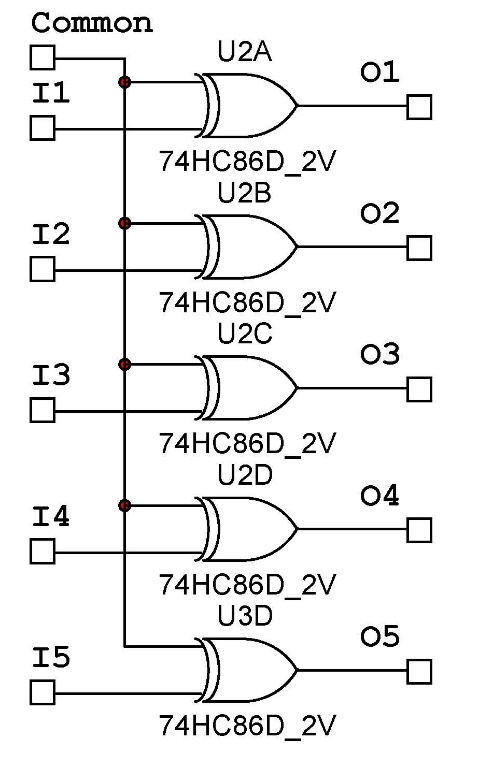
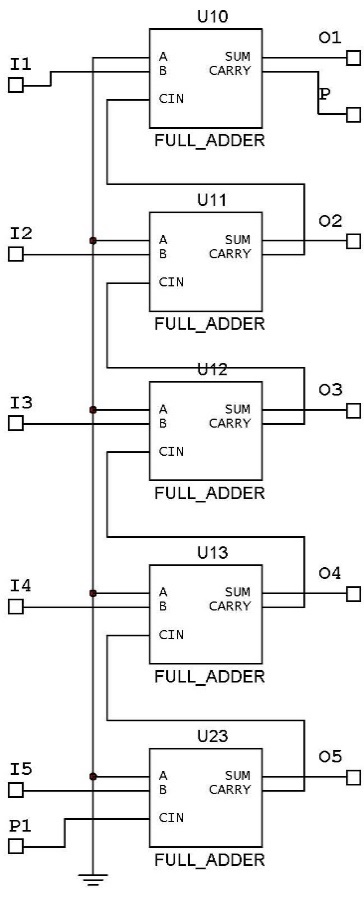
***а)***

***б)***

***в)***



***г) д)***



***е) ж)***

***Рисунок 2.1 Схемы блоков: KEY (а), XOR(б), S1(в), S\_AB(г), S\_Z(д), XOR\_AB(е), S1\_AB(ж).***

Принцип работы: пользователь с помощью размыкания или замыкания ключей формирует на выходе элемента A\_B двоичные кода двух чисел, затем каждый двоичный код числа переводиться из прямого в обратный элементом XOR, элементы S1 переводят двоичный код числа из обратного в двоичный, затем два числа складываются с помощью элемента S\_AB, а полученная сумма переводиться из дополнительного двоичного кода в прямой через элементы XOR\_AB, S1\_AB и результат выводиться на индикаторы. В порядке: сначала знак, десятки, потом единицы

Кнопки: 1, 2, 3, 4 - полубайт первого числа; 5, 6, 7, 8 - полубайт второго числа, С и V - знак первого и второго числа соответственно.

Например, чтобы сложить -4 и 6 нужно нажать: С, 3 и 6, 7. Сумматор сложит два числа и результат выведет в шеснадцатиричном виде -4+6=2(HEX).