

Министерство науки и высшего образования Российской Федерации
Южно-Уральский государственный университет
Кафедра «Информационно-измерительная техника»

004.3(07)

B858

Вставская Е.В., Константинов В.И., Константинова О.В.

МИКРОПРОЦЕССОРНАЯ ТЕХНИКА И КОМПЬЮТЕРЫ В ПРИБОРОСТРОЕНИИ

Учебное пособие

Челябинск
Издательский центр ЮУрГУ
2022

УДК 004.3(07) + 004.312(075.8)
В858

*Одобрено
учебно-методической комиссией
Высшей школы электроники и компьютерных наук*

*Рецензенты:
С.В. Колодий, С.И. Платов*

Вставская, Е.В.

В858 Микропроцессорная техника и компьютеры в приборостроении: учебное пособие / Е.В. Вставская, В.И. Константинов, О.В. Константинова – Челябинск: Издательский центр ЮУрГУ, 2022. – 64 с.

Учебное пособие предназначено для обучения студентов очной и заочной формы по дисциплине «Микропроцессорная техника и компьютеры в приборостроении» специальности «Приборостроение», а также по дисциплине «Микропроцессоры, микроконтроллеры и вычислительная техника» специальности «Управление в технических системах».

Приведены основные сведения о логических элементах, способах записи и минимизации функций алгебры логики. Даны рекомендации по составлению схем, а также оцениванию результатов моделирования цифровых схем. Для моделирования используется свободно распространяемая программа LTSpice.

УДК 004.3(07) + 004.312(075.8)

© Издательский центр ЮУрГУ, 2022

ОГЛАВЛЕНИЕ

Введение	
Общие положения	4
Основные логические законы	5
Логические элементы.....	6
Способы представления функций алгебры логики	16
Минимизация функций алгебры логики.....	18
Моделирование цифровых схем	20
Лабораторная работа 1. Схема равнозначности кодов.....	23
Лабораторная работа 2. Минимизация булевых функций.....	26
Лабораторная работа 3. Полные двоичные дешифраторы - демультиплексоры.....	31
Лабораторная работа 4. Мультиплексоры	35
Лабораторная работа 5. Преобразователи кодов	41
Лабораторная работа 6. Двоичные счетчики.....	46
Лабораторная работа 7. Проектирование кодированного цифрового автомата.....	52
Лабораторная работа 8. Арифметико-логическое устройство	59
Библиографический список.....	64

ВВЕДЕНИЕ

Общие положения

Логическая (булева) переменная – это такая величина x , которая может принимать только два значения: $x = \{0,1\}$.

Логическая функция (функция алгебры логики, ФАЛ) – это функция многих аргументов $f(x_{n-1}, x_{n-2}, \dots, x_0)$, принимающая на всех наборах логических переменных $x_{n-1}, x_{n-2}, \dots, x_0$ значения равные нулю или единице.

Под **логическими элементами** понимаются устройства, в которых обрабатывается или хранится логическая информация.

Операционные узлы вычислительной машины строятся на базе элементарных логических узлов (логических элементов).

Элементарные логические узлы предназначены для выполнения различных логических (функциональных) операций над дискретными сигналами при двоичном способе их представления.

Благодаря способности к микроминиатюризации средствами интегральной технологии преимущественное распространение получили логические элементы потенциального типа. В них для передачи значения логической переменной используются потенциальные сигналы на выходе элемента, причем в элементах **положительной логики** нулевому логическому значению переменной соответствует уровень низкого потенциала, а единичному логическому значению – уровень высокого потенциала. Характерной особенностью, выделяющей логические элементы из класса импульсных схем, является свойство восстановления логического уровня. Это свойство обусловлено особенностями передаточной характеристики логического элемента и позволяет однозначно определить уровень сигнала на выходе цепи из однотипных логических элементов, даже если входной сигнал окажется линейной комбинацией значений низкого и высокого уровней.

Основными логическими операциями являются:

- **конъюнкция И (AND) ($\cap$)** (логическое умножение): на выходе имеет логическую 1 только в том случае, если на все входы подается 1;
- **дизъюнкция ИЛИ (OR) ($\cup$)** (логическое сложение): на выходе имеет логическую 1, если хотя бы на один из входов подается 1;
- **инверсия НЕ (NOT) ($\neg$)** (логическое отрицание): устройство, имеющее один вход, логическое состояние выхода противоположно (комплементарно) логическому состоянию входа.

Основными логическими элементами являются **И, ИЛИ, НЕ**, а также их производные **И-НЕ, ИЛИ-НЕ, исключающее ИЛИ**.

Элементы **И-НЕ, ИЛИ-НЕ** являются **базисными**, поскольку на их основе возможно воспроизведение всех остальных логических функций. Базисность элементов инвертирующего типа обуславливается с одной стороны

симметричностью запасов статической помехоустойчивости при передаче низкого и высокого логических уровней, а с другой стороны одинаковостью времени распространения фронтов сигнала в цепи.

Основные логические законы

Ассоциативный (сочетательный) закон:

$$X2 \cup (X1 \cup X0) = (X2 \cup X1) \cup X0$$

$$X2 \cap (X1 \cap X0) = (X2 \cap X1) \cap X0$$

Можно опускать те скобки, которые определяют, в каком порядке должна выполняться конъюнкция или дизъюнкция.

Коммутативный (переместительный) закон:

$$X1 \cup X0 = X0 \cup X1$$

$$X1 \cap X0 = X0 \cap X1$$

Операнды коммутативных операций можно менять друг с другом местами.

Дистрибутивный (распределительный) закон:

$$X2 \cap (X1 \cup X0) = (X2 \cap X1) \cup (X2 \cap X0)$$

Свойство дистрибутивности одной операции относительно другой позволяет «раскрывать» скобки аналогично процедуре из элементарной алгебры.

Законы де Моргана (законы общей инверсии или дуальности):

$$\overline{X1 \cap X0} = \overline{X1} \cup \overline{X0} \qquad X1 \cup X0 = \overline{\overline{X1} \cap \overline{X0}}$$

$$\overline{X1 \cup X0} = \overline{X1} \cap \overline{X0} \qquad X1 \cap X0 = \overline{\overline{X1} \cup \overline{X0}}$$

Закон поглощения (элиминации):

$$X1 \cap (X1 \cup X0) = X1$$

$$X1 \cup (X1 \cap X0) = X1$$

Если к выражению применяется с одним и тем же операндом сначала одна операция, а потом, с тем же самым операндом, поглощающая ее, то значение выражения поглощается, становясь равно операнду.

Закон склеивания (исключения):

$$(X1 \cap X0) \cup (X1 \cap \overline{X0}) = X1$$

$$(X1 \cup X0) \cap (X1 \cup \overline{X0}) = X1$$

Свойства единицы и нуля:

$$X \cap 1 = X$$

$$X \cup 1 = 1$$

$$X \cap 0 = 0$$

$$X \cup 0 = X$$

**Закон
идемпотентности:**

$$X \cap X = X$$

$$X \cup X = X$$

Закон дополнения:

$$X \cup \bar{X} = 1$$

$$X \cap \bar{X} = 0$$

**Закон двойного
отрицания:**

$$\bar{\bar{X}} = X$$

Логические элементы

Элемент И (AND) (логическое умножение): на выходе имеет логическую 1 только в том случае, если на все входы подается 1. Обозначение элемента представлено на рис. 1. Условное графическое обозначение элемента И представлено на рис. 1.

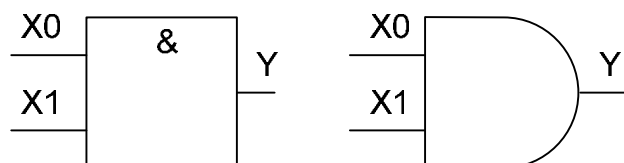


Рис. 1. Условное графическое обозначение И

В таблице 1 представлена таблица истинности элемента И

Таблица 1

Таблица истинности 2-входового элемента И

X1	X0	Y
0	0	0
0	1	0
1	0	0
1	1	1

Логический элемент И имеет несколько входов и один выход и выполняет операцию логического умножения (конъюнкции):

$$Y = X1 \cap X0$$

$$Y = Xn \cap \dots \cap X1 \cap X0$$

Функция $Y=0$ если хотя бы один из ее аргументов равен нулю, и $Y=1$ если все аргументы равны 1.

Наращивание размерности элемента И производится в соответствии с ассоциативным (сочетательным) законом:

$$X2 \cap X1 \cap X0 = X2 \cap (X1 \cap X0)$$

$$X3 \cap X2 \cap X1 \cap X0 = (X3 \cap X2) \cap (X1 \cap X0)$$

по следующим схемам (рис. 2).

Для сокращения размерности элемента И рассмотрим таблицу истинности 3-входового элемента, где серым цветом выделена таблица истинности для 2-входового элемента (табл. 2).

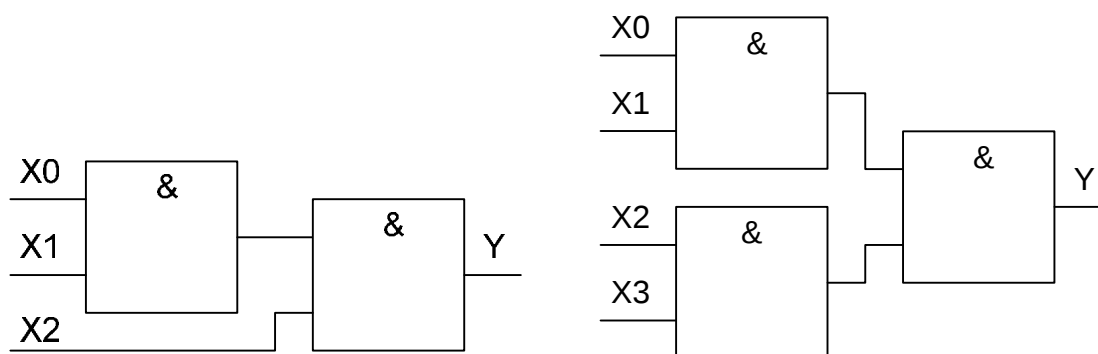


Рис. 2. Нарращивание размерности элемента И

Как видим, для выделенной части таблицы переменная X_2 во всех случаях принимает единичное значение. Следовательно, неиспользуемые входы элемента И можно подключить к логической единице.

Таблица 2

Таблица истинности 3-входового элемента И

X_2	X_1	X_0	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Другим способом сокращения размерности элемента И будет соединение неиспользуемого входа с одним из используемых. На примере таблицы истинности 3-входового элемента рассмотрим соединение входов X_2 и X_1 :

Таблица 3

Таблица истинности 3-входового элемента И

X_2	X_1	X_0	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

В таблице 3 цветом выделены строчки, в которых $X_2 = X_1$. В результате из выделенных ячеек получим таблицу истинности для 2-входового элемента.

Аналогичным образом можно получить таблицу истинности для 2-входового элемента если объединить неиспользуемый вход X_2 со входом X_0 .

Способы сокращения размерности элемента И представлены на рис. 3.

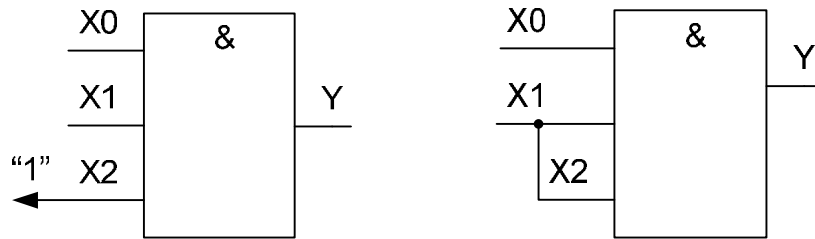


Рис. 3. Сокращение размерности элемента И

Элемент ИЛИ (OR) (логическое сложение): на выходе имеет логическую 1, если хотя бы на один из входов подается 1. Условное графическое обозначение элемента ИЛИ представлено на рис. 4.

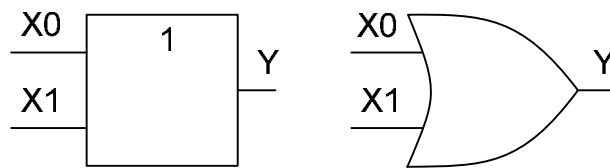


Рис. 4. Условное графическое обозначение элемента ИЛИ

В таблице 5 представлена таблица истинности элемента ИЛИ

Таблица 4

Таблица истинности 2-входового элемента ИЛИ

X1	X0	Y
0	0	0
0	1	1
1	0	1
1	1	1

Логический элемент ИЛИ имеет несколько входов и один выход и выполняет операцию логического сложения (дизъюнкции):

$$Y = X1 \cup X0$$

$$Y = Xn \cup \dots \cup X1 \cup X0$$

Функция $Y=1$ если хотя бы один из ее аргументов равен 1, и $Y=0$ если все аргументы равны 0.

Наращивание размерности элемента ИЛИ производится в соответствии с ассоциативным (сочетательным) законом:

$$X2 \cup X1 \cup X0 = X2 \cup (X1 \cup X0)$$

$$X3 \cup X2 \cup X1 \cup X0 = (X3 \cup X2) \cup (X1 \cup X0)$$

по следующим схемам (рис. 5).

Для сокращения размерности элемента ИЛИ рассмотрим таблицу истинности 3-входового элемента, где серым цветом выделена таблица истинности для 2-входового элемента.

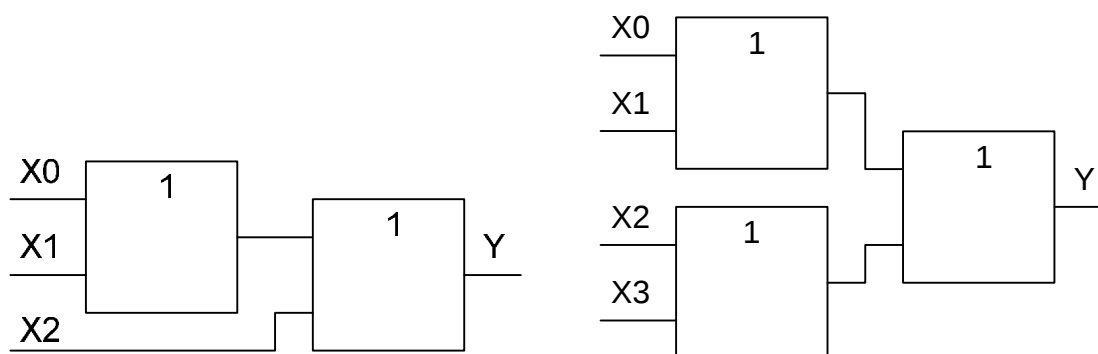


Рис. 5. Нарастивание размерности элемента ИЛИ

Как видим, для выделенной части таблицы переменная X_2 во всех случаях принимает нулевое значение. Следовательно, неиспользуемые входы элемента ИЛИ можно подключить к логическому нулю.

Таблица 5

Таблица истинности 3-входового элемента ИЛИ

X_2	X_1	X_0	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Другим способом сокращения размерности элемента ИЛИ будет соединение неиспользуемого входа с одним из используемых. На примере таблицы истинности 3-входового элемента рассмотрим соединение входов X_2 и X_1 .

Таблица 6

Таблица истинности 3-входового элемента ИЛИ

X_2	X_1	X_0	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

В таблице цветом выделены строчки, в которых $X_2 = X_1$. В результате из выделенных ячеек получим таблицу истинности для 2-входового элемента.

Аналогичным образом можно получить таблицу истинности для 2-входового элемента если объединить неиспользуемый вход X_2 со входом X_0 .

Способы сокращения размерности элемента ИЛИ представлены на рис. 6.

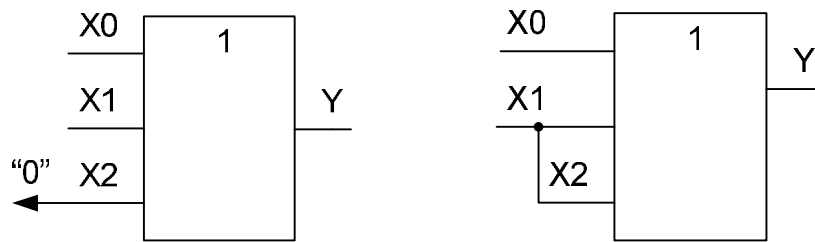


Рис. 6. Сокращение размерности элемента ИЛИ

Элемент НЕ (NOT) (логическое отрицание) – инвертор: 1-входовое устройство, логическое состояние выхода которого противоположно (комплементарно) логическому состоянию входа. Условное графическое обозначение элемента представлено на рис. 7.

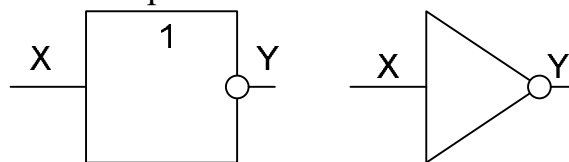


Рис. 7. Условное графическое обозначение элемента НЕ

Логический элемент НЕ имеет один вход и один выход и выполняет операцию инверсии (отрицания):

$$Y = \bar{X}$$

Функция $Y=1$ если сигнал $X=0$ и наоборот, $Y=0$ если $X=1$.

Элемент И-НЕ (NAND) (элемент Шеффера) можно представить комбинацией элементов И и НЕ (рис. 8).

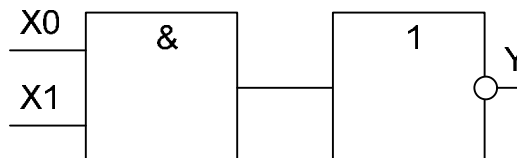


Рис. 8. Реализация элемента И-НЕ

На выходе элемент И-НЕ имеет логический 0 только в том случае, если на все входы подается логическая 1. Условное графическое обозначение элемента И-НЕ представлено на рис. 9.

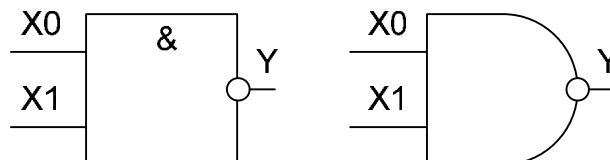


Рис. 9. Условное графическое обозначение элемента И-НЕ

Логический элемент И-НЕ имеет несколько входов и один выход и выполняет операцию логического умножения с отрицанием (конъюнкции с отрицанием):

$$Y = \overline{X_1 \cap X_0}$$

$$Y = \overline{X_n \cap \dots \cap X_1 \cap X_0} = \overline{X_n} \cup \dots \cup \overline{X_1} \cup \overline{X_0}$$

Функция $Y=0$, если все ее аргументы равны 1, и $Y=1$, если хотя бы один аргумент равен 0.

В табл. 7 приведена таблица истинности элемента И-НЕ.

Таблица 7

X1	X0	Y
0	0	1
0	1	1
1	0	1
1	1	0

Элемент И-НЕ является базисным, и на его основе можно получить все три основных логических функции (рис. 10).

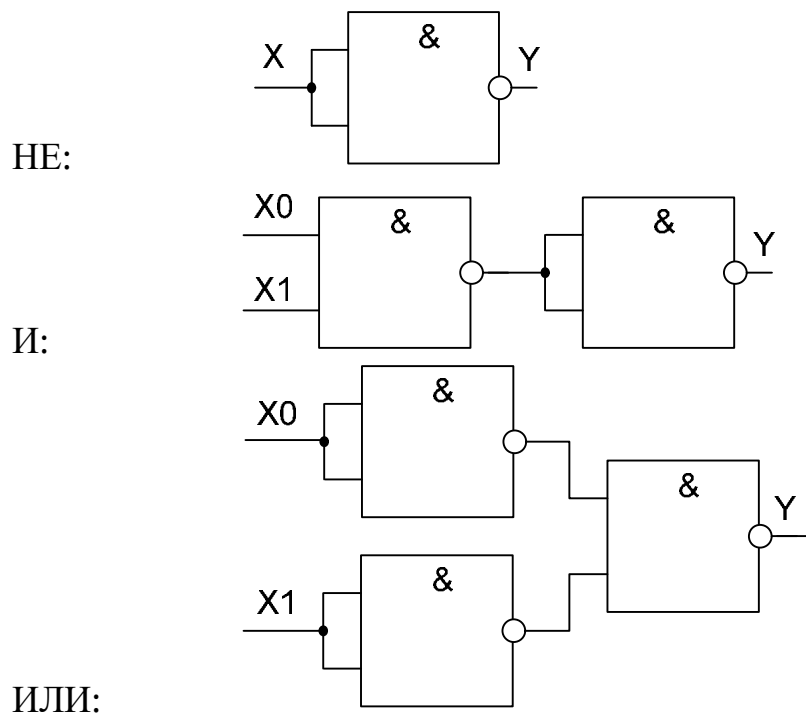


Рис. 10. Реализация элементов И, ИЛИ, НЕ на базе элемента И-НЕ

Наращивание размерности элемента И-НЕ производится в соответствии с уравнениями:

$$\overline{X_2 \cap X_1 \cap X_0} = \overline{X_2 \cap (\overline{\overline{X_1 \cap X_0}})}$$

$$\overline{X_3 \cap X_2 \cap X_1 \cap X_0} = \overline{(\overline{\overline{X_3 \cap X_2}}) \cap (\overline{\overline{X_1 \cap X_0}})}$$

по следующим схемам (рис. 11).

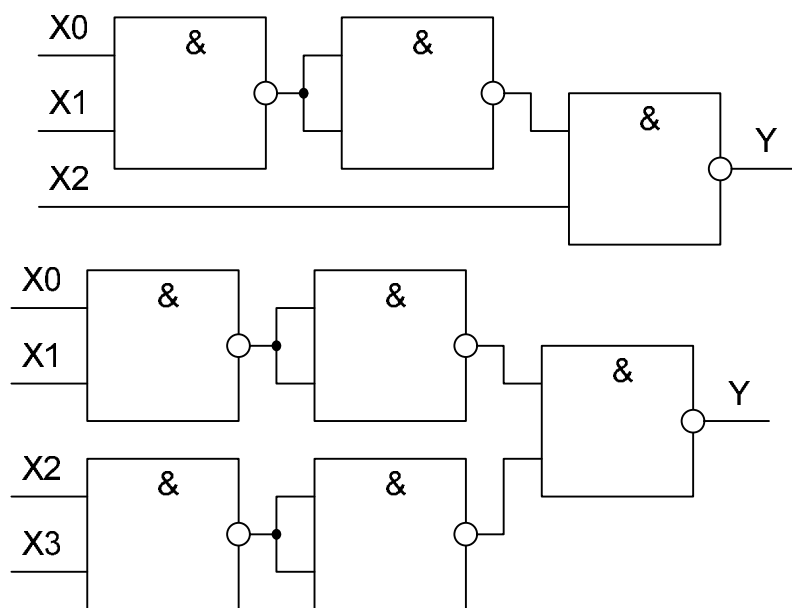


Рис. 11. Нарастание размерности элемента И-НЕ

Сокращение размерности элемента И-НЕ осуществляется аналогично сокращению размерности элемента И.

Элемент ИЛИ-НЕ (NOR) (элемент Пирса) можно представить комбинацией элементов ИЛИ и НЕ (рис. 12). Элемент ИЛИ-НЕ на выходе имеет логическую 1 только в том случае, если на все входы подается 0.

Условное графическое обозначение элемента ИЛИ-НЕ представлено на рис. 13.

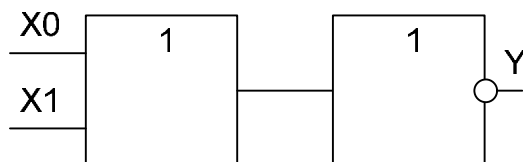


Рис. 12. Реализация элемента ИЛИ-НЕ

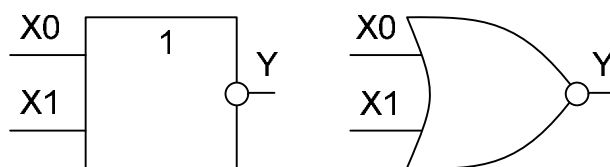


Рис. 13. Условное графическое обозначение элемента ИЛИ-НЕ

В табл. 8 приведена таблица истинности элемента ИЛИ-НЕ.

Таблица 8

Таблица истинности 2-входового элемента ИЛИ-НЕ

X1	X0	Y
0	0	1
0	1	0
1	0	0
1	1	0

Логический элемент ИЛИ-НЕ имеет несколько входов и один выход и выполняет операцию логического сложения с отрицанием (дизъюнкции с отрицанием):

$$Y = \overline{X_1 \cup X_0}$$

$$Y = \overline{X_n \cup \dots \cup X_1 \cup X_0} = \overline{X_n} \cap \dots \cap \overline{X_1} \cap \overline{X_0}$$

Функция $Y=1$, если все ее аргументы равны 0, и $Y=0$, если хотя бы один аргумент равен 1.

Элемент ИЛИ-НЕ является базисным, т.е. на его основе можно получить все 3 основные логические функции (рис. 14).

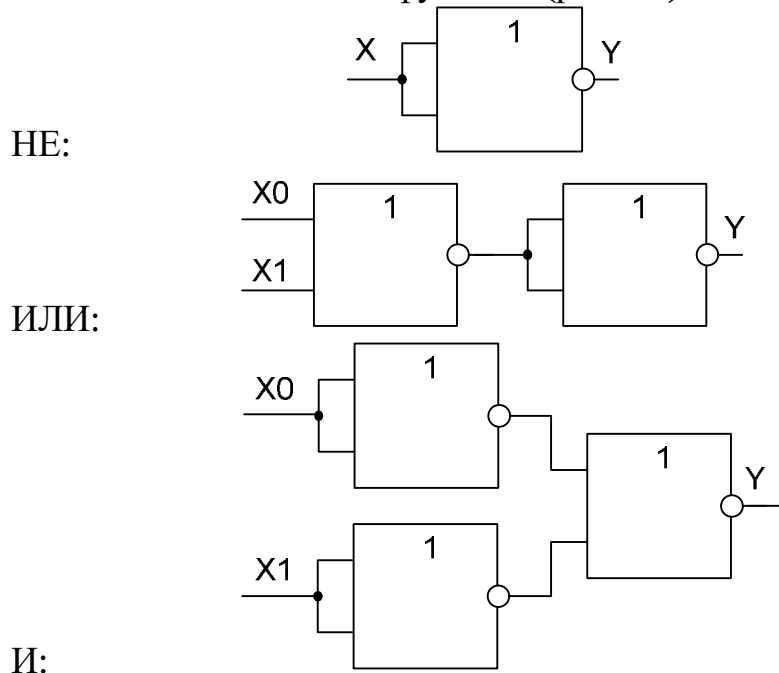


Рис. 14. Реализация элементов И, ИЛИ, НЕ на базе элемента ИЛИ-НЕ

Наращивание размерности элемента ИЛИ-НЕ производится в соответствии с уравнениями:

$$\overline{X_2 \cup X_1 \cup X_0} = \overline{X_2 \cup (\overline{\overline{X_1 \cup X_0}})}$$

$$\overline{X_3 \cup X_2 \cup X_1 \cup X_0} = \overline{(\overline{\overline{X_3 \cup X_2}}) \cup (\overline{\overline{X_1 \cup X_0}})}$$

по следующим схемам (рис. 15).

Сокращение размерности элемента ИЛИ-НЕ осуществляется аналогично сокращению размерности элемента ИЛИ.

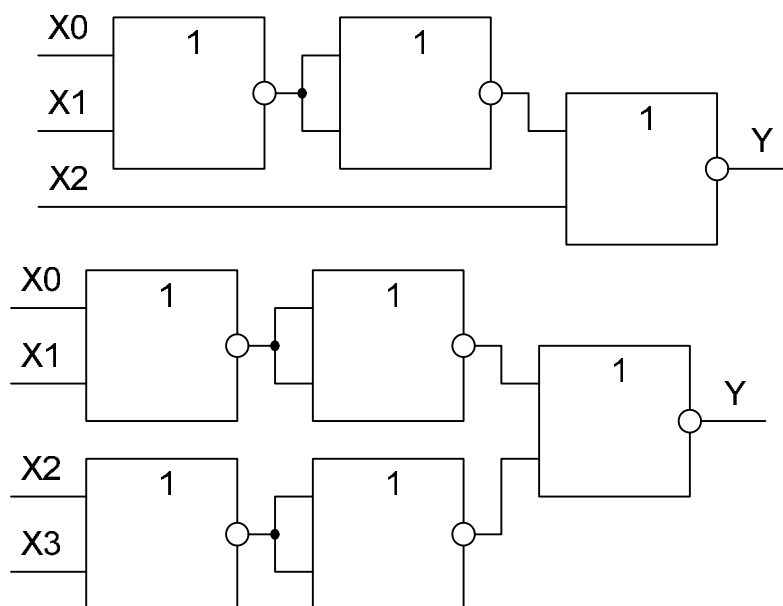


Рис. 15. Нарастивание размерности элемента ИЛИ-НЕ

Элемент «Исключающее ИЛИ» (XOR) («Сложение по модулю 2»): на выходе имеет логическую 1 только в том случае, если один и только один из входов имеет логическую 1.

В общем случае функция n логических переменных равна 1, если количество единиц в ее входном наборе нечетно. Условное графическое обозначение элемента «Исключающее ИЛИ» представлены на рис. 16.

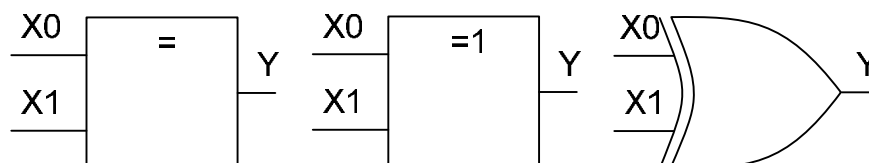


Рис. 16. Условное графическое обозначение элемента «Исключающее ИЛИ»

В табл. 9 представлена таблица истинности элемента «Исключающее ИЛИ»

Таблица 9

Таблица истинности 2-входового элемента «Исключающее ИЛИ»

X1	X0	Y
0	0	0
0	1	1
1	0	1
1	1	0

Элемент реализует функцию

$$Y = (X1 \cap \overline{X0}) \cup (\overline{X1} \cap X0)$$

С использованием рассмотренных выше базисных логических элементов данная функция может быть реализована как (рис. 17, 18).

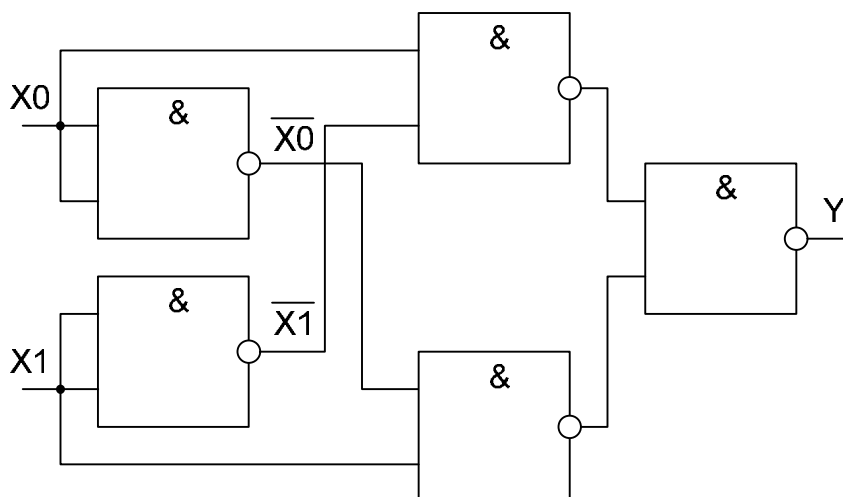


Рис. 17. Реализация элемента «Исключающее ИЛИ» в базисе И-НЕ

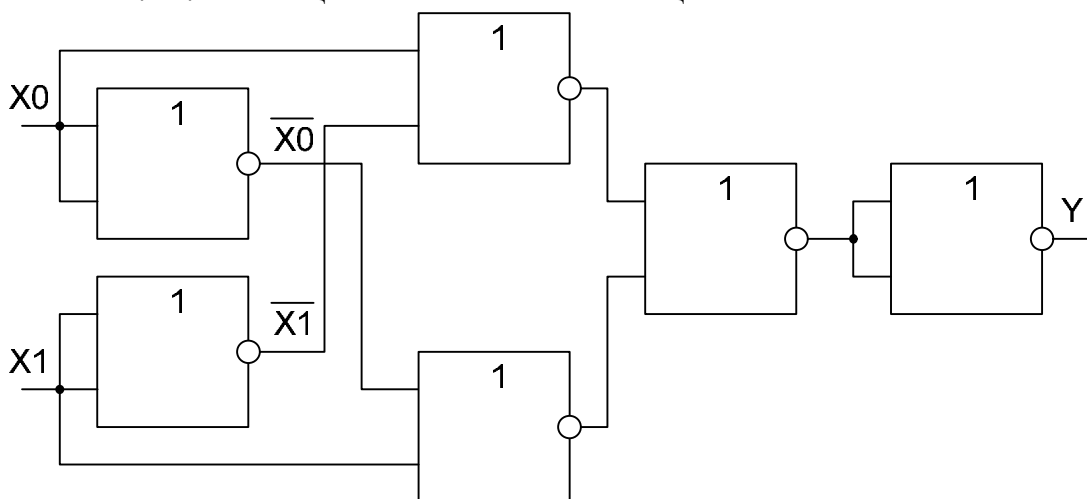


Рис. 18. Реализация элемента «Исключающее ИЛИ» в базисе ИЛИ-НЕ

Элемент «Исключающее ИЛИ-НЕ» (XNOR): на выходе имеет логическую 1, если значения входов совпадают.

В общем случае функция n логических переменных равна 1, если количество единиц в ее входном наборе четно. Условное графическое обозначение элемента представлены на рис. 19.

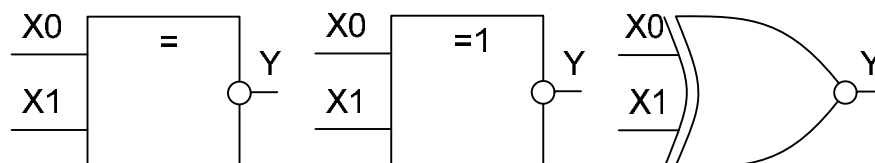


Рис. 19. Условное графическое обозначение элемента «Исключающее ИЛИ-НЕ»

Элемент реализует функцию

$$Y = (\overline{X1} \cap \overline{X0}) \cup (X1 \cap X0)$$

В таблице 10 представлена таблица истинности элемента «Исключающее ИЛИ-НЕ».

Таблица истинности 2-входового элемента «Исключающее ИЛИ-НЕ»

X1	X0	Y
0	0	1
0	1	0
1	0	0
1	1	1

На рассмотренных выше базисных логических элементах данная функция может быть реализована как (рис. 20, 21).

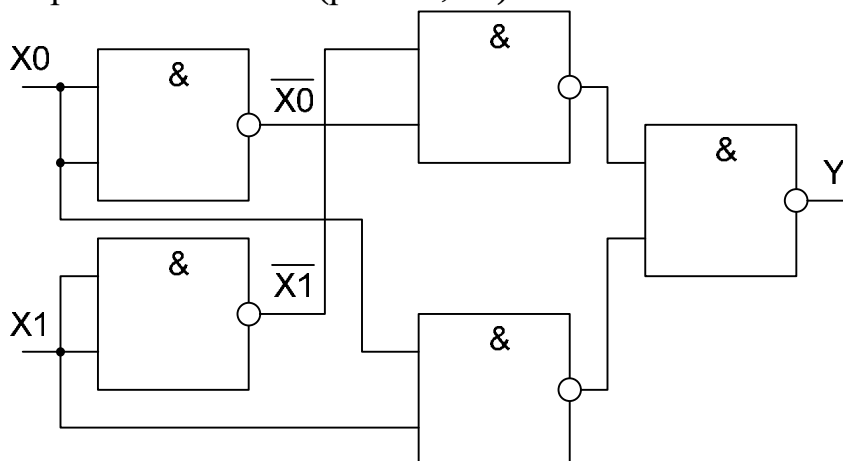


Рис. 20. Реализация элемента «Исключающее ИЛИ-НЕ» в базисе И-НЕ

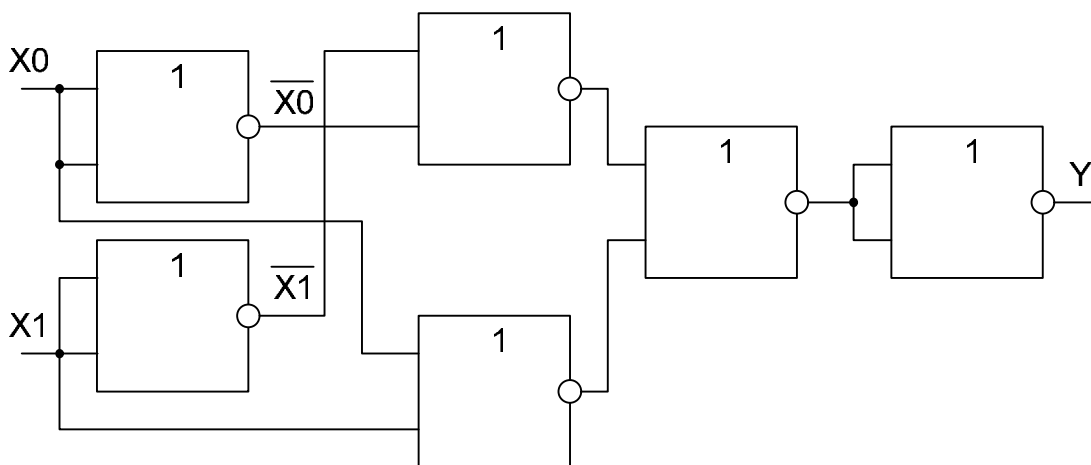


Рис. 21. Реализация элемента «Исключающее ИЛИ-НЕ» в базисе ИЛИ-НЕ

Любая функция алгебры логики может быть представлена в виде дизъюнкции конъюнкций наборов входных переменных, на которых эта функция принимает единичное значение.

Способы представления функций алгебры логики

Функция алгебры логики (ФАЛ) может принимать единичное, нулевое или неопределенное значение на каждом наборе входных переменных. Неопределенное значение предполагает, что данный набор входных переменных не является критичным, и функция может принимать на нем любое из двух логических значений.

Табличный способ: ФАЛ задается таблицей зависимости выходных значений от входных наборов. Такая таблица называется **таблицей истинности**.

Наборы входных переменных, как правило, перечисляются в лексикографическом порядке. Значение функции на каждом наборе входных переменных может быть нулевым, единичным или неопределенным.

Например, в таблице 11 задана ФАЛ для логического элемента «исключающее ИЛИ».

Таблица 11

Табличное представление ФАЛ элемента «Исключающее ИЛИ»

№ состояния	X1	X0	Y
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	0

Числовой способ: каждому набору входных переменных ставится в соответствие число в некоторой системе счисления (чаще всего двоичной или восьмеричной), называемое **номером состояния**. Состояния нумеруются по порядку, начиная с 0. ФАЛ задается в виде номеров состояний, на которых она принимает единичные значения, а также неопределенные значения (которые указываются в скобках). Например, для ФАЛ элемента «исключающее ИЛИ», представленной в таблице 11, числовая запись ФАЛ будет иметь вид:

$$Y = 1 \cup 2$$

Функция принимает единичное значение в состояниях 1 или 2. Во всех остальных случаях функция принимает нулевое значение.

Аналитический способ: ФАЛ задается в виде алгебраического выражения, получаемого при применении каких-либо логических операций к входным переменным.

Алгебраическое выражение показывает, какие логические операции должны выполняться над аргументами функции и какова должна быть их последовательность. Для элемента «исключающее ИЛИ», заданного в таблице 11, аналитическое задание ФАЛ будет выглядеть как:

$$Y = (\overline{X1} \cap X0) \cup (X1 \cap \overline{X0})$$

Как в арифметике, операция логического умножения может опускаться:

$$Y = (\overline{X1} X0) \cup (X1 \overline{X0})$$

Координатный способ: ФАЛ задается в виде карты состояний (карта Карно, диаграмма Вейча), которая представляет собой прямоугольную таблицу, разделенную на клетки. Общее число клеток карты равно числу наборов входных переменных. Выделяются клетки, значение функции в которых единично или неопределенно. Обычно такой способ представления исполь-

зуется при количестве переменных не более 4. Пример для элемента «исключающее ИЛИ» приведен на рис. 22.

	$\overline{X1}$	
	1	3
$\overline{X0}$	0	2

Рис. 22. Координатный способ задания ФАЛ

Рядом с картой подписано состояние входных переменных: например, для состояний 0 и 2 переменная $X0$ принимает нулевое значение, что обозначается чертой рядом с нижней строкой. Для состояний 0 и 1 переменная $X1$ принимает нулевое значение, что обозначается чертой рядом с левым столбцом. В данном случае черта обозначает инверсию.

Минимизация функций алгебры логики

Минимизация – это процесс нахождения такого эквивалентного выражения ФАЛ, которое содержит минимальное число вхождений переменных.

При минимизации могут быть различные варианты получения выражений с минимальным числом вхождений переменных.

Если значение ФАЛ однозначно определено на всех возможных наборах значений входных переменных, то она называется **полностью определенной**.

Если на некоторых наборах входных переменных значение ФАЛ является безразличным и однозначно не определено, то функция называется **не полностью (частично) определенной**.

Кодовой комбинацией называется набор логических состояний входных переменных.

Кодовое расстояние между двумя кодовыми комбинациями – число разрядов (или число переменных), в которых эти комбинации отличаются друг от друга.

Минимальное кодовое расстояние – минимальное кодовое расстояние для любой пары комбинаций, входящих в данный код. Для двоичного кода минимальное кодовое расстояние равно 1 (две кодовых комбинации отличаются как минимум логическим состоянием одной переменной).

Две кодовых комбинации с кодовым расстоянием, равным 1, называются **соседними**. У каждой кодовой комбинации количество соседних комбинаций определяется количеством переменных во входном наборе.

Правила минимизации функций алгебры логики:

1. Все единичные состояния должны войти в результирующее выражение минимизированной ФАЛ.

2. Ни одно нулевое состояние не должно попасть в выражение минимизированной ФАЛ.

3. Любое неопределенное состояние может как входить, так и не входить в результирующее выражение минимизированной ФАЛ.

4. Число объединяемых состояний должно быть кратно степени 2.

5. Единичные и неопределенные состояния могут использоваться при минимизации несколько раз.

6. Для получения наиболее простого результирующего выражения минимизированной ФАЛ следует объединять максимально возможное количество рядом стоящих состояний.

Минимизация при помощи диаграммы Вейча или карты Карно: для любой клетки карты существует соседняя по строке столбцу такая, что координаты этой клетки отличаются от координат рассматриваемой клетки лишь одной переменной. Это свойство позволяет выполнить на карте минимизацию функции, используя закон склеивания соседних состояний.

На рис. 23 приведена диаграмма Вейча, на которой задана ФАЛ вида

$$Y = 0 \cup (1) \cup (6) \cup 7 \cup 10 \cup 11 \cup (12)$$

Номера состояний на диаграмме Вейча указываются в восьмеричной системе счисления. Единичные состояние обведены кружком, неопределенные отмечены серым цветом.

		$\overline{X1}$				
		7	5	1	3	
		6	4	0	2	
		16	14	10	12	
		17	15	11	13	
		$X2$				
	$\overline{X0}$					$X3$

Рис. 23. Диаграмма Вейча с отмеченными номерами состояний
Каждому состоянию соответствует набор входных переменных:

$$\begin{array}{ll}
 0: \overline{X3} \cap \overline{X2} \cap \overline{X1} \cap \overline{X0} & 10: X3 \cap \overline{X2} \cap \overline{X1} \cap \overline{X0} \\
 (1): \overline{X3} \cap \overline{X2} \cap \overline{X1} \cap X0 & 11: X3 \cap \overline{X2} \cap \overline{X1} \cap X0 \\
 (6): \overline{X3} \cap X2 \cap X1 \cap \overline{X0} & (12): X3 \cap \overline{X2} \cap X1 \cap \overline{X0} \\
 7: \overline{X3} \cap X2 \cap X1 \cap X0 &
 \end{array}$$

Таким образом, можем записать ФАЛ в аналитическом виде:

$$\begin{aligned}
 Y = & \overline{X3} \overline{X2} \overline{X1} \overline{X0} \cup (\overline{X3} \overline{X2} \overline{X1} X0) \cup (\overline{X3} X2 X1 \overline{X0}) \cup \overline{X3} X2 X1 X0 \cup \\
 & \cup X3 \overline{X2} \overline{X1} \overline{X0} \cup X3 \overline{X2} \overline{X1} X0 \cup X3 \overline{X2} X1 \overline{X0}
 \end{aligned}$$

Минимизация предполагает объединение пар, четверок, восьмерок и т.д. соседних состояний, причем выбирать состояния, совмещаемые с соседними, следует исходя из заданных единичных состояний ФАЛ.

На диаграмме Вейча у каждого состояния можно выделить 4 соседних: вправо, влево, вверх и вниз. Так, у состояния 0 соседними являются 1, 2, 4, 10. У состояния 7 соседними являются 3, 5, 6, 17. Любую пару соседних состояний можно совместить, используя закон поглощения, если функция на каждом из состояний этой пары принимает единичное или неопределенное значение.

$$0 \cup (1) = \overline{X_3} \overline{X_2} \overline{X_1} \overline{X_0} \cup (\overline{X_3} \overline{X_2} \overline{X_1} X_0) = \overline{X_3} \overline{X_2} \overline{X_1}$$

Состояние 0, в котором ФАЛ принимает единичное значение, объединили с состоянием 1, в котором ФАЛ принимает неопределенное значение, и применили закон поглощения. Аналогичным образом запишем объединения для всех остальных состояний.

$$10 \cup 11 = X_3 \overline{X_2} \overline{X_1} \overline{X_0} \cup X_3 \overline{X_2} \overline{X_1} X_0 = X_3 \overline{X_2} \overline{X_1}$$

$$(6) \cup 7 = (\overline{X_3} X_2 X_1 \overline{X_0}) \cup \overline{X_3} X_2 X_1 X_0 = \overline{X_3} X_2 X_1$$

Применив закон поглощения, можем также записать:

$$0 \cup (1) \cup 10 \cup 11 = \overline{X_3} \overline{X_2} \overline{X_1} \cup X_3 \overline{X_2} \overline{X_1} = \overline{X_2} \overline{X_1}$$

Таким образом, запись минимизированной ФАЛ будет выглядеть как

$$Y = \overline{X_2} \overline{X_1} \cup \overline{X_3} X_2 X_1$$

В минимизацию вошли все единичные состояния и неопределенные состояния (1) и (6), которые были доопределены единичным значением. Неопределенное состояние (12) не вошло в запись минимизированной ФАЛ и, следовательно, было доопределено нулевым значением.

Моделирование цифровых схем

Для моделирования схем, приведенных в лабораторных работах, рекомендуется использовать свободно распространяемую программу LTSpice. Также необходимо установить дополнительно библиотеки с моделями цифровых узлов [6].

Для установки библиотек запускаем LTSpice, создаем новую схему и нажимаем кнопку вставки компонента (рис. 24).

В верхней строке отображается путь, где хранятся библиотеки(рис. 25).



Рис. 24. Добавление компонентов в LTSpice

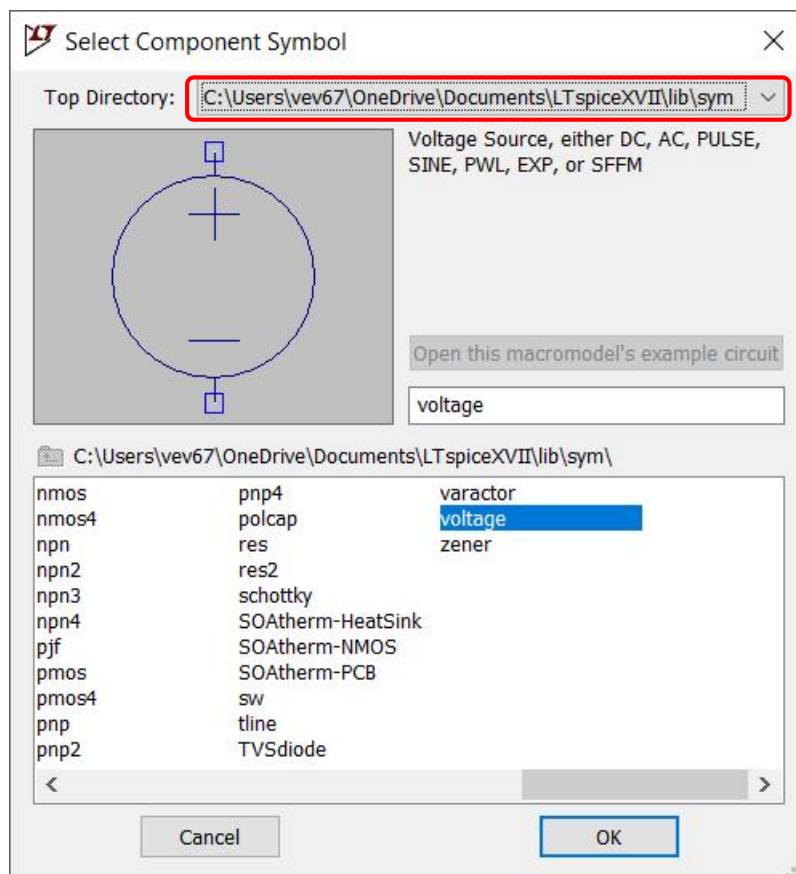


Рис. 25. Путь до библиотечных файлов

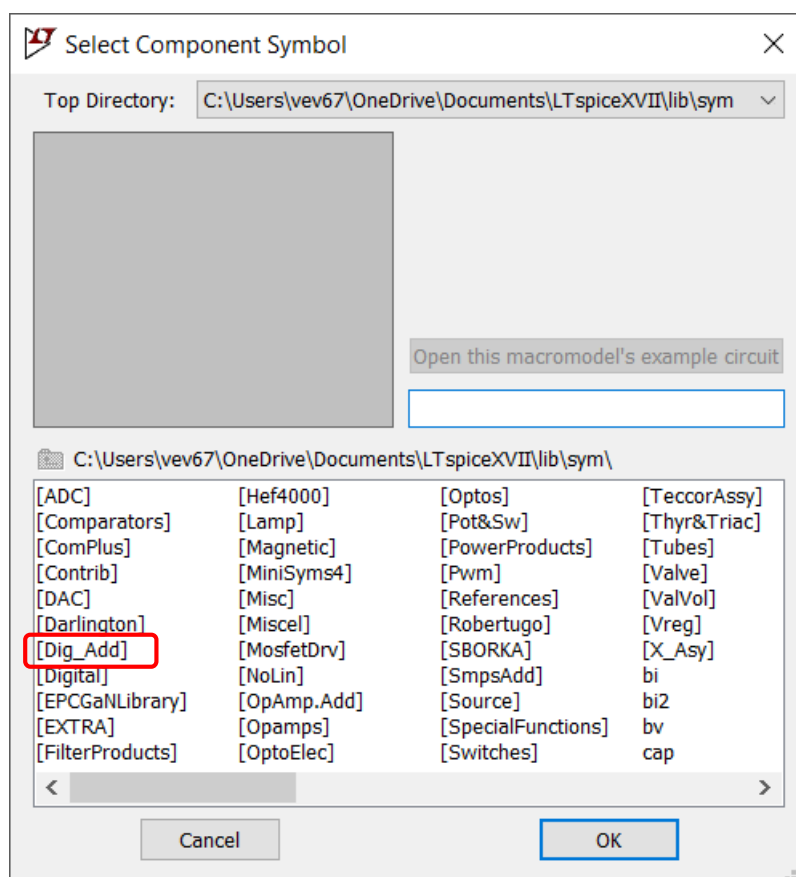


Рис. 26. Папка с дополнительными компонентами

Из скачанного архива библиотек нас интересует содержимое папки lib. Копируем папку lib по пути, указанному в LTSpice.

Теперь при добавлении компонентов доступны новые компоненты из установленных библиотек.

Библиотеки для моделирования хранятся в папке ../lib/sub/.

Чтобы моделировать схему с цифровыми компонентами, на схеме необходимо дополнительно указать директиву подключения моделей компонентов (рис. 27, 28)

```
.inc 74hc.lib
```

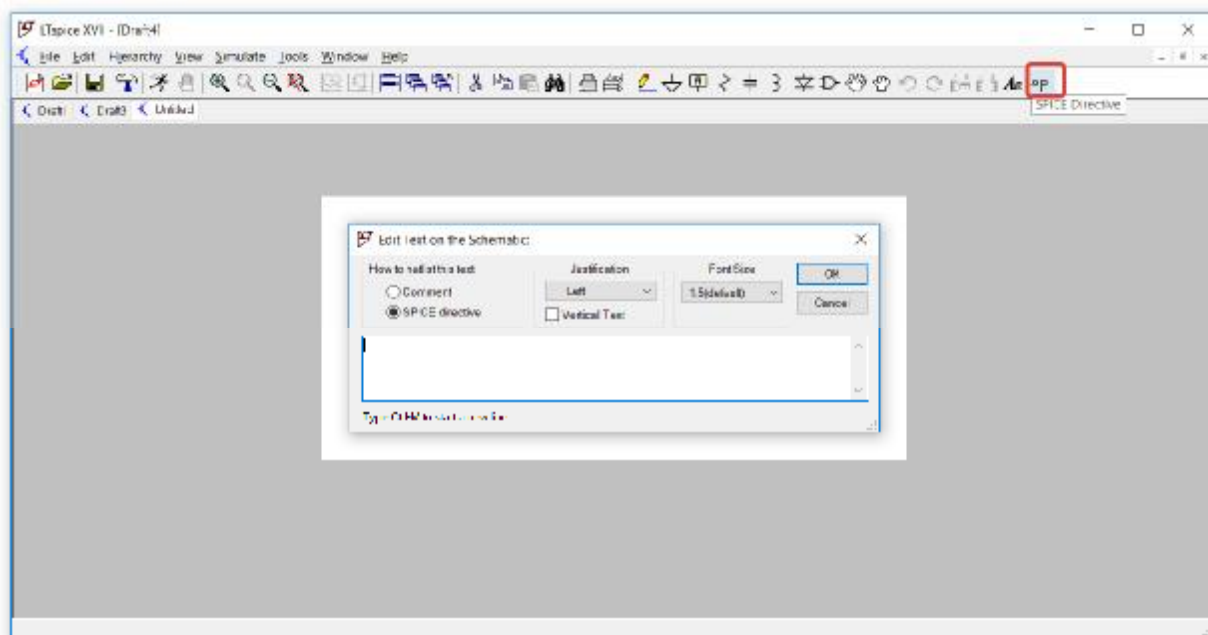


Рис. 27. Директива для подключения моделей компонентов

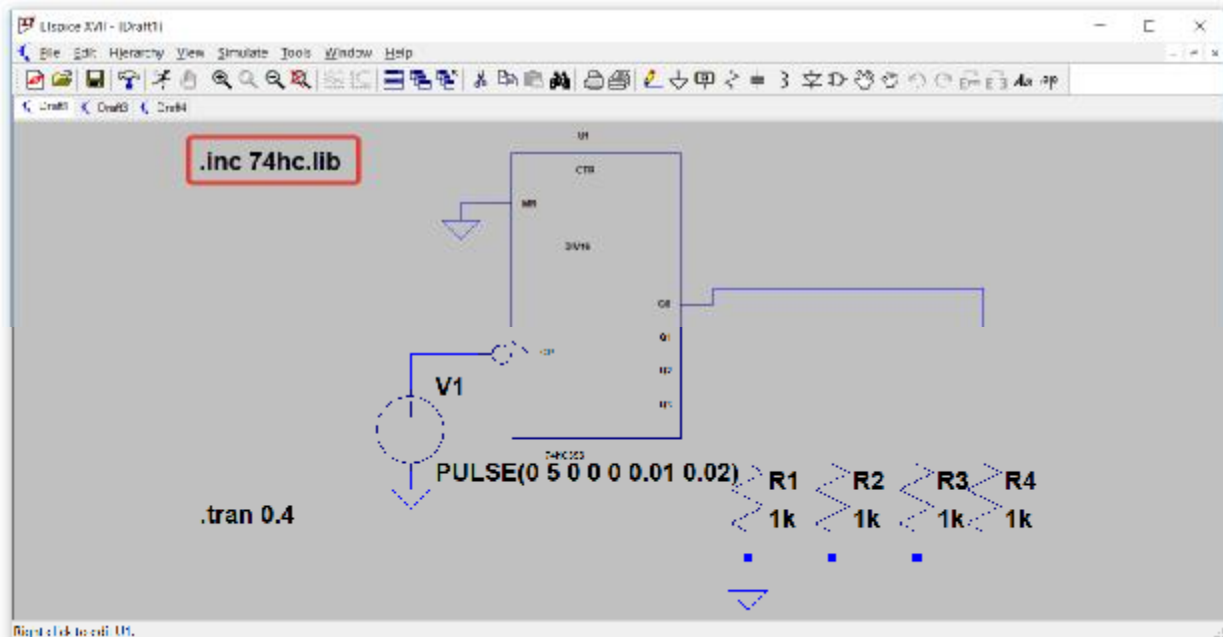


Рис. 28. Размещение директивы на схеме

ЛАБОРАТОРНАЯ РАБОТА 1. СХЕМА РАВНОЗНАЧНОСТИ КОДОВ

Цель работы: изучение принципов построения схем сравнения кодов и примера их практического использования как неполных перенастраиваемых дешифраторов.

Пояснения к работе. Схема равнозначности кодов является частным случаем схем сравнения кодов и выполняет формирование выходного логического сигнала при равенстве входного произвольно задаваемого на разрядные входы кода установленному коду, фиксированному на других разрядных входах. На рис. 29 изображена схема равнозначности двух 5-разрядных кодов.

Для задания всех возможных комбинаций входного кода (наборов значений входных переменных) используются два двоичных таймера-счетчика 74НС393. При этом сигналом синхронизации второго счетчика является старший разряд первого счетчика. Входные коды поразрядно сравниваются со значениями разрядов фиксированного кода с помощью логических элементов «ИСКЛЮЧАЮЩЕЕ ИЛИ» или «СЛОЖЕНИЕ ПО МОДУЛЮ 2» (XOR).

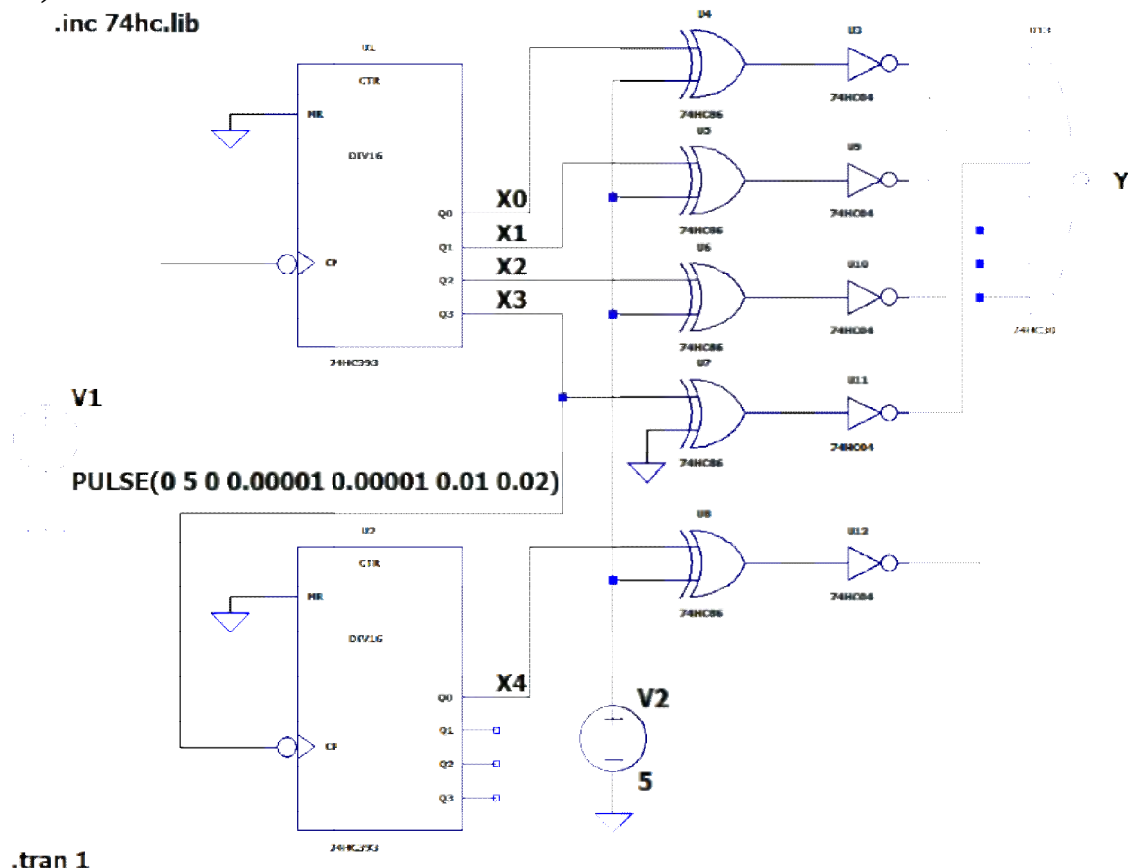


Рис. 29. Схема равнозначности кодов

Логический элемент «ИСКЛЮЧАЮЩЕЕ ИЛИ» формирует единичный сигнал на выходе тогда, когда значения логических сигналов на его входах

имеют различное значение. При одинаковых значениях входных сигналов элемент «ИСКЛЮЧАЮЩЕЕ ИЛИ» формирует нулевой сигнал.

Наличие на выходах всех ЛЭ «ИСКЛЮЧАЮЩЕЕ ИЛИ» нулевых сигналов (все разряды кода совпали) обнаруживается при подключении выходов ЛЭ «ИСКЛЮЧАЮЩЕЕ ИЛИ» к схеме ИЛИ. Только в случае если все входные сигналы равны нулю схема ИЛИ даст нулевое значение.

Таким образом, выходной сигнал принимает нулевое значение только в том случае, когда входной код совпадает с фиксированным кодом, набранным на входах элементов «ИСКЛЮЧАЮЩЕЕ ИЛИ».

Для перебора всех возможных кодовых комбинаций входных переменных используется меандровый тактовый сигнал частотой 50 Гц, подаваемый на вход счетчика. Настройки используемого тактового сигнала приведены на рис. 30.

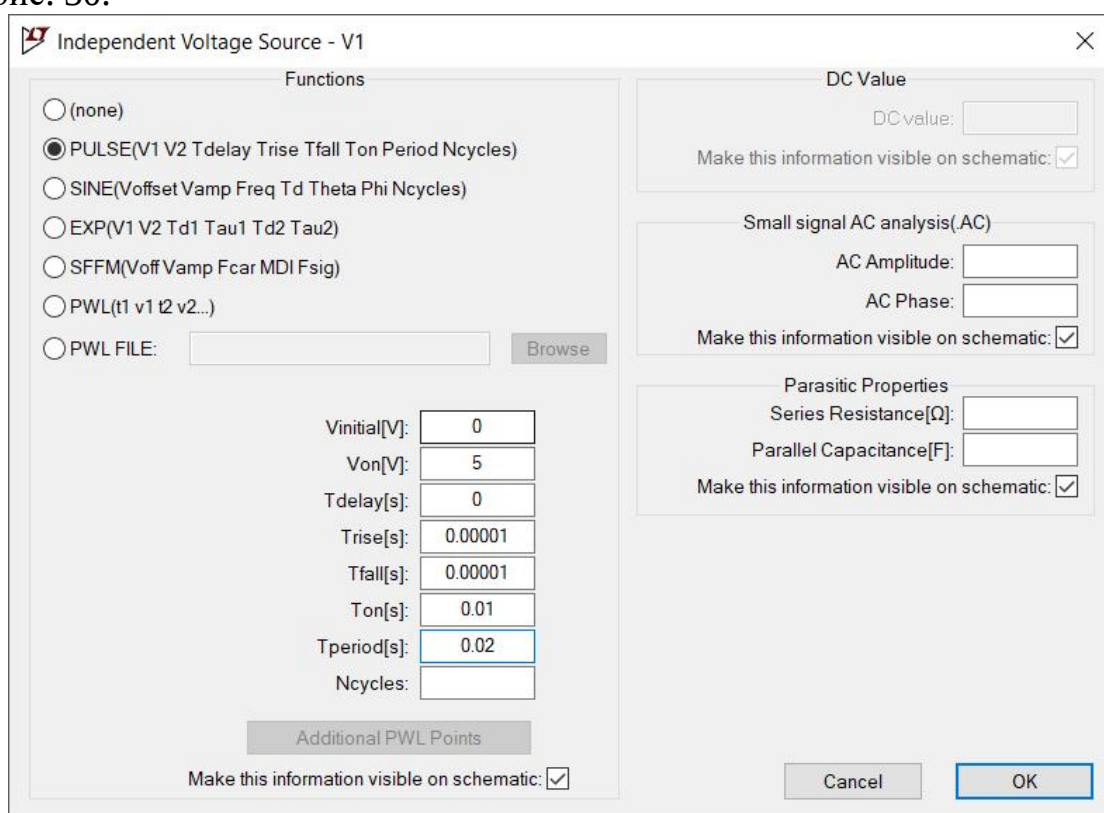


Рисунок 30. Настройки тактового сигнала

На примере схемы (рис. 29) набран код варианта 10111. Результат моделирования в программе LTSpice приведен на рис. 31.

Как видим, выход функции Y принимает нулевое значение только на наборе входных переменных, соответствующем варианту: 10111.

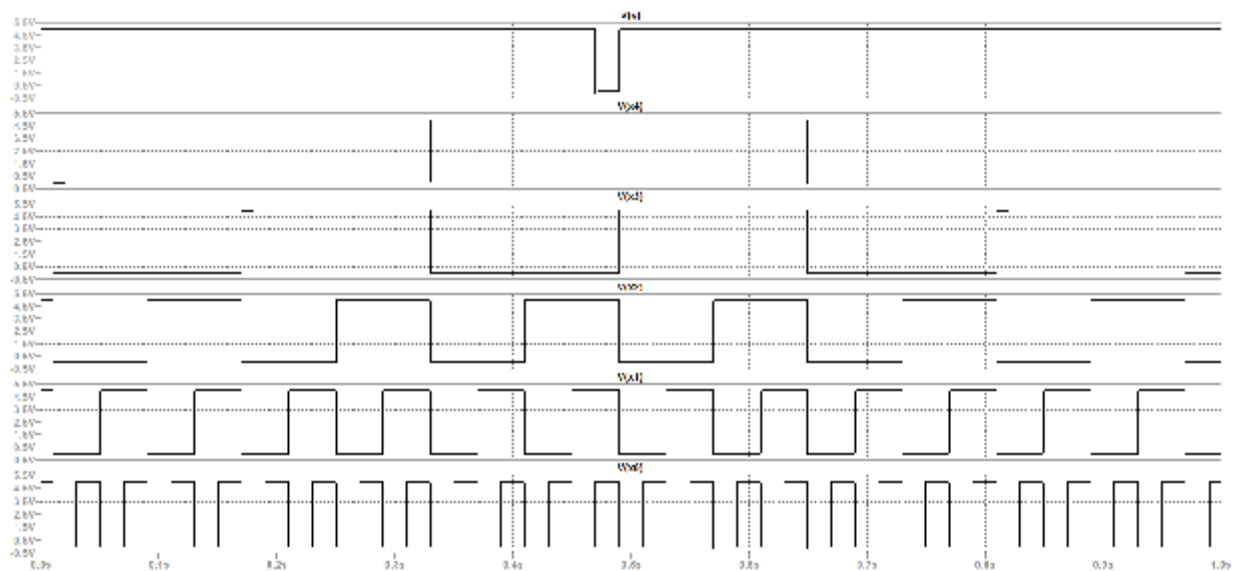


Рис. 31. Результат моделирования схемы равнозначности кодов

Содержание работы

1. Заполнить таблицу истинности для логического элемента «ИСКЛЮЧАЮЩЕЕ ИЛИ».
2. Перевести номер варианта в двоичный код.
3. Набрать схему моделирования.
4. Подключить входы задания фиксированного кода в соответствии с вариантом задания.
5. Запустить процесс моделирования схемы.
6. Привести результат моделирования схемы аналогично рис. 31.
7. Заполнить таблицу истинности для заданного варианта схемы равнозначности кодов по экспериментальным данным.

Таблица 12

Результат моделирования схемы равнозначности кодов

№	X4	X3	X2	X1	X0	Y
0	0	0	0	0	0	1
1	0	0	0	0	1	1
2	0	0	0	1	0	1
3	0	0	0	1	1	1
4	0	0	1	0	0	1
5	0	0	1	0	1	1
6	0	0	1	1	0	1
7	0	0	1	1	1	1
10	0	1	0	0	0	1
11	0	1	0	0	1	1
12	0	1	0	1	0	1
13	0	1	0	1	1	1
14	0	1	1	0	0	1
15	0	1	1	0	1	1
16	0	1	1	1	0	1
17	0	1	1	1	1	1

№	X4	X3	X2	X1	X0	Y
20	1	0	0	0	0	1
21	1	0	0	0	1	1
22	1	0	0	1	0	1
23	1	0	0	1	1	1
24	1	0	1	0	0	1
25	1	0	1	0	1	1
26	1	0	1	1	0	1
27	1	0	1	1	1	0
30	1	1	0	0	0	1
31	1	1	0	0	1	1
32	1	1	0	1	0	1
33	1	1	0	1	1	1
34	1	1	1	0	0	1
35	1	1	1	0	1	1
36	1	1	1	1	0	1
37	1	1	1	1	1	1

Содержание отчёта

Отчёт должен содержать результаты исследования схемы по следующим пунктам.

1. Таблица истинности ЛЭ «ИСКЛЮЧАЮЩЕЕ ИЛИ».
2. Схему равнозначности кода для заданного варианта.
3. Результат моделирования схемы.
4. Таблица истинности, полученная в результате исследования результата моделирования схемы равнозначности кодов.
5. Выводы по результатам выполнения работы.

ЛАБОРАТОРНАЯ РАБОТА 2. МИНИМИЗАЦИЯ БУЛЕВЫХ ФУНКЦИЙ

Цель работы: изучение способов описания и минимизации функций алгебры логики (ФАЛ) с использованием диаграмм Вейча и реализация ФАЛ в базисе И-НЕ (ЛЭ Шеффера). Варианты ФАЛ заданы в табл. 13.

Пояснения к работе: Коды наборов переменных даны в таблицах в восьмеричной системе счисления. На тех кодах наборов, где ФАЛ не определена, проставлена метка (х).

Выполнение лабораторной работы рассмотрим для примера ФАЛ, заданной в кодированной совершенной дизъюнктивной нормальной форме (СДНФ):

$$Y = 0 \cup 3 \cup 6 \cup 13 \cup (2) \cup (12) \cup (14).$$

В ФАЛ Y коды наборов переменных даны в восьмеричной системе счисления, а на наборах, указанных в круглых скобках, ФАЛ не определена.

Поскольку ФАЛ является функцией четырёх переменных, то её аналитическое описание в СДНФ имеет вид

$$Y = \overline{X_3} \overline{X_2} \overline{X_1} \overline{X_0} \cup \overline{X_3} \overline{X_2} X_1 X_0 \cup \overline{X_3} X_2 X_1 \overline{X_0} \cup X_3 \overline{X_2} X_1 X_0 \cup (\overline{X_3} \overline{X_2} X_1 \overline{X_0}) \cup (X_3 \overline{X_2} X_1 X_0) \cup (X_3 X_2 \overline{X_1} \overline{X_0})$$

Значения ФАЛ для всех наборов переменных представлены в табл. 14.

Таблица 13

Варианты для выполнения лабораторной работы

Вариант	Состояние (8-ричная система)																
	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	
1	0	0	x	0	1	0	1	x	1	0	1	0	0	0	x	0	
2	0	1	0	0	1	0	1	x	0	0	0	x	0	1	x	0	
3	x	0	1	0	0	0	1	0	0	x	x	0	1	0	0	1	
4	1	1	0	0	0	1	1	x	0	0	0	0	x	0	x	0	
5	0	0	1	x	1	0	x	1	0	0	x	0	0	0	1	0	
6	x	x	x	0	0	1	0	1	0	1	0	0	0	0	0	1	
7	0	0	x	0	0	1	1	0	1	0	0	1	x	x	0	0	
8	x	0	0	x	1	0	0	0	0	1	0	0	x	0	1	1	
9	1	0	0	x	0	0	0	1	0	0	0	1	x	1	x	0	

Вариант	Состояние (8-ричная система)																
	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17	
10	0	1	0	0	x	0	1	x	1	0	x	1	0	0	0	0	
11	1	0	0	1	0	0	1	0	1	0	0	x	x	x	0	0	
12	0	1	0	x	0	1	0	x	1	0	0	0	x	1	0	0	
13	0	x	0	0	0	1	0	1	0	1	1	0	0	0	x	x	
14	1	0	0	0	0	1	0	x	1	1	0	0	x	0	x	0	
15	0	1	x	0	1	1	x	0	x	1	0	0	0	0	0	0	
16	0	0	0	0	1	x	x	0	0	1	1	0	0	1	x	0	
17	0	1	1	1	0	0	0	x	0	x	0	0	0	1	x	0	
18	0	x	0	0	0	0	1	1	x	0	0	1	0	x	1	0	
19	1	0	0	x	x	0	0	x	1	0	0	1	0	1	0	0	
20	1	1	1	0	x	0	0	0	0	1	0	0	0	x	x	0	
21	0	0	x	0	0	x	0	1	1	0	x	0	0	1	1	0	
22	x	1	1	0	0	0	1	x	0	x	0	0	1	0	0	0	
23	0	0	x	1	0	1	0	x	1	0	0	x	0	1	0	0	
24	1	0	x	0	x	0	0	x	1	1	0	0	0	1	0	0	
25	x	x	0	0	0	1	0	0	1	0	0	1	1	0	x	0	
26	1	0	1	0	x	1	0	0	0	x	0	0	0	1	0	x	
27	0	0	0	1	0	0	1	1	0	0	1	x	0	x	0	x	
28	x	1	x	0	0	1	0	0	0	0	x	1	0	0	0	1	
29	0	0	1	1	x	x	0	0	1	1	0	0	x	0	0	0	
30	0	1	0	x	1	0	0	1	0	0	1	x	x	0	0	0	

Таблица 14

Табличный способ представления ФАЛ

Состояние	X3	X2	X1	X0	Y
0	0	0	0	0	1
1	0	0	0	1	0
2	0	0	1	0	X
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	1
7	0	1	1	1	0
10	1	0	0	0	0
11	1	0	0	1	0
12	1	0	1	0	X
13	1	0	1	1	1
14	1	1	0	0	X
15	1	1	0	1	0
16	1	1	1	0	0
17	1	1	1	1	0

Минимизация по диаграмме Вейча представлена на рис. 32.

	$\overline{X1}$				
	7	5	1	3	
$\overline{X0}$	6	4	0	2	$\overline{X3}$
	16	14	10	12	
	17	15	11	13	
	$\overline{X2}$				

Рис. 32. Минимизация ФАЛ по диаграмме Вейча

Диаграмма Вейча представляет собой набор восьмеричных состояний, где каждые два соседних состояния отличаются только одной логической переменной. При этом крайние состояния также являются соседними. По сторонам диаграммы полосой обозначено нулевое значение логической переменной. То есть, например, переменная $X3$ принимает нулевые значения в верхней части диаграммы.

В данном случае единичные состояния отмечены на диаграмме кругами, неопределенные состояния – серым цветом.

Каждые два рядом стоящих состояния можно совместить, при этом результатом являются 3 логических переменных, имеющих одинаковое значение для этих состояний. Используется свойство склеивания логических выражений.

Так, например, два соседних состояния 0 и 2 отличаются только переменной $X1$, в результате чего логическое ИЛИ этих двух состояний будет выглядеть как

$$0 \cup 2 = \overline{X3} \overline{X2} \overline{X1} \overline{X0} \cup \overline{X3} \overline{X2} X1 \overline{X0} = \overline{X3} \overline{X2} \overline{X0}$$

Целью минимизации является использование всех определенных единичных состояний в результирующем выражении, по крайней мере, 1 раз. Неопределенные состояния можно доопределить 0, а можно 1. Если неопределенное состояние включается в минимизацию, то оно доопределяется как 1. Если не включается – то 0.

Совместим остальные пары состояний

$$3 \cup 13 = \overline{X3} \overline{X2} X1 X0 \cup X3 \overline{X2} X1 X0 = \overline{X2} X1 X0$$

$$6 \cup 2 = \overline{X3} X2 X1 \overline{X0} \cup \overline{X3} \overline{X2} X1 \overline{X0} = \overline{X3} X1 \overline{X0}$$

Проверяем, что использованы все единичные состояния: 0, 3, 6, 13.

В результате минимизированная ФАЛ будет иметь вид

$$Y = \overline{X3} \overline{X2} \overline{X0} \cup \overline{X2} X1 X0 \cup \overline{X3} X1 \overline{X0}$$

Совмещать можно также по 4, 8, 16 и т. д. рядом стоящих состояния. В результате при склеивании 4 состояний сократятся две переменных.

Минимизированная ФАЛ при использовании базиса И-НЕ (ЛЭ Шеффера) имеет описание

$$Y = \overline{\overline{\overline{X3} \overline{X2} \overline{X0}}} \cap \overline{\overline{\overline{X2} X1 X0}} \cap \overline{\overline{\overline{X3} X1 \overline{X0}}}$$

Описание схемы исследования

На рис. 33 представлена схема исследования для неминимизированной ФАЛ. Генератором сигналов служит двоичный счетчик 74НС393, перебирающий все возможные состояния четырех логических переменных (рис.

34). В результате моделирования ФАЛ получили, что выход F имеет единичное значение в состояниях 0, 3, 6, 13.

На рис. 30 представлена настройка импульсного генератора, сигнал с которого поступает на счетный вход элемента 74НС393.

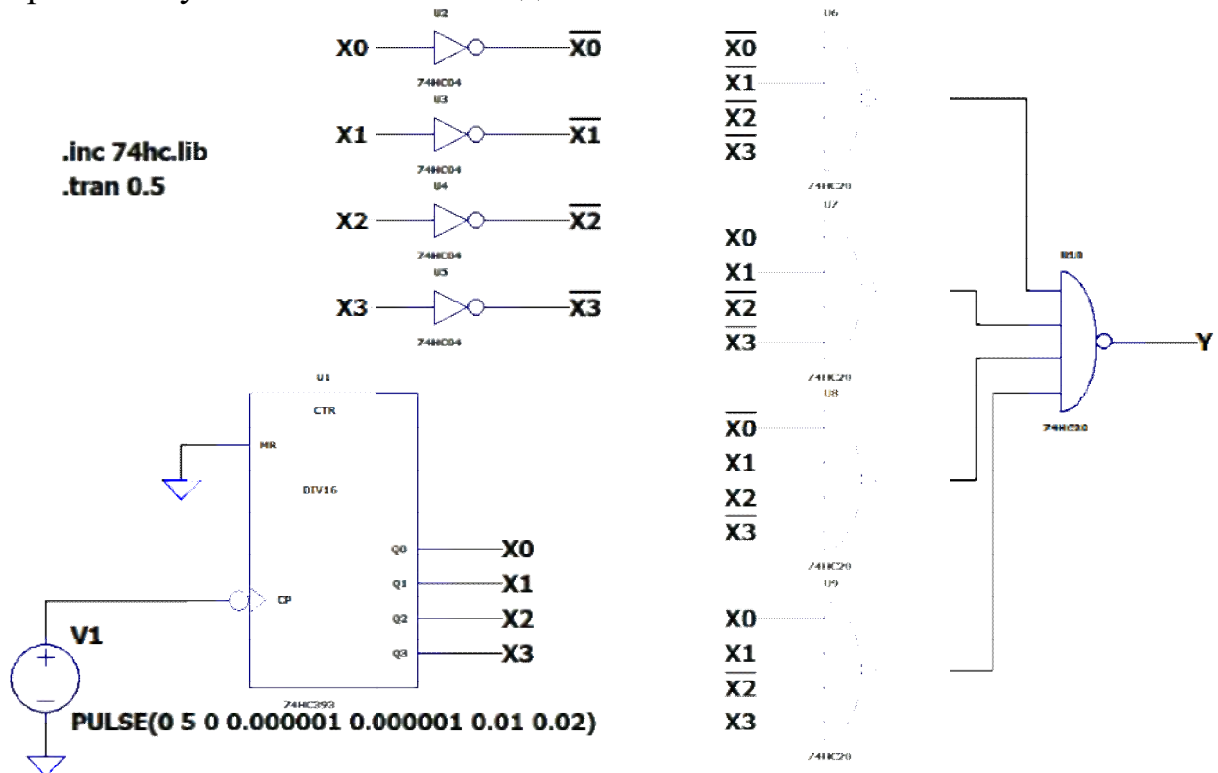


Рис. 33. Схема исследования неминимизированной ФАЛ

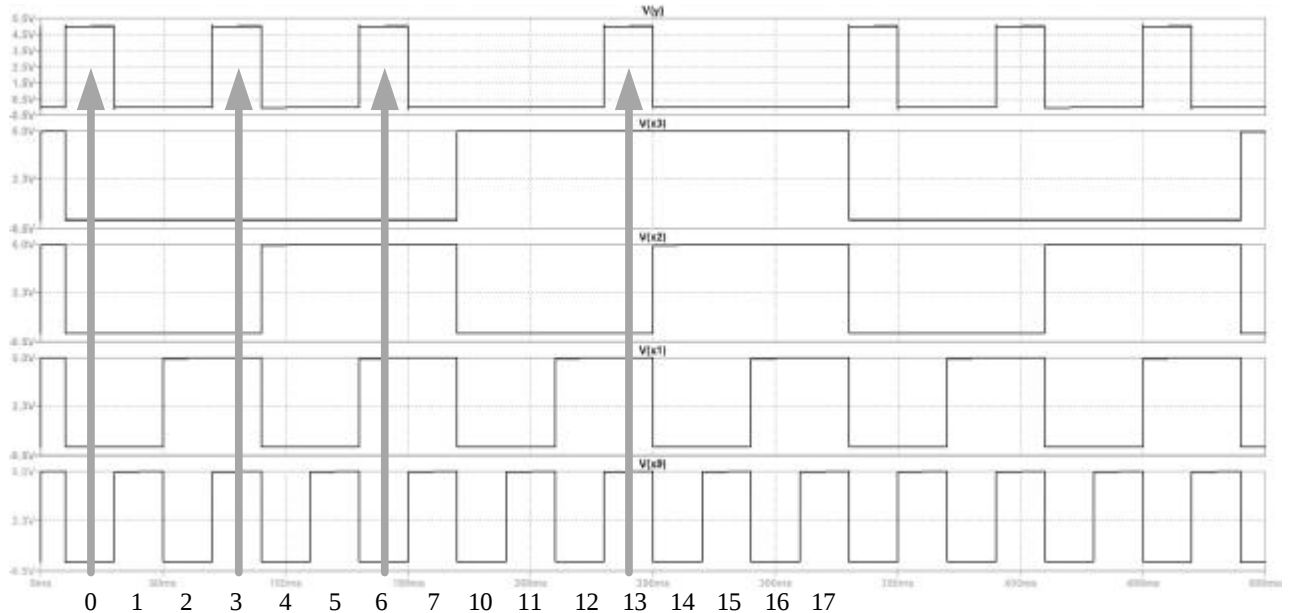


Рис. 34. Результат моделирования схемы неминимизированной ФАЛ

На рис. 35 представлена схема исследования минимизированной ФАЛ. На рис. 36 представлен результат моделирования схемы минимизированной ФАЛ.

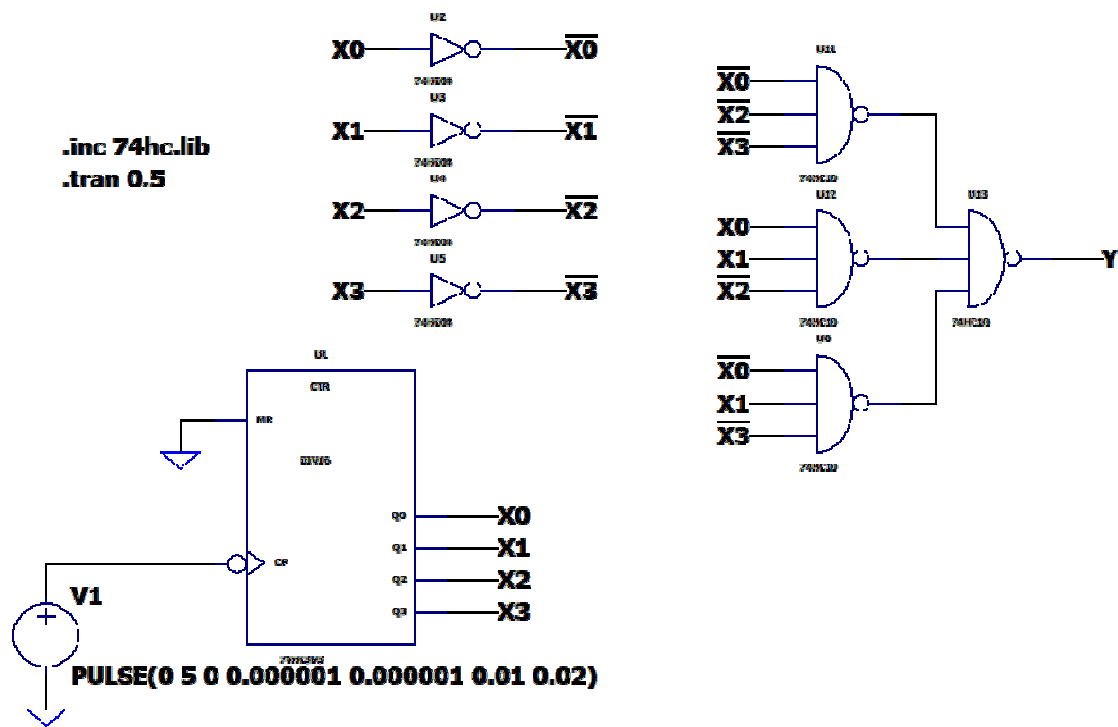


Рис. 35. Схема исследования минимизированной ФАЛ

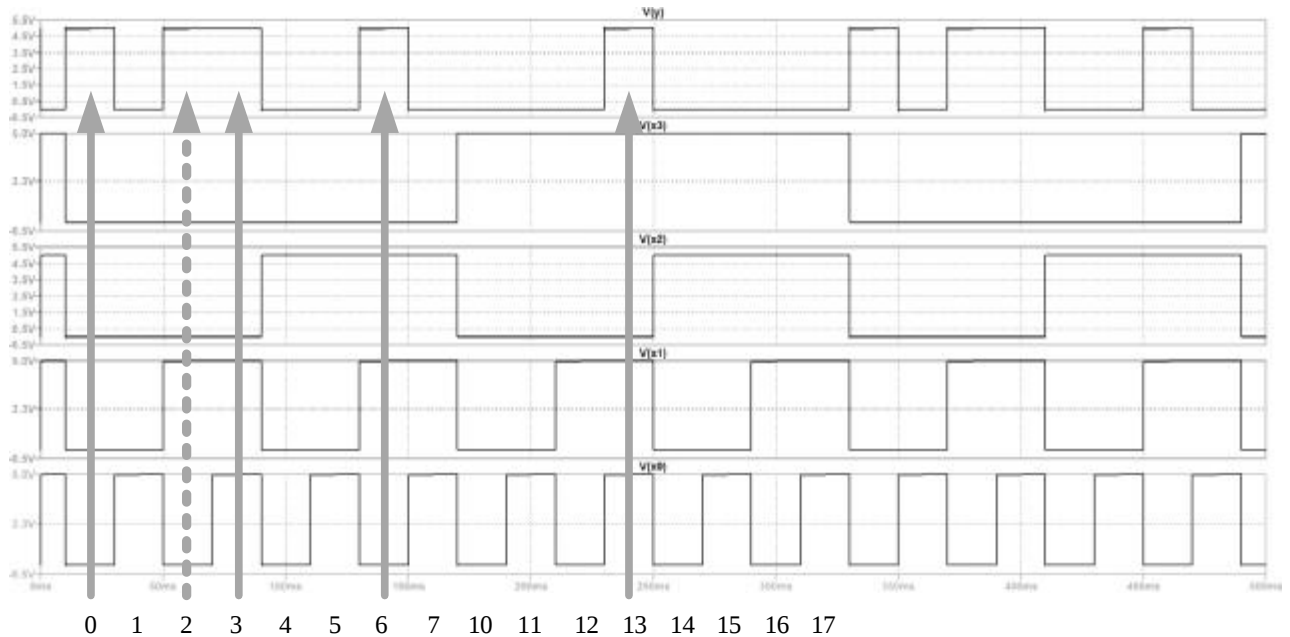


Рис. 36. Результат моделирования схемы минимизированной ФАЛ

Как видим, результат отличается наличием 1 в состоянии 2 для минимизированной ФАЛ. Это объясняется тем, что неопределенное состояние 2 было включено в минимизацию.

Содержание работы

1. Набрать схему реализации неминимизированной ФАЛ из библиотечных элементов.
2. Установить параметры сигнала прямоугольной формы с выхода импульсного генератора.
3. Запустить процесс моделирования схемы.
4. Добавить на диаграмму сигналы с выхода счетчика и выход функции.
5. Набрать схему реализации минимизированной ФАЛ из библиотечных элементов.
6. Запустить процесс моделирования схемы.
7. Добавить на диаграмму сигналы с выхода счетчика и выход функции

Содержание отчёта

Отчёт должен содержать результаты исследования двух схем по следующим пунктам.

1. Две принципиальные схемы исследований реализаций ФАЛ (неминимизированной и минимизированной).
2. Временные диаграммы работы схем реализации ФАЛ.
3. Формулы, описывающие заданную неминимизированную ФАЛ в аналитическом и числовом виде.
4. Формулы, описывающие заданную ФАЛ после минимизации в аналитическом виде.
5. Выводы по результатам выполнения работы.

ЛАБОРАТОРНАЯ РАБОТА 3. ПОЛНЫЕ ДВОИЧНЫЕ ДЕШИФРАТОРЫ - ДЕМУЛЬТИПЛЕКСОРЫ

Цель работы: изучение принципов построения полных двоичных дешифраторов, методов увеличения размерности и примеров их практического использования.

Определение: полным двоичным дешифратором называется комбинационная схема, имеющая n входов и 2^n выходов и реализующая на каждом выходе ФАЛ, представляющую собой конституенту 1 $K_i(X)$ n переменных, где $X = X_{n-1}X_{n-2}\dots X_1X_0$ – двоичный код набора входных переменных, $i = 0, 1, 2, \dots, 2^{n-1}$.

Таким образом, в полном дешифраторе каждому коду наборов входного сигнала соответствует выходной сигнал, равный единице только на одном выходе.

Дешифратор может быть и неполным, реализующим только $m < 2^n$ конституент 1.

Дешифраторы являются преобразователями кодов, выполняющих преобразование входных двоичных кодов в выходной **унитарный код**. Уни-

тарный код двоичного n – разрядного числа представляется 2^n разрядами, только один из разрядов которого равен 1. Условное описание дешифратора задаётся формулой $n \times m$ (n на m), для полного дешифратора это формула $n \times 2^n$.

Описание схемы исследования. На рис. 37 приведена схема исследования полного дешифратора-демультиплексора 3×8 . Для задания всех возможных наборов значений входных переменных используется четырехразрядный двоичный счетчик 74НС393, тактируемый от внешнего импульсного сигнала.

В состав дешифратора-демультиплексора входит 4 инвертора 74НС04 и 8 4-входовых логических элементов Шеффера (И-НЕ) 74НС20.

Сигнал X_3 с выхода счетчика имитирует сигнал разрешения работы OE с высоким активным уровнем.

Этот сигнал в схеме интерпретируется как входной сигнал, который демультиплексируется на все выходы дешифратора. Активный уровень выходного сигнала появляется на том выходе, адрес которого указан кодом на входах $X_2X_1X_0$ дешифратора.

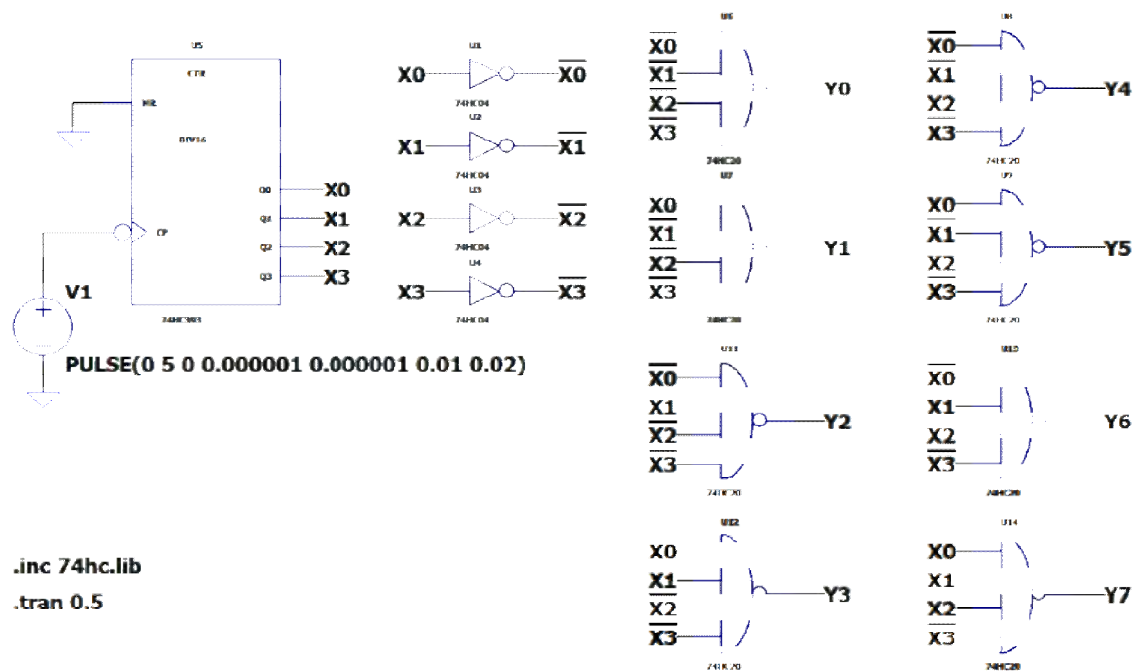


Рис. 37. Схема исследования дешифратора 3×8

Для получения полной временной диаграммы работы дешифратора-демультиплексора на экране график необходимо вывести 4 сигнала с выходов двоичного счетчика и 8 сигналов с выходов ЛЭ И-НЕ. Итого – 12 графиков (рис. 38).

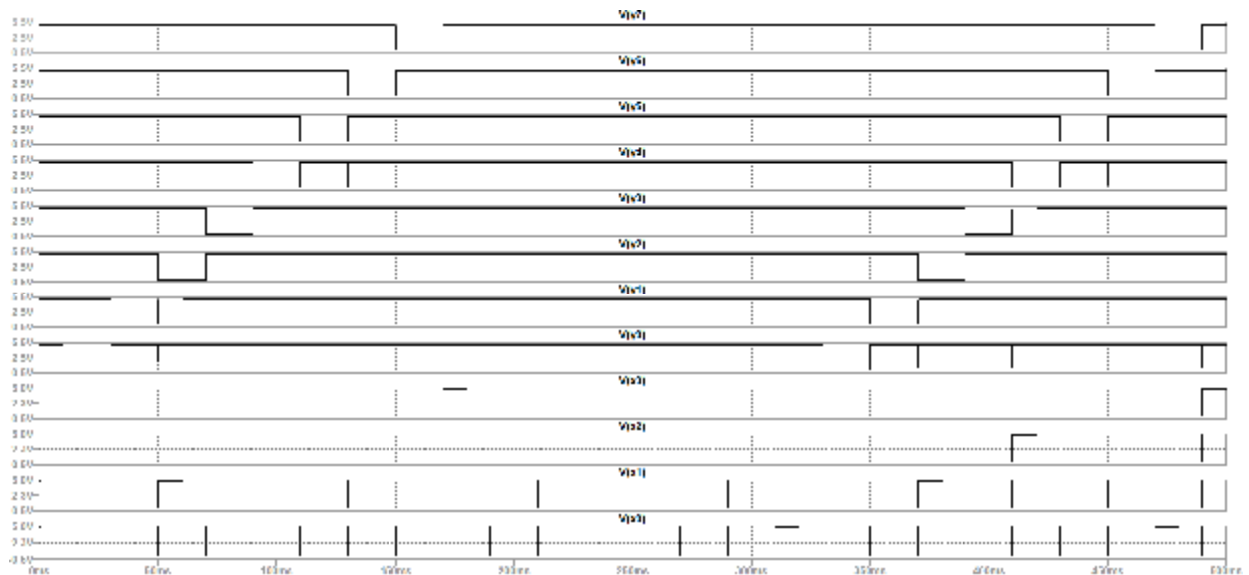


Рис. 38. Моделирование работы дешифратора

На рис. 39 показан пример, как расширения размерности дешифратора от 3×8 до 4×16 , так и практического применения дешифраторов для реализации функций алгебры логики, например для ФАЛ, рассмотренной в лабораторной работе № 2.

Для такой реализации необходимо доопределить частично определённую ФАЛ, как имеющую нулевые значения на неопределённых наборах и не минимизировать имеющуюся функцию, поскольку в состав полного дешифратора входят логические элементы, реализующие все конstituенты 0 от заданного количества логических переменных. Доопределение ФАЛ, как имеющей нулевые значения в неопределённых состояниях уменьшает количество входов у дизъюнктора, выполняющего объединение выходных активных сигналов дешифратора низкого уровня.

Аналитическое описание реализуемой ФАЛ имеет вид

$$Y = 0 \cup 3 \cup 6 \cup 13$$

Входы разрешения работы E1 и E2 имеют инверсную логику, и дешифратор становится активным при нулевом сигнале на них. В течение первых 8 состояний входных переменных переменная X3 имеет нулевое значение, и активным является верхний дешифратор. Когда переменная X3 меняет свое состояние с 0 на 1, сигнал X3, поданный через инвертор на входы разрешения работы нижнего дешифратора, делает микросхему нижнего дешифратора активной. Вход E3 разрешения работы дешифраторов имеет прямую логику, поэтому для работы дешифраторов на E3 необходимо подать напряжение, соответствующее логической единице (на схеме – сигнал VCC).

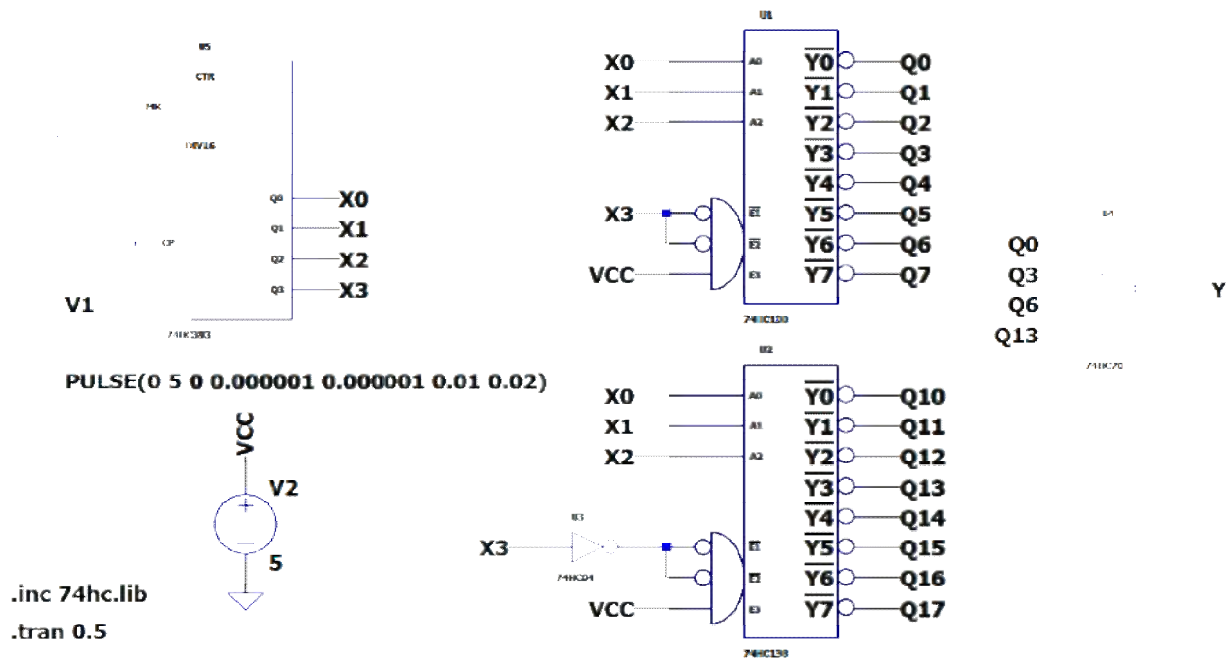


Рис. 39. Реализация ФАЛ на дешифраторах 74НС138

Результат моделирования приведен на рис. 40. Как видим, выход схемы Y имеет единичное значение в состояниях 0, 3, 6 и 13.

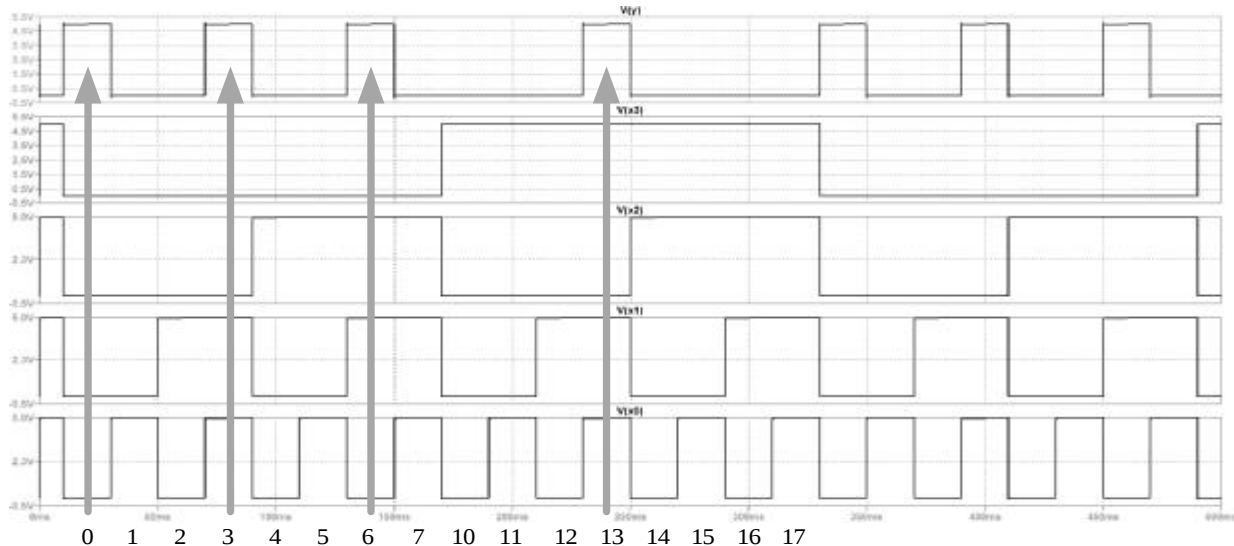


Рис. 40. Результат моделирования реализации ФАЛ на дешифраторе

Содержание работы

1. Набрать схему реализации полного дешифратора-демультиплексора 3×8 из библиотечных элементов программы (см. рис. 39).
2. Установить параметры сигнала прямоугольной формы для тактирования двоичного счетчика.
3. Запустить процесс моделирования схемы.
4. Изучить временные диаграммы работы полного дешифратора-демультиплексора 3×8 с помощью временных диаграмм (рис. 40).
5. Набрать схему реализации принципиальной схемы ФАЛ из двух библиотечных интегральных схем дешифраторов-демультиплексоров 3×8

74138, которые образуют дешифратор-демультиплексор 4×16 , одной 4-входовой схемы И-НЕ 74НС20 для своего варианта из работы 2.

6. Запустить процесс моделирования схемы.

7. Изучить временные диаграммы работы схемы реализации полностью определённой неминимизированной ФАЛ.

Содержание отчёта

Отчёт должен содержать результаты исследования двух схем по следующим пунктам.

1. Принципиальную схему исследования дешифратора-демультиплексора 3×8 , собранного из ЛЭ И-НЕ.

2. Временную диаграмму работы дешифратора-демультиплексора 3×8 .

3. Формулы, описывающие заданную полностью определённую неминимизированную ФАЛ в аналитическом и числовом виде.

4. Принципиальную схему исследования реализации полностью определённой неминимизированной ФАЛ на дешифраторах-демультиплексорах 3×8 восьмивходовом ЛЭ И-НЕ.

5. Временную диаграмму работы схемы реализации заданной ФАЛ на дешифраторах-демультиплексорах 3×8 и восьмивходовом ЛЭ И-НЕ.

6. Выводы по результатам выполнения работы.

ЛАБОРАТОРНАЯ РАБОТА 4. МУЛЬТИПЛЕКСОРЫ

Цель работы: изучение принципов построения мультиплексоров, методов увеличения размерности и примеров их практического использования, реализация ФАЛ на мультиплексорах.

Определение: мультиплексором называется комбинационная схема, имеющая $m + 2^m$ входов и один выход, где m – количество адресных входов, а 2^m – количество информационных входов мультиплексора. Адреса представляются в двоичном коде, и им присваивается номер j . Каждому адресу с номером j соответствует свой информационный вход A_j , сигнал с которого при данном адресе проходит на выход.

Основным назначением мультиплексора является коммутация 2^m информационных входных сигналов на один выход под воздействием адресных сигналов. Таким образом, мультиплексор выполняет функцию

$$f = \bigvee_{j=0}^{m-1} A_j \cdot K_j(X),$$

где X – двоичный код адреса; A_j – входные информационные сигналы мультиплексора; $K_j(X)$ – конstituенты 1, принимающие значения, равные 1, на единственном наборе переменных $X = j$. Условное описание мультиплексора задаётся формулой $2^m \times 1$.

Примерами являются мультиплексоры: 2×1 , 4×1 , 8×1 , 16×1 .

Описание схемы исследования

На рис. 41 приведена схема исследования мультиплексора 8×1. Для задания всех возможных кодов адреса (наборов значений входных переменных) используется двоичный 4-разрядный таймер-счетчик 74НС393.

В состав мультиплексора входит три инвертора и пять четырёхвходовых ЛЭ Шеффера (И-НЕ). На инверторах и четырёх ЛЭ И-НЕ собран полный дешифратор-демультиплексор 3×8, по одному входу этих ЛЭ А0...А7 используются для подачи информационных сигналов. Если на информационный вход подана логическая «1», то в соответствующий момент времени на выходе появится логическая «1».

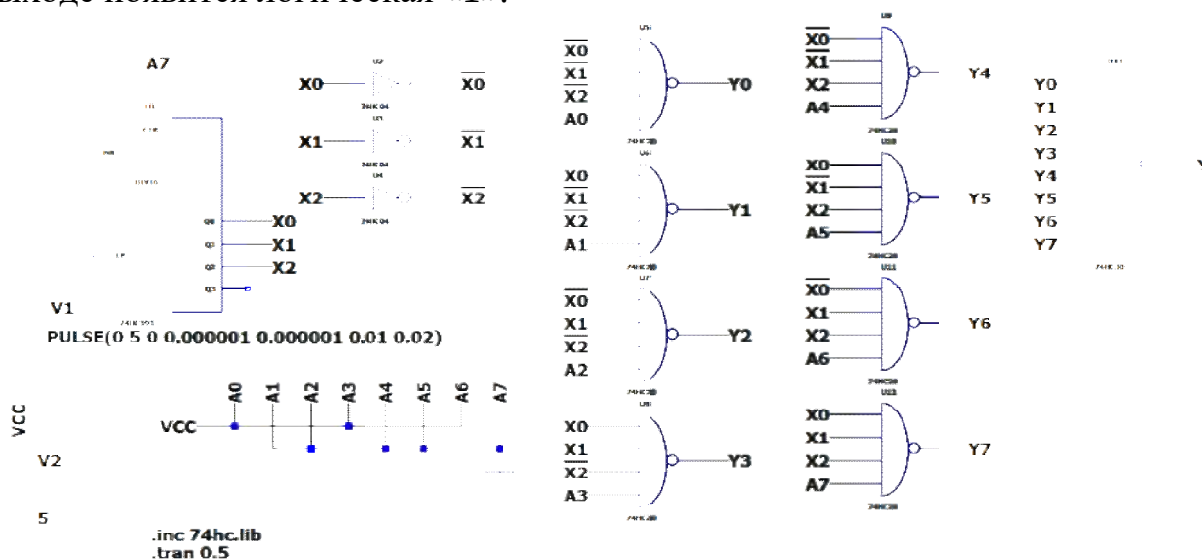


Рис. 41. Схема исследования мультиплексора 8×1

Для отчета информационные входы подключить таким образом, чтобы на них была реализована первая часть ФАЛ из варианта (состояния 0 – 7). Рассмотрим для примера выражение для ФАЛ

$$Y = 0 \cup 3 \cup 6 \cup 13.$$

Информационные входы А0, А3, А6 входят в первую половину двоичных состояний, поэтому подключаются к логической «1». Результат моделирования схемы приведен на рис. 42.

Вторая часть работы включает реализацию полной ФАЛ мультиплексорах 8x1 74НС151. Для того чтобы на этих элементах реализовать ФАЛ четырех переменных используются входы разрешения работы. Верхний элемент мультиплексора работает при сигнале на линии X3, равном логическому «0». Когда сигнал X3 становится равным логической «1», разрешается работа нижнего мультиплексора, на вход разрешения работы которого приходит инвертированный сигнал X3.

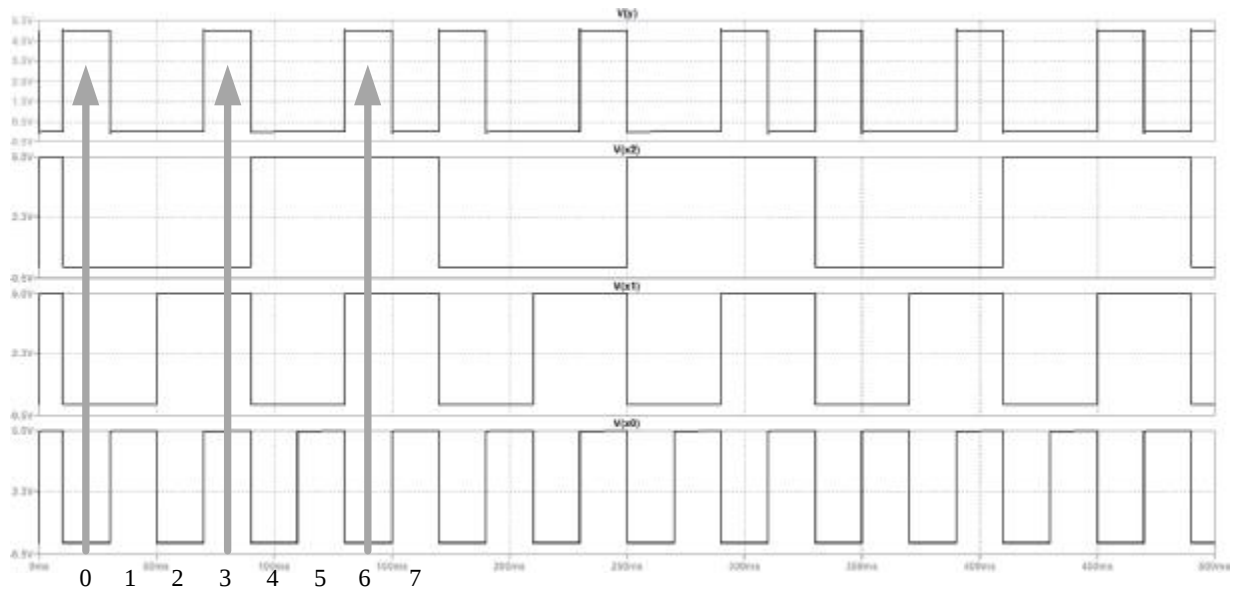


Рис. 42. Результат моделирования схемы мультиплексора

На рис. 43 приведена схема реализации ФАЛ, записанной выражением (1), на мультиплексорах.

На рис. 44 приведен результат моделирования схемы, представленной на рис. 43.

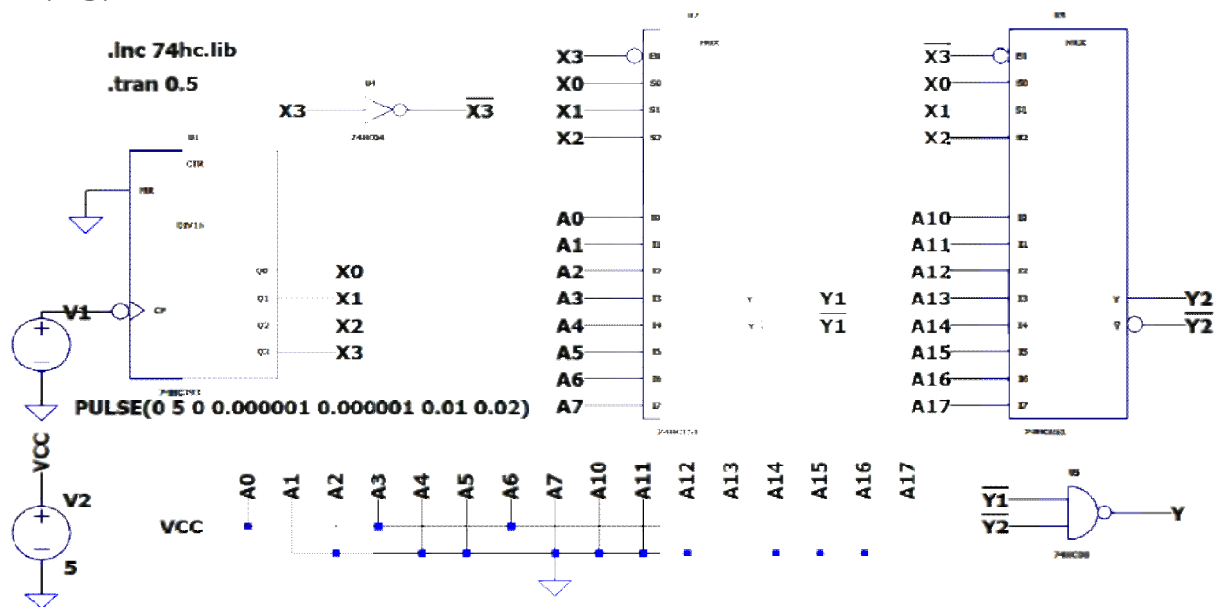


Рис. 43. Схема реализации ФАЛ с помощью мультиплексора

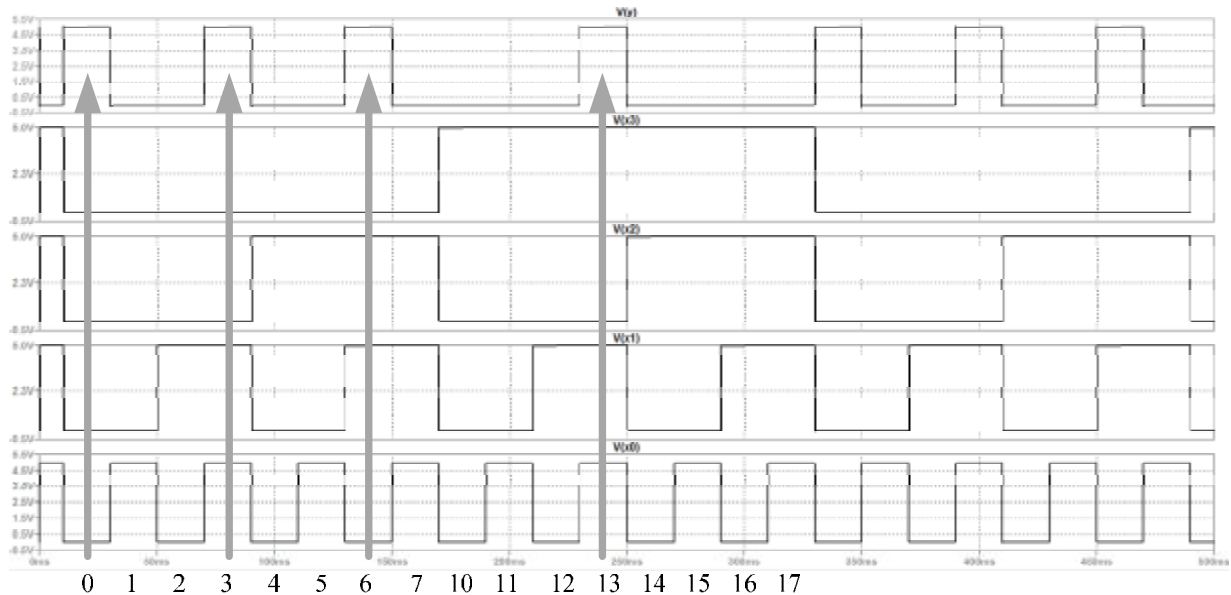


Рис. 44. Результат моделирования схемы реализации ФАЛ

Третья часть работы представляет собой реализацию ФАЛ на мультиплексоре с использованием универсальных логических модулей УЛМ.

Запишем выражение для ФАЛ с учетом неопределенных состояний:

$$Y = 0 \cup 3 \cup 6 \cup 13 \cup (2) \cup (11) \cup (14).$$

Минимизируем ФАЛ. Можно воспользоваться результатом минимизации из работы №2 (см. рис. 32).

$$Y = \overline{X_2} X_1 X_0 \cup \overline{X_3} \overline{X_2} \overline{X_0} \cup \overline{X_3} X_1 \overline{X_0}$$

Разложим заданную ФАЛ в ряд Шеннона по двум переменным, имеющим наибольшее число вхождений в минимизированную ФАЛ. Наибольшее число вхождений в минимизированную ФАЛ имеет переменная X_0 , поэтому она обязательно должна участвовать в разложении. Таким образом, можно выполнить разложение по $X_3 X_0$, $X_2 X_0$ или $X_1 X_0$.

Исходная ФАЛ в переменных состояния имеет вид:

$$Y = \overline{X_3} \overline{X_2} \overline{X_1} \overline{X_0} \cup \overline{X_3} \overline{X_2} X_1 X_0 \cup \overline{X_3} X_2 X_1 \overline{X_0} \cup X_3 \overline{X_2} X_1 X_0 \cup (\overline{X_3} \overline{X_2} X_1 \overline{X_0}) \cup (X_3 \overline{X_2} \overline{X_1} X_0) \cup (X_3 X_2 \overline{X_1} \overline{X_0})$$

Разложение в ряд Шеннона по $X_3 X_0$:

$$Y = \overline{X_3} \overline{X_0} [\overline{X_2} \overline{X_1} \cup X_2 X_1 \cup (\overline{X_2} X_1)] \cup \overline{X_3} X_0 [\overline{X_2} X_1] \cup X_3 \overline{X_0} [(X_2 \overline{X_1})] \cup X_3 X_0 [\overline{X_2} X_1 \cup (\overline{X_2} \overline{X_1})]$$

В квадратных скобках указываются остаточные функции, полученные при разложении. В круглых скобках указываются остаточные функции неопределенных состояний.

Для проверки: в исходной функции было 7 слагаемых. После разложения должно получиться 7 остаточных функций.

Теперь минимизируем остаточные функции. Получаем:

$$Y = \overline{X_3} \overline{X_0} [\overline{X_2} \cup X_1] \cup \overline{X_3} X_0 [\overline{X_2} X_1] \cup X_3 \overline{X_0} [0] \cup X_3 X_0 [\overline{X_2}]$$

$$Y = \overline{X_3} \overline{X_0} [\overline{X_2} \overline{X_1}] \cup \overline{X_3} X_0 [\overline{X_2} X_1] \cup X_3 \overline{X_0} [0] \cup X_3 X_0 [\overline{X_2}]$$

На рис. 45 представлен пример реализации ФАЛ на мультимплексоре 4×1 74НС153. Остаточные функции представляют собой сигналы, подаваемые на информационные входы мультимплексора. Сигналы, по которым производилось разложение в ряд Шеннона, подаются на адресные входы мультимплексора, причем старший сигнал обязательно соответствует старшему адресному входу.

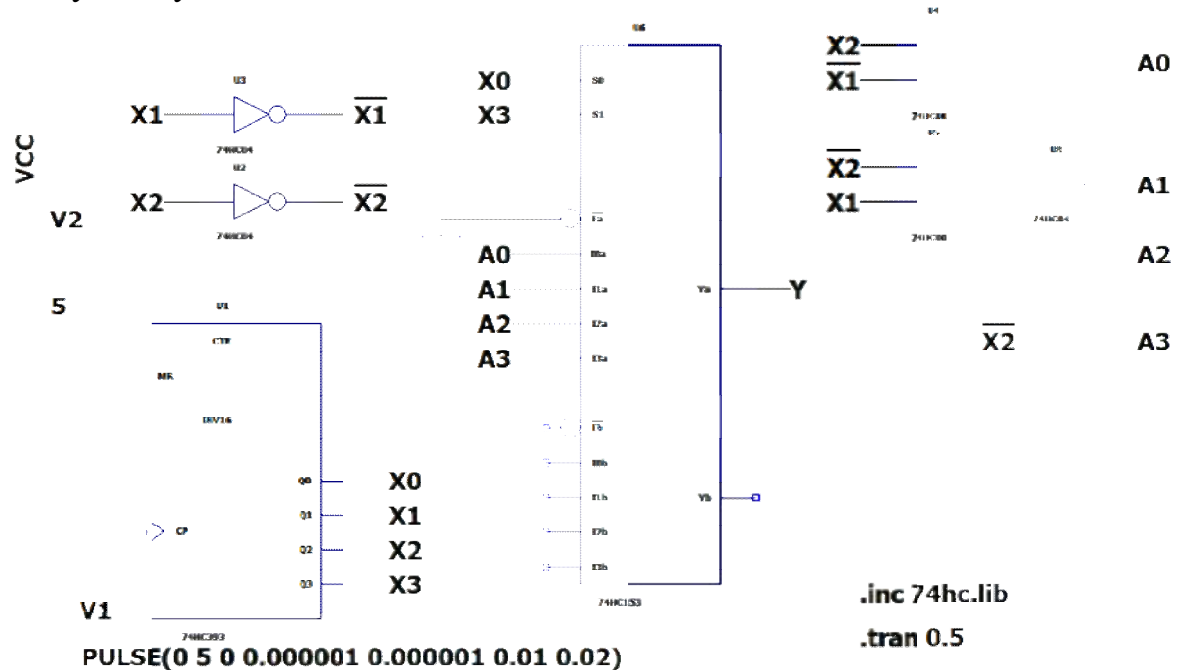


Рис. 45. Реализация ФАЛ на УЛМ

На рис. 46 приведен результат моделирования схемы (рис. 45). Пунктиром отмечены номера неопределенных состояний функции, которые вошли в минимизацию при разложении.



Рис. 46. Результат моделирования ФАЛ на УЛМ

Содержание работы

1. Набрать схему реализации мультиплексора 8×1 из библиотечных элементов программы. Для моделирования схемы установить единичными те информационные состояния, которые соответствуют единичным значениям варианта из первой части таблицы состояний.
2. Запустить процесс моделирования схемы.
3. Изучить временные диаграммы работы мультиплексора 8×1 .
4. Набрать схему реализации ФАЛ из двух библиотечных интегральных схем мультиплексоров 8×1 74НС151 и двухвходового логического элемента И-НЕ.
5. Запустить процесс моделирования схемы.
6. Изучить временные диаграммы работы схемы реализации полностью определённой неминимизированной ФАЛ.
7. Выполнить разложение в ряд Шеннона заданной ФАЛ.
8. Набрать схему реализации ФАЛ из интегральной схемы мультиплексора 4×1 74НС153 и остаточных функций.
9. Запустить процесс моделирования схемы.
10. Изучить временные диаграммы работы схемы реализации ФАЛ на УЛМ.

Содержание отчёта

Отчёт должен содержать результаты исследования трех схем по следующим пунктам.

1. Принципиальную схему исследования мультиплексора 8×1 , собранного из логических элементов И-НЕ.
2. Временную диаграмму работы мультиплексора 8×1 .
3. Формулы, описывающие заданную полностью определённую неминимизированную ФАЛ в аналитическом и числовом виде.
4. Принципиальную схему исследования реализации полностью определённой неминимизированной ФАЛ на мультиплексорах 8×1 и двухвходовом логическом элементе И-НЕ.
5. Временную диаграмму работы схемы реализации заданной ФАЛ на двух мультиплексорах 8×1 и двухвходовом логическом элементе И-НЕ.
6. Формулы, описывающие разложение заданной ФАЛ в ряд Шеннона.
7. Принципиальную схему исследования реализации заданной ФАЛ на мультиплексоре 4×1 и логических элементах, реализующих остаточные функции.
8. Временную диаграмму работы схемы реализации заданной ФАЛ на мультиплексоре 4×1 и логических элементах, реализующих остаточные функции.
9. Выводы по результатам выполнения работы.

ЛАБОРАТОРНАЯ РАБОТА 5. ПРЕОБРАЗОВАТЕЛИ КОДОВ

Цель работы: изучение принципов построения преобразователей кодов (схем преобразования любых заданных входных кодов в требуемые выходные коды), и примеров их практического использования.

Описание: Преобразователь кодов осуществляет однозначное преобразование каждого m -местного набора входных переменных в n -местный набор выходных ФАЛ.

Формой задания описания преобразователя кодов является таблица истинности для системы ФАЛ. Поскольку таблица предполагает полный перебор вариантов входных кодов, она способна задавать любой закон работы преобразователя.

Варианты заданий представлены в табл. 15.

Таблица 15

Варианты выполнения лабораторной работы

Out\In	1248	2421	5121	53-21	75-31	5421	642-1	462-1
1248	*	1	2	3	4	5	6	7
2421	8	*	9	10	11	12	13	14
5121	15	16	*	17	18	19	20	21
53-21	22	23	24	*	25	26	27	28
75-31	29	30	31	32	*	33	34	35
5421	36	37	38	39	40	*	41	42
642-1	43	44	45	46	47	48	*	49
462-1	50	51	52	53	54	55	56	*

Реализация схемы преобразователя кодов может быть выполнена с использованием любой известной элементной базы, например, ЛЭ (И, ИЛИ, НЕ), дешифраторов, мультиплексоров, постоянных запоминающих устройств (ПЗУ), программируемых логических матриц (ПЛМ).

В табл. 15 представлены коды в двоично-кодированной десятичной системе счисления (ДКДС), в которых каждая десятичная цифра от 0 до 9 изображается тетрадой из двоичных символов, вес каждого символа тетрады задан десятичным числом на соответствующей позиции тетрады. Входной код преобразователя находится в первой строке табл. 15, помеченной **In**, а выходной код находится в столбце, помеченном символом **Out**. Номер варианта, задаваемого для реализации преобразователя кода, помещён на пересечении строки и таблицы.

Построение преобразователя кодов. Для построения преобразователя кодов строится таблица соответствия (табл. 16), в которой записывается полный набор входных и соответствующий набор выходных слов. Если входные и выходные слова записаны в двоичном коде, то синтез преобразователя кодов сводится к нахождению для каждого разряда выходного слова ФАЛ, устанавливающей связь данного разряда с входными наборами двоичных переменных. Нахождение такой связи и минимизация булевого вы-

ражения осуществляется с помощью карт Карно (диаграмм Вейча). Полученная функция преобразуется к виду, удобному для реализации в заданном (выбранном) элементном базисе.

В обозначениях кодов 8421, 63-21 и т. д. указан десятичный вес p_i двоичной единицы x_i соответствующего разряда. Рассмотрим преобразователь кодов 63-21 в 5121. Для построения такого преобразователя строится таблица, в левой части которой перебираются все состояния от 0_8 (0000) до 17_8 (1111). Каждое состояние соответствует некоторой десятичной цифре в весах входного кода 63-21. Например, входной код 1110 соответствует десятичной цифре

$$1110 \rightarrow 1*6+1*3+(-2)*1+1*0 = 7$$

Аналогичная таблица строится отдельно для выходного кода, после чего приводится в соответствие каждая десятичная цифра входного кода комбинации выходного. Если какая-либо цифра может быть представлена во входном (или выходном) коде несколькими способами, то задействуется только одна комбинация весов. Остальные состояния, определяющие данную цифру, считаются неопределенными, например

$$0001 \rightarrow 6*0+3*0+(-2)*0+1*1 = 1;$$

$$0110 \rightarrow 6*0+3*1+(-2)*1+1*0 = 1.$$

Задействовано будет только одно из этих состояний (на выбор).

Если в результате комбинации весов не получается значение десятичной цифры, то данная комбинация также является неопределенной, например

$$0010 \rightarrow 6*0+3*0+(-2)*1+1*0 = -2.$$

Таблица 16

Таблица соответствия для преобразователя кодов 63–21 в 5121

№ ₈	Входной код				цифра	Выходной код			
	6	3	–2	1		5	1	2	1
	x_3	x_2	x_1	x_0		y_3	y_2	y_1	y_0
0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	1	0	0	0	1
2	0	0	1	0	-2	*	*	*	*
3	0	0	1	1	-1	*	*	*	*
4	0	1	0	0	3	0	0	1	1
5	0	1	0	1	4	0	1	1	1
6	0	1	1	0	1!	*	*	*	*
7	0	1	1	1	2	0	0	1	0
10	1	0	0	0	6	1	0	0	1
11	1	0	0	1	7	1	0	1	0
12	1	0	1	0	4!	*	*	*	*
13	1	0	1	1	5	1	0	0	0
14	1	1	0	0	9	1	1	1	1
15	1	1	0	1	10!	*	*	*	*
16	1	1	1	0	7!	*	*	*	*
17	1	1	1	1	8	1	1	1	0

Для проверки: поскольку во входном коде мы должны набрать ровно 10 цифр, то в выходном коде должно остаться 6 неопределенных состояний.

Каждая выходная функция y_i минимизируется по диаграмме Вейча (согласно рис. 47).

$$Y3 = 10 \cup 11 \cup 13 \cup 14 \cup 17 \cup (2) \cup (3) \cup (6) \cup (12) \cup (15) \cup (16)$$

$$Y2 = 5 \cup 14 \cup 17 \cup (2) \cup (3) \cup (6) \cup (12) \cup (15) \cup (16)$$

$$Y1 = 4 \cup 5 \cup 7 \cup 11 \cup 14 \cup 17 \cup (2) \cup (3) \cup (6) \cup (12) \cup (15) \cup (16)$$

$$Y0 = 1 \cup 4 \cup 5 \cup 10 \cup 14 \cup (2) \cup (3) \cup (6) \cup (12) \cup (15) \cup (16)$$

Y3:

	$\overline{X1}$				
	7	5	1	3	
$\overline{X0}$	6	4	0	2	$\overline{X3}$
	16	(14)	(10)	12	
	17	15	(11)	(13)	
	$X2$				

$$Y3 = X3$$

Y2:

	$\overline{X1}$				
	7	(5)	1	3	
$\overline{X0}$	6	4	0	2	$\overline{X3}$
	16	(14)	10	12	
	(17)	15	11	13	
	$X2$				

$$Y2 = X2 \overline{X1} X0 \cup X3 X2$$

Y1:

	$\overline{X1}$				
	(7)	(5)	1	3	
$\overline{X0}$	6	(4)	0	2	$\overline{X3}$
	16	(14)	10	12	
	(17)	15	(11)	13	
	$X2$				

$$Y1 = X2 \cup X3 \overline{X1} X0$$

Y0:

	$\overline{X1}$				
	7	(5)	(1)	3	
$\overline{X0}$	6	(4)	0	2	$\overline{X3}$
	16	(14)	(10)	12	
	17	15	11	13	
	$X2$				

$$Y0 = X3 \overline{X0} \cup \overline{X3} \overline{X1} X0 \cup X2 \overline{X1}$$

Рис. 47. Минимизация ФАЛ преобразователя кодов

Описание схемы исследования

Для задания всех возможных комбинаций входного кода (наборов значений входных переменных) используются двоичный счетчик 74НС393. Схема преобразователя кодов построена на ЛЭ И-НЕ после выполнения независимой минимизации всех ФАЛ системы.

Для реализации запишем выражения для ФАЛ в базисе И-НЕ:

$$Y3 = X3$$

$$Y2 = \overline{\overline{X2} \overline{X1} X0 \cap \overline{X3} \overline{X2}}$$

$$Y1 = \overline{\overline{X2} \cap \overline{X3} \overline{X1} X0}$$

$$Y0 = \overline{\overline{X3} \overline{X0} \cap \overline{X3} \overline{X1} X0 \cap \overline{X2} \overline{X1}}$$

На рис. 48 приведена схема реализации преобразователя кодов, рассмотренного в качестве примера. На рис. 49 приведен результат моделирования схемы исследования.

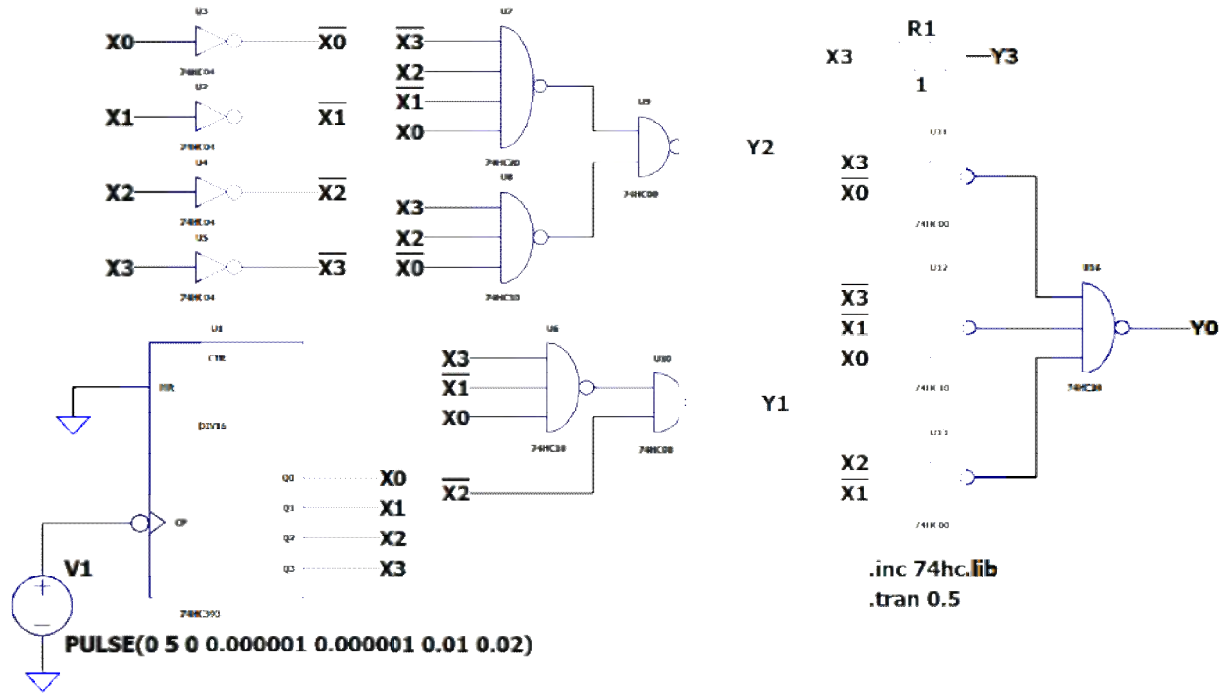


Рис. 48. Схема исследования преобразователя кодов

Рис. 49. Результат моделирования преобразователя кодов

По результатам моделирования заполняем полную таблицу истинности преобразователя кодов по экспериментальным данным (табл. 17).

Таблица 17

Таблица соответствия для преобразователя кодов 63–21 в 5121

№ ₈	Входной код				цифра	Выходной код			
	6	3	–2	1		5	1	2	1
	x ₃	x ₂	x ₁	x ₀		y ₃	y ₂	y ₁	y ₀
0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	1	0	0	0	1
2	0	0	1	0	-2	0	0	0	0
3	0	0	1	1	-1	0	0	0	0
4	0	1	0	0	3	0	0	1	1
5	0	1	0	1	4	0	1	1	1
6	0	1	1	0	1!	0	0	1	0
7	0	1	1	1	2	0	0	1	0
10	1	0	0	0	6	1	0	0	1
11	1	0	0	1	7	1	0	1	0
12	1	0	1	0	4!	1	0	0	1
13	1	0	1	1	5	1	0	0	0
14	1	1	0	0	9	1	1	1	1
15	1	1	0	1	10!	1	1	1	1
16	1	1	1	0	7!	1	1	1	1
17	1	1	1	1	8	1	1	1	0

Содержание работы

1. Заполнить таблицу истинности для заданного варианта преобразователя кодов.
2. Выполнить независимую минимизацию всех выходных ФАЛ преобразователя кодов.
3. Записать все минимизированные выходные ФАЛ в аналитической форме для базиса И-НЕ.
4. Набрать схему реализации преобразователя заданного входного кода в требуемый выходной код из библиотечных элементов программы.
5. Запустить процесс моделирования схемы.
6. Заполнить таблицу истинности для заданного варианта преобразователя кодов по экспериментальным данным.

Содержание отчёта

Отчёт должен содержать результаты исследования преобразователя кодов по следующим пунктам.

1. Таблица истинности для заданного варианта преобразователя кодов.
2. Описание минимизации всех ФАЛ с использованием диаграмм Вейча.
3. Аналитическое описание минимизированных ФАЛ для базиса И-НЕ.

4. Принципиальная схема исследования реализации заданного варианта преобразователя кода, собранного из ЛЭ И-НЕ.

5. Таблица истинности, полученную в результате экспериментального исследования схемы реализации заданного варианта шифратора, построенного на логических элементах И-НЕ.

6. Выводы по результатам выполнения работы.

ЛАБОРАТОРНАЯ РАБОТА 6. ДВОИЧНЫЕ СЧЕТЧИКИ

Цель работы: изучение принципов построения и функционирования счётчиков как цифровых автоматов специального назначения для подсчёта количества входных импульсов.

Определение: счётчиком называется цифровой автомат, служащий для формирования многоразрядных двоичных слов (кодов) соответствующих количеству входных импульсов. По мере поступления входных импульсов счётчик последовательно перебирает свои состояния в определённом для данной схемы порядке. Количество возможных состояний счетчика называется модулем пересчёта M , основанием пересчёта или ёмкостью счётчика. Одно из возможных состояний счётчика принимается как начальное или нулевое.

Если счётчик начал считать с начального состояния и через каждые M входных сигналов в нём снова устанавливается начальное состояние, а на выходе счётчика формируется сигнал M -ичного переноса P , то такой счётчик называется счётчиком–делителем частоты. Частота следования M -ичного переноса P в таком счётчике меньше частоты входных импульсов ровно в M раз. Различные схемы счётчиков могут перебирать свои состояния в самом различном порядке. Порядок перебора кодов состояний счётчика определяется системой кодирования состояний счётчика как цифрового автомата. Чаще всего применяются двоичные счётчики, у которых порядок смены состояний триггеров соответствует последовательности двоичных чисел. Счетчик может перебирать свои состояния в возрастающем порядке чисел и тогда он называется **суммирующим**. Если счётчик перебирает свои состояния в порядке убывания двоичных чисел, то он называется **вычитающим**. Если под действием сигналов управления порядок перебора может изменяться на противоположный, то счётчик называется **реверсивным**.

Счётчики строятся с использованием синхронизированных триггеров. Если несколько триггеров блока памяти имеют общий синхросигнал, то они образуют синхронный счётчик. Несколько синхронных счётчиков соединённых по схеме подачи синхросигналов последовательно образуют асинхронный счётчик. В этом случае сигнал выходного переноса предыдущего счётчика используется как сигнал синхронизации последующего счётчика. Примером асинхронного суммирующего двоичного счётчика является микросхема 74НС393, используемая во всех предыдущих работах.

По описанию работы цифрового автомата (ЦА) для счёта импульсов, очевидно, что кроме блока памяти в состав этого ЦА входит и комбинационная схема, формирующая соответствующие алгоритму работы сигналы возбуждения блока памяти.

Блок памяти ЦА может быть построен, как на одноступенчатых (прозрачных) триггерах с двухфазной системой синхронизации двух узлов памяти, так и на двухступенчатых триггерах. Внешне при использовании двухступенчатых триггеров в составе цифрового автомата его схема выглядит проще. В лабораторной работе исследуется счётчик, построенный с использованием двухступенчатых триггеров JK в блоке памяти.

Описание схемы исследования

На рис. 54 изображена схема исследования синхронного суммирующего двоичного счётчика–делителя частоты по модулю 10 на двухступенчатых (непрозрачных) триггерах типа JK из библиотеки элементов программы. Кодированная таблица переходов этого счётчика – табл. 18.

Таблица 18

Таблица переходов счетчика с коэффициентом пересчета 10

Состояние	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001
Переход	0001	0010	0011	0100	0101	0110	0111	1000	1001	0000

Недостижимые коды состояний – 1010, 1011, 1100, 1101, 1110, 1111.

Кодированная таблица переходов строится в соответствии с табл. 19 и представлена в табл. 20.

Таблица 19

Таблица переходов JK-триггера

Q	Q ⁺	J	K
0	0	0	*
0	1	1	*
1	0	*	1
1	1	*	0

Таблица 20

Таблица переходов двоичного счетчика на JK-триггерах

Номер	Состояние	Переход	J3	J2	J1	J0	K3	K2	K1	K0
0	0000	0001	0	0	0	1	*	*	*	*
1	0001	0010	0	0	1	*	*	*	*	1
2	0010	0011	0	0	*	1	*	*	0	*
3	0011	0100	0	1	*	*	*	*	1	1
4	0100	0101	0	*	0	1	*	0	*	*
5	0101	0110	0	*	1	*	*	0	*	1
6	0110	0111	0	*	*	1	*	0	0	*
7	0111	1000	1	*	*	*	*	1	1	1

Номер	Состояние	Переход	J3	J2	J1	J0	K3	K2	K1	K0
10	1000	1001	*	0	0	1	0	*	*	*
11	1001	0000	*	0	0	*	1	*	*	1
12	1010	*****	*	*	*	*	*	*	*	*
13	1011	*****	*	*	*	*	*	*	*	*
14	1100	*****	*	*	*	*	*	*	*	*
15	1101	*****	*	*	*	*	*	*	*	*
16	1110	*****	*	*	*	*	*	*	*	*
17	1111	*****	*	*	*	*	*	*	*	*

Минимизированные с использованием диаграмм Вейча функции возбуждения JK триггеров счётчика имеют вид (рис. 50-53):

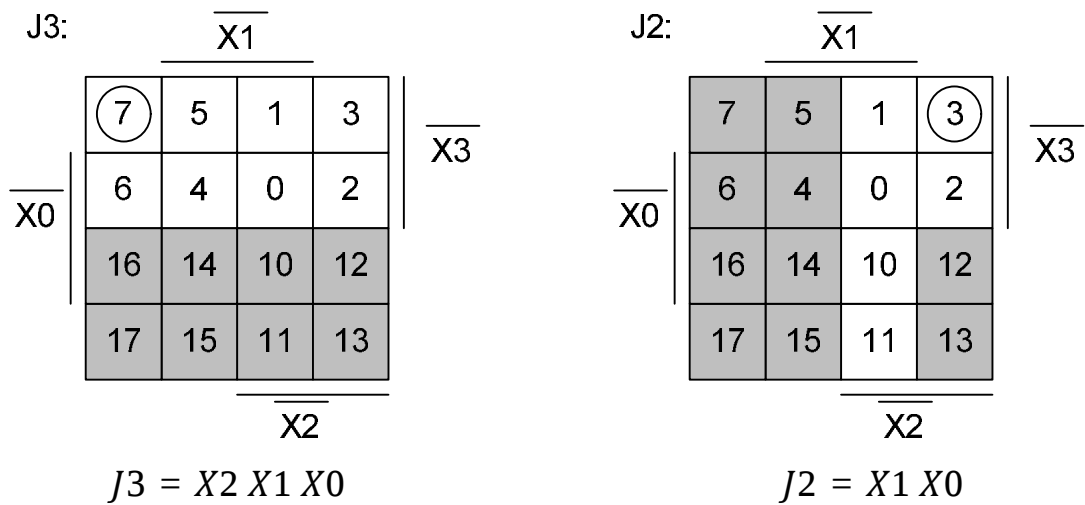


Рис. 50. Минимизация J3 и J2

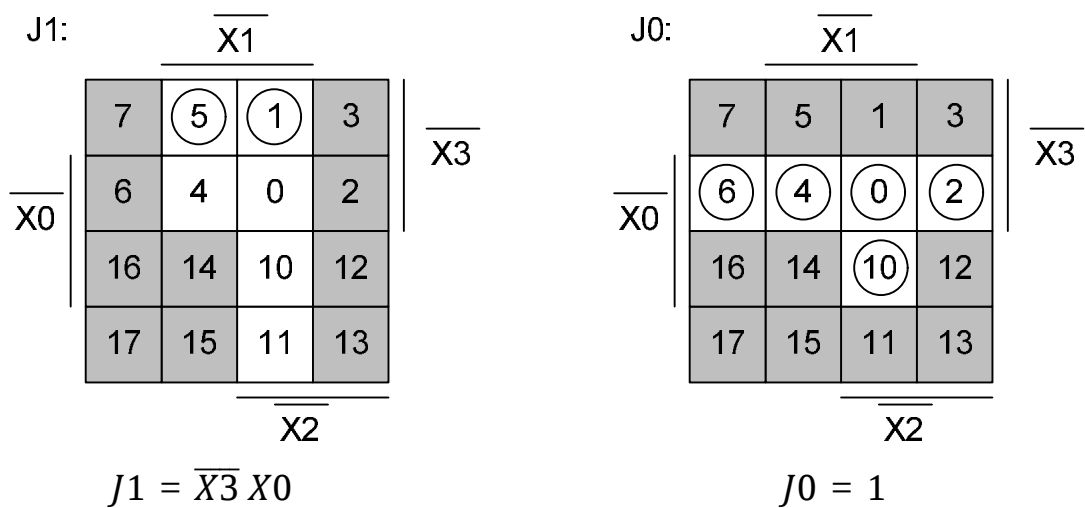


Рис. 51. Минимизация J1 и J0

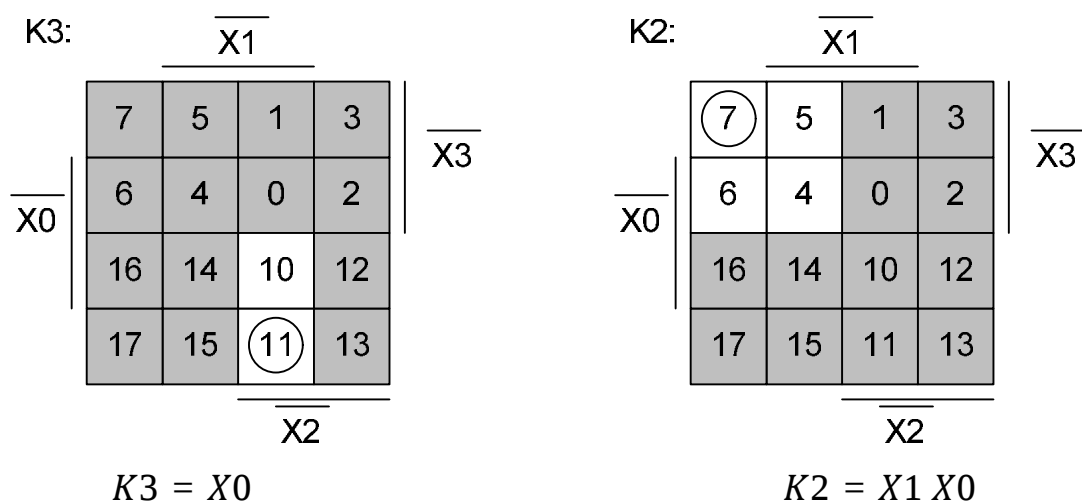


Рис. 52. Минимизация K3 и K2

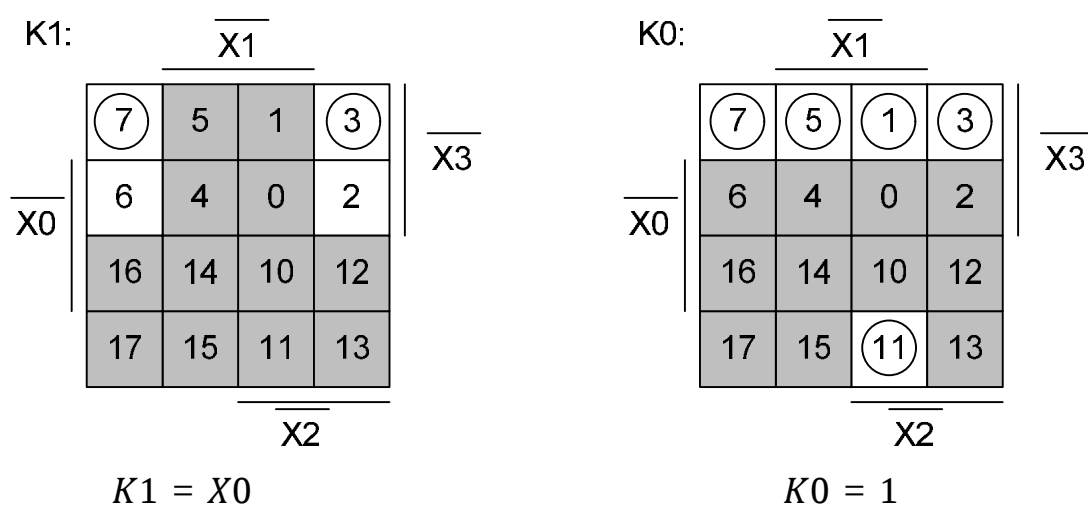


Рис. 53. Минимизация K1 и K0

По этим уравнениям и набирается комбинационная схема цифрового автомата – счётчика по модулю 10. Для построения счётчика используются триггеры типа JK 74НС107 со входами сброса, подключаемыми к уровню логической единицы. При включении двоичный счётчик по модулю 10 начинает последовательно перебирать все разрешённые состояния, никогда не переходя в какое-либо из запрещённых состояний.

Для контроля работы схемы следует вывести на анализатор выходы триггеров, обозначенные на схеме (см. рис. 54) как X0, X1, X2, X3.

На рис. 55 представлен результат моделирования схемы счетчика по модулю 10.

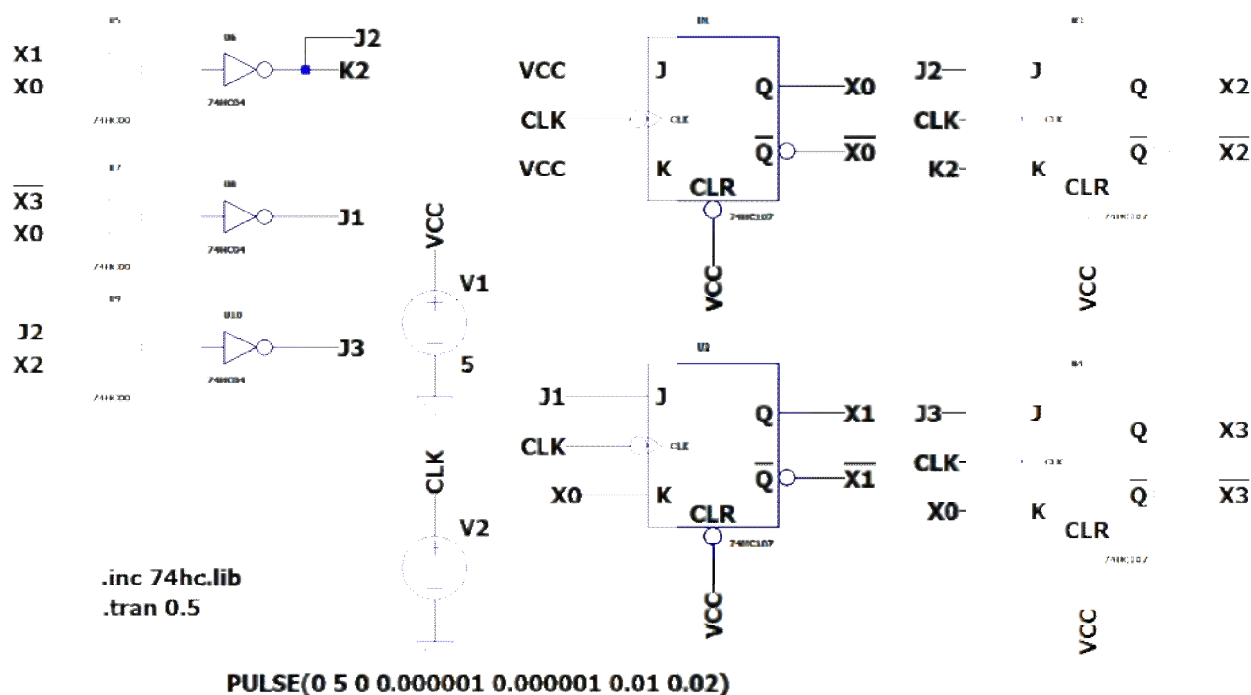


Рис. 54. Схема счётчика – делителя частоты по модулю 10

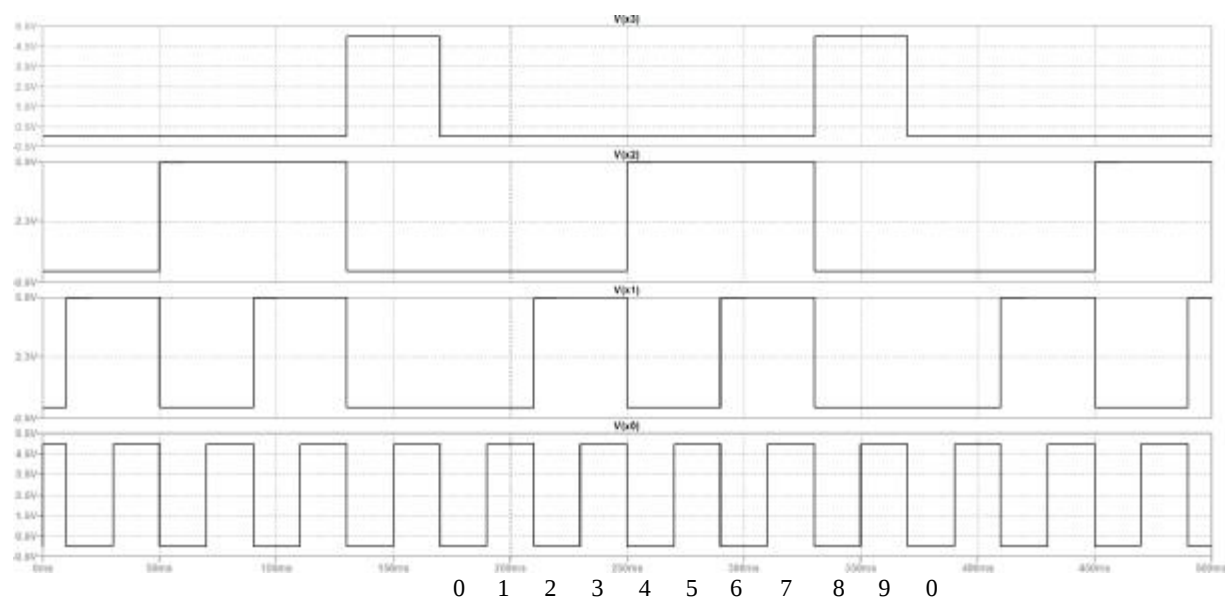


Рис. 55. Результат моделирования схемы счетчика по модулю 10

На рис. 56 изображена схема исследования счётчика в интегральном исполнении – микросхема 74НС393. Коэффициент пересчета для схемы задается по номеру варианта следующим образом:

$$K = \text{Var} \% 7 + 9,$$

где % - остаток от деления. В результате получится значение от 9 до 15. На рис. 57 приведена схема исследования для коэффициента пересчета 11 (вариант 2) = 1011.

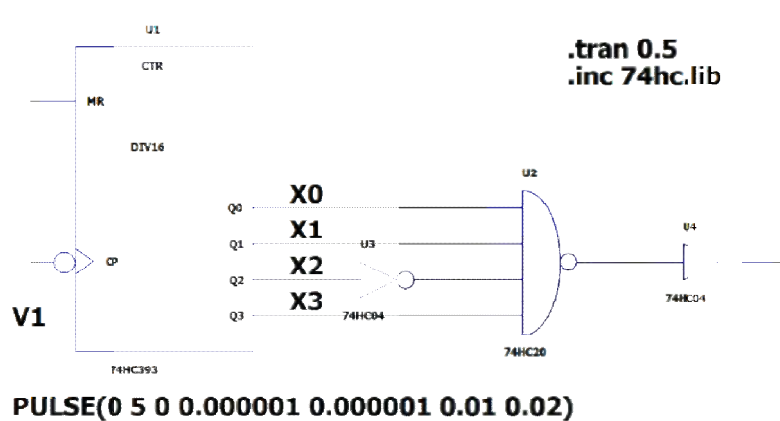


Рис. 56. Схема исследования двоичного счетчика с коэффициентом пересчета 11

На рис. 57 приведен результат моделирования схемы (см. рис. 56).

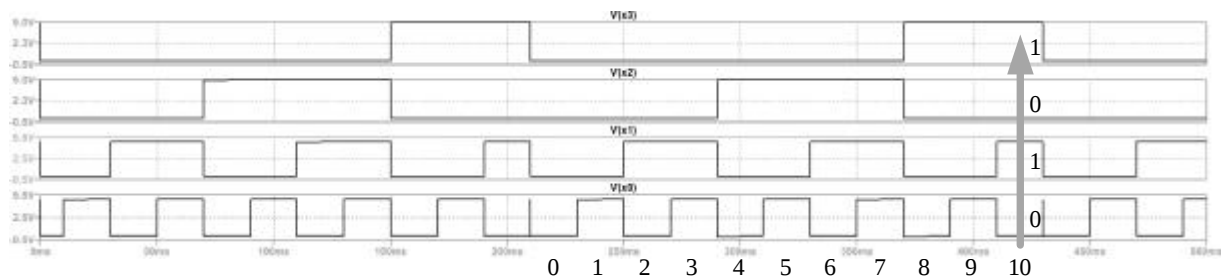


Рис. 57. Результат моделирования работы счетчика с коэффициентом пересчета 11

Содержание работы

1. По рис. 55 набрать схему исследования счётчика – делителя частоты на двухступенчатых триггерах JK и ЛЭ Шеффера из библиотеки элементов программы. Установить параметры импульсного сигнала с частотой 50 Гц и коэффициентом заполнения 50%.
2. Запустить процесс моделирования работы схемы. Получить временные диаграммы работы схемы.
3. Определить коэффициент пересчета для своего варианта.
4. Набрать схему исследования для своего варианта по аналогии с рис. 57. Параметры импульсного сигнала установить аналогично п.1.
5. Запустить процесс моделирования работы схемы. Сохранить временную диаграмму работы счётчика.

Содержание отчёта

Отчёт должен содержать результаты исследования двух схем по следующим пунктам.

1. Схему исследования работы счётчика – делителя частоты по модулю 10 на JK-триггерах.
2. Временную диаграмму работы счётчика по модулю 10.
3. Вычисленное значение коэффициента пересчета для своего варианта.

4. Схему исследования двоичного счётчика на микросхеме 74НС393 с заданным коэффициентом пересчета.

5. Изображение временной диаграммы работы счетчика с заданным коэффициентом пересчета.

6. Выводы по результатам исследований, выполненных в настоящей работе.

ЛАБОРАТОРНАЯ РАБОТА 7. ПРОЕКТИРОВАНИЕ КОДИРОВАННОГО ЦИФРОВОГО АВТОМАТА

Цель работы: изучение принципов оптимального проектирования и построения счётчиков как цифровых автоматов специального назначения в заданной кодировке состояний.

Определение: счётчиком называется цифровой автомат, служащий для формирования многоразрядных двоичных слов (кодов) соответствующих количеству входных импульсов. По мере поступления входных импульсов счётчик последовательно перебирает все коды своих состояний в определённом для данной схемы порядке.

Степень сложности и стоимость производства цифровых автоматов зависят от выбранной элементной базы и использованных при их проектировании методов минимизации принципиальной схемы. Для цифровых автоматов основным принципом оптимизации является перебор и сопоставление различных вариантов схем построения этих цифровых автоматов по критериям сложности и стоимости производства.

Варианты кодирования состояний цифровых автоматов как счётчиков по модулю 10, приведены в табл. 21.

Таблица 21

Варианты кодов состояний для выполнения работы

№ варианта	Код состояния СТ10
1	4821
2	2421
3	5121
4	53-21
5	75-31
6	5421
7	642-1
8	462-1
9	4821
10	4221
11	5211
12	632-1
13	65-31
14	4521

№ варианта	Код состояния СТ10
15	1234
16	4321
17	2-146
18	1215
19	13-15
20	531-1
21	2124
22	1248
23	1245
24	4512
25	425-1
26	4321
27	1234
28	1432
29	432-1
30	8124

В обозначениях кодов 8421, 63-21 и т. д. указан десятичный вес p_i двоичной единицы X_i соответствующего разряда. Для построения цифрового автомата строится таблица (см. таблицу 23), в левой части которой перебираются все состояния от 0_8 (0000) до 17_8 (1111). Каждое состояние соответствует некоторой десятичной цифре в весах входного кода 63-21. Например, входной код 1110 соответствует десятичной цифре

$$1110 \rightarrow 1 \cdot 6 + 1 \cdot 3 + (-2) \cdot 1 + 0 \cdot 1 = 7$$

В правой части таблицы необходимо последовательно перебрать все цифры от 0 до 9 в указанной кодировке.

Например, из цифры 0 мы должны перейти в цифру 1. Поэтому напротив цифры 0 в правой части формируем выходы, соответствующие коду цифры 1. Напротив цифры 1 заполняем выходы кодом цифры 2 и т.д.

Как пример рассмотрим оптимальное проектирование счётчика по модулю 10 в коде 63-21. В табл. 22 приведена кодировка списка состояний цифрового автомата.

Недостижимыми являются состояния 2, 3, 6, 12, 15, 16. Поскольку общее количество состояний 16, а автомат реализует 10 цифр от 0 до 9, то должно остаться 6 недостижимых (неопределённых) состояний.

Автомат реализует последовательный перебор цифр от 0 до 9 в заданной кодировке.

Кодировка состояний в коде 63-21

Сост.	X3 (6)	X2 (3)	X1 (-2)	X0 (1)	Цифра	Y3	Y2	Y1	Y0
0	0	0	0	0	0	0	0	0	1
1	0	0	0	1	1	0	1	1	1
2	0	0	1	0	-2	*	*	*	*
3	0	0	1	1	-1	*	*	*	*
4	0	1	0	0	3	0	1	0	1
5	0	1	0	1	4	1	0	1	1
6	0	1	1	0	1	*	*	*	*
7	0	1	1	1	2	0	1	0	0
10	1	0	0	0	6	1	0	0	1
11	1	0	0	1	7	1	1	1	1
12	1	0	1	0	4	*	*	*	*
13	1	0	1	1	5	1	0	0	0
14	1	1	0	0	9	0	0	0	0
15	1	1	0	1	10	*	*	*	*
16	1	1	1	0	7	*	*	*	*
17	1	1	1	1	8	1	1	0	0

Таблицу возбуждений для D-триггеров блока памяти цифрового автомата строить не нужно, так как она совпадает с кодированной таблицей переходов. Для построения схемы исследования цифрового автомата на D-триггерах, произведем минимизацию функций выходов.

$$Y3 = 5 \cup 10 \cup 11 \cup 13 \cup 17 \cup (2) \cup (3) \cup (6) \cup (12) \cup (15) \cup (16)$$

$$Y2 = 1 \cup 4 \cup 7 \cup 11 \cup 17 \cup (2) \cup (3) \cup (6) \cup (12) \cup (15) \cup (16)$$

$$Y1 = 1 \cup 5 \cup 11 \cup (2) \cup (3) \cup (6) \cup (12) \cup (15) \cup (16)$$

$$Y0 = 0 \cup 1 \cup 4 \cup 5 \cup 10 \cup 11 \cup (2) \cup (3) \cup (6) \cup (12) \cup (15) \cup (16)$$

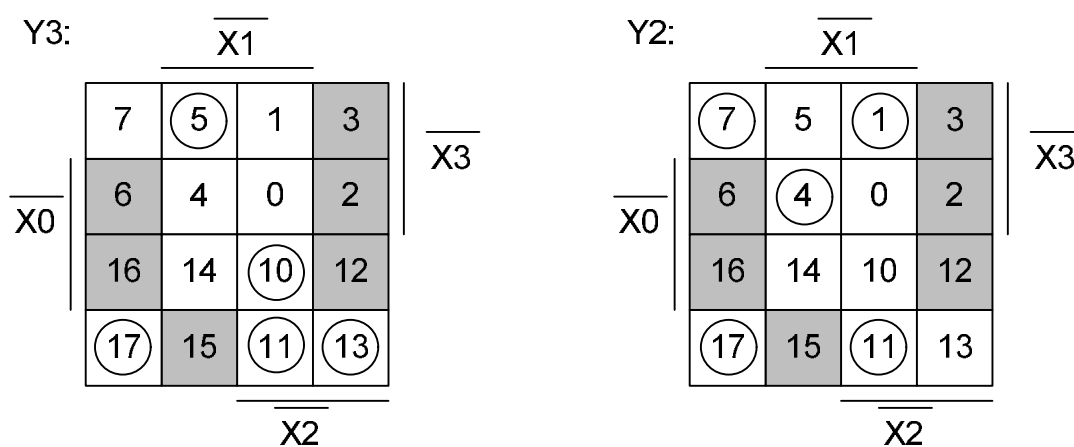


Рис. 58. Минимизация Y3 и Y2

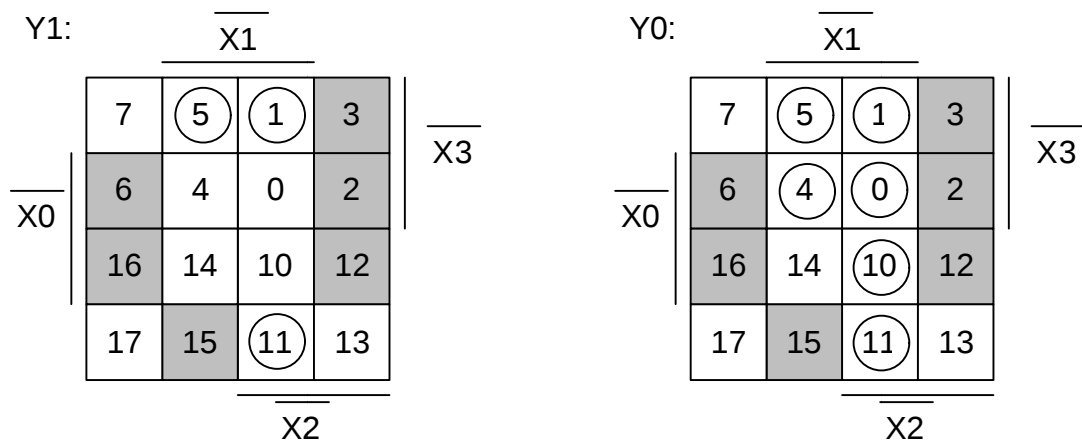


Рис. 59. Минимизация Y1 и Y0

$$Y3 = X3 \overline{X2} \cup X3 X0 \cup X2 \overline{X1} X0$$

$$Y2 = X2 X1 \cup X3 \overline{X1} X0 \cup \overline{X3} \overline{X2} X0 \cup \overline{X3} X2 \overline{X0}$$

$$Y1 = \overline{X1} X0$$

$$Y0 = \overline{X2} \overline{X1} \cup \overline{X3} \overline{X1}$$

Схема исследования цифрового автомата на D-триггерах представлена на рис. 60.

На рис. 61 приведен результат моделирования работы схемы. На графиках выносим сигналы X0, X1, X2, X3.

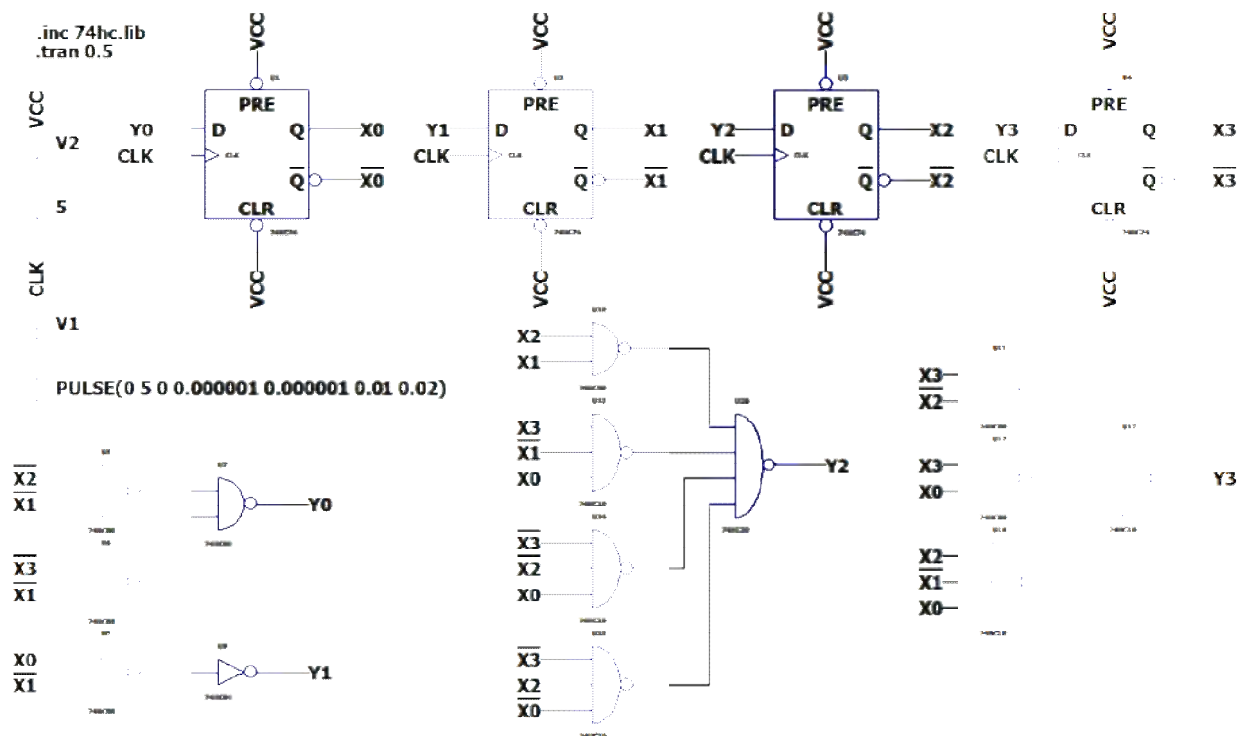


Рис. 60. Схема исследования цифрового автомата на D-триггерах

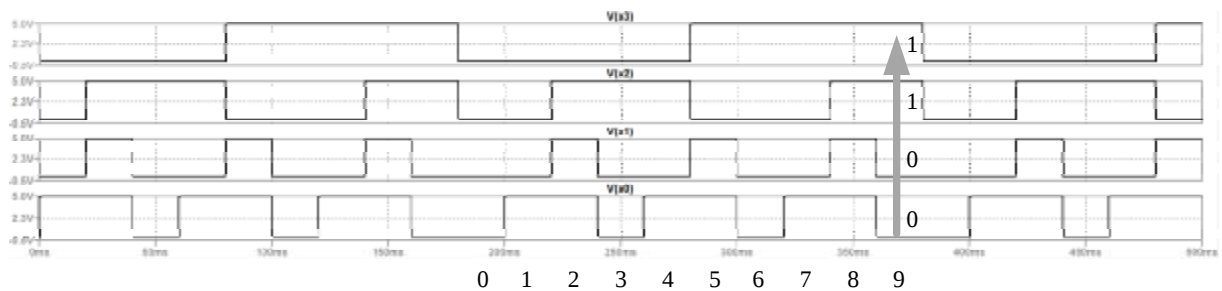


Рис. 61. Временные диаграммы цифрового автомата на D-триггерах

Для построения кодированной таблицы возбуждения JK триггеров используется матрица переходов, приведенная в табл. 19.

Функции возбуждения JK триггеров блока памяти второго варианта схемы построения цифрового автомата минимизированы с использованием диаграмм Вейча и записаны с использованием базиса И-НЕ.

$$J3 = 5 \cup (2) \cup (3) \cup (6) \cup (10) \cup (11) \cup (12) \cup (13) \cup (14) \cup (15) \cup (16) \cup (17)$$

$$J2 = 1 \cup 11 \cup (2) \cup (3) \cup (4) \cup (5) \cup (6) \cup (7) \cup (12) \cup (14) \cup (15) \cup (16) \cup (17)$$

$$J1 = 1 \cup 5 \cup 11 \cup (2) \cup (3) \cup (6) \cup (7) \cup (12) \cup (13) \cup (15) \cup (16) \cup (17)$$

$$J0 = 0 \cup 4 \cup 10 \cup (1) \cup (2) \cup (3) \cup (5) \cup (6) \cup (7) \cup (11) \cup (12) \cup (13) \cup (15) \cup (16) \cup (17)$$

$$K3 = 14 \cup (0) \cup (1) \cup (2) \cup (3) \cup (4) \cup (5) \cup (6) \cup (7) \cup (12) \cup (15) \cup (16)$$

$$K2 = 5 \cup 14 \cup (0) \cup (1) \cup (2) \cup (3) \cup (6) \cup (10) \cup (11) \cup (12) \cup (13) \cup (15) \cup (16)$$

$$K1 = 7 \cup 13 \cup 17 \cup (0) \cup (1) \cup (2) \cup (3) \cup (4) \cup (5) \cup (6) \cup (10) \cup (11) \cup (12) \cup (14) \cup (15) \cup (16)$$

$$K0 = 7 \cup 13 \cup 17 \cup (0) \cup (2) \cup (3) \cup (4) \cup (6) \cup (10) \cup (12) \cup (14) \cup (15) \cup (16)$$

Таблица 23

Таблица переходов JK-триггеров

Сост.	X3 (6)	X2 (3)	X1 (-2)	X0 (1)	Цифра	Y3	Y2	Y1	Y0	J3	J2	J1	J0	K3	K2	K1	K0
0	0	0	0	0	0	0	0	0	1	0	0	0	1	*	*	*	*
1	0	0	0	1	1	0	1	1	1	0	1	1	*	*	*	*	0
2	0	0	1	0	-2	*	*	*	*	*	*	*	*	*	*	*	*
3	0	0	1	1	-1	*	*	*	*	*	*	*	*	*	*	*	*
4	0	1	0	0	3	0	1	0	1	0	*	0	1	*	0	*	*
5	0	1	0	1	4	1	0	1	1	1	*	1	*	*	1	*	0
6	0	1	1	0	1	*	*	*	*	*	*	*	*	*	*	*	*
7	0	1	1	1	2	0	1	0	0	0	*	*	*	*	0	1	1
10	1	0	0	0	6	1	0	0	1	*	0	0	1	0	*	*	*
11	1	0	0	1	7	1	1	1	1	*	1	1	*	0	*	*	0
12	1	0	1	0	4	*	*	*	*	*	*	*	*	*	*	*	*
13	1	0	1	1	5	1	0	0	0	*	0	*	*	0	*	1	1
14	1	1	0	0	9	0	0	0	0	*	*	0	0	1	1	*	*
15	1	1	0	1	10	*	*	*	*	*	*	*	*	*	*	*	*
16	1	1	1	0	7	*	*	*	*	*	*	*	*	*	*	*	*
17	1	1	1	1	8	1	1	0	0	*	*	*	*	0	0	1	1

После минимизации этих ФАЛ по диаграммам Вейча получим:

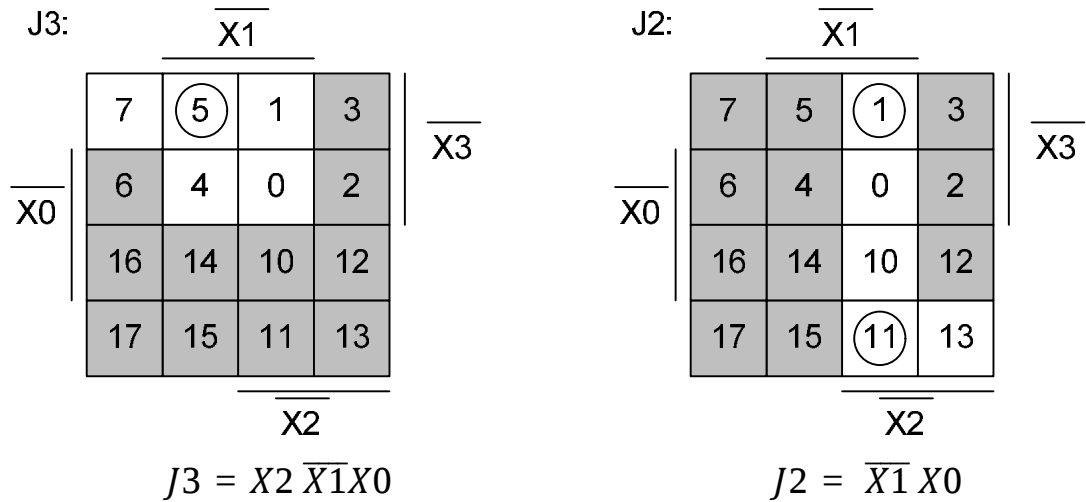


Рис. 62. Минимизация J3, J2

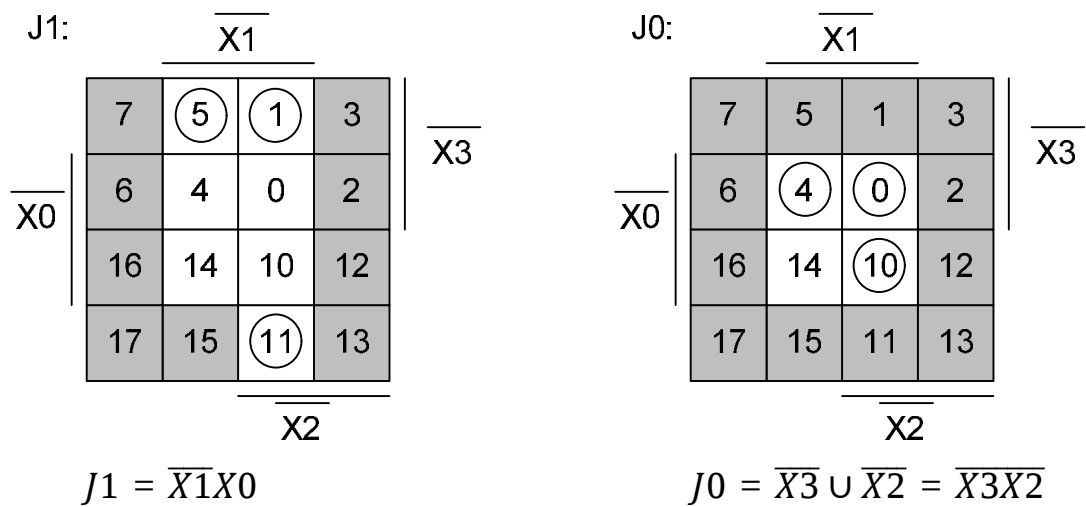


Рис. 63. Минимизация J1, J0

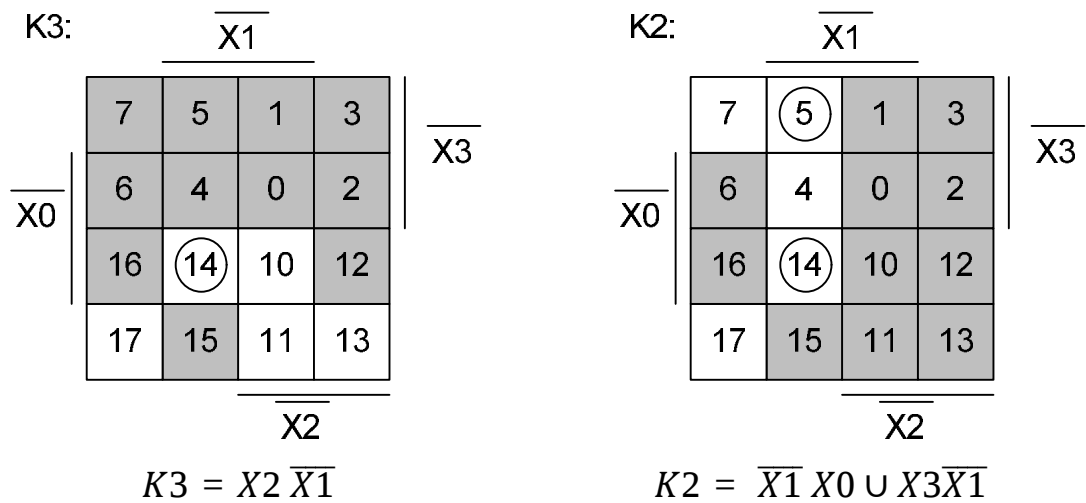


Рис. 64. Минимизация K3, K2

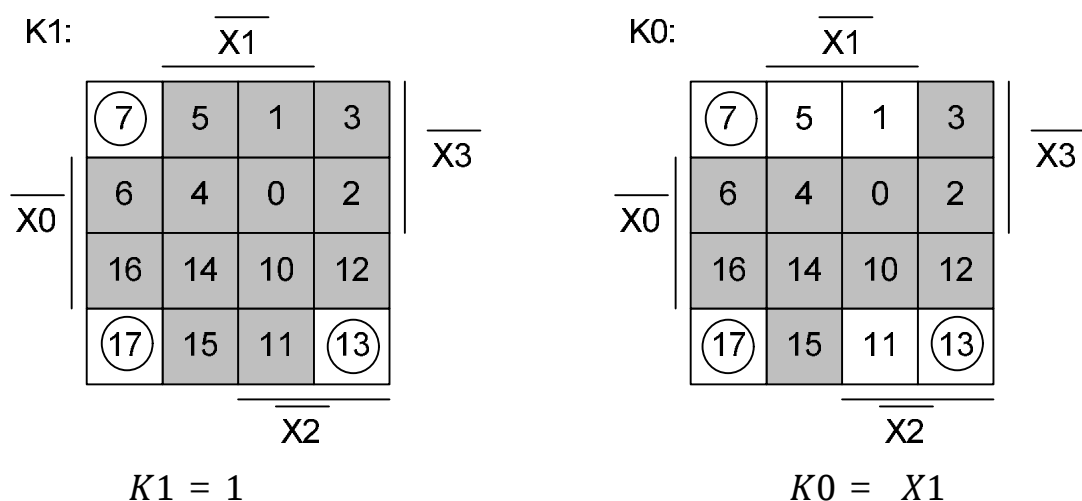


Рис. 65. Минимизация K1, K0

Комбинационная схема кодированного цифрового автомата с использованием JK триггеров представлена на рис. 66.

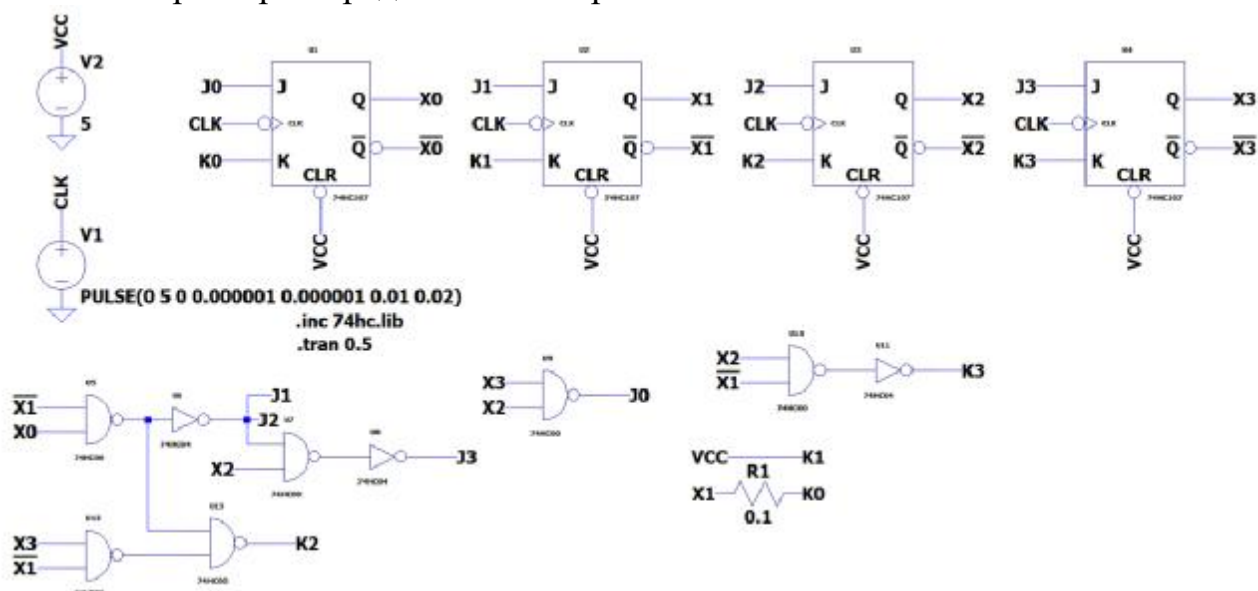


Рис. 66. Схема реализации цифрового автомата на JK-триггерах

На рис. 67 приведен результат моделирования схемы цифрового автомата, собранного на JK-триггерах. На графики выводятся сигналы X0, X1, X2, X3.

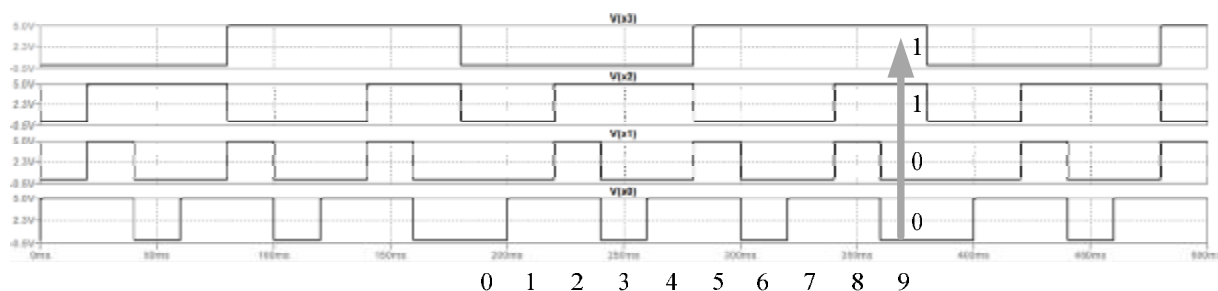


Рис. 67. Временные диаграммы цифрового автомата на JK-триггерах

Содержание работы

1. Набрать схему исследования кодированного цифрового автомата по модулю 10 в заданной кодировке (табл. 22) с использованием D-триггеров.
2. Установить параметры импульсного сигнала с частотой 50 Гц.
3. Запустить процесс моделирования работы схемы. Время моделирования 0,5 с.
4. Получить временные диаграммы работы схемы.
5. Набрать схему исследования кодированного цифрового автомата по модулю 10 в заданной кодировке (табл. 22) с использованием JK-триггеров.
6. Запустить процесс моделирования работы схемы. Время моделирования 0,5 с.
7. Получить временные диаграммы работы схемы.

Содержание отчёта

Отчёт должен содержать результаты исследования выбранной схемы счётчика по следующим пунктам.

1. Таблицу переходов кодированного цифрового автомата для заданного варианта.
2. Минимизацию функций выходов цифрового автомата для реализации на D-триггерах.
3. Схему исследования работы кодированного цифрового автомата для заданного варианта на базе D-триггеров.
4. Временные диаграммы моделирования работы схемы кодированного цифрового автомата на базе D-триггеров.
5. Таблицу переходов кодированного цифрового автомата для формирования функций возбуждения JK-триггеров.
6. Минимизацию функций возбуждения JK-триггеров.
7. Схему исследования работы кодированного цифрового автомата для заданного варианта на базе JK-триггеров.
8. Временные диаграммы моделирования работы схемы.
9. Выводы по результатам выполнения работы.

ЛАБОРАТОРНАЯ РАБОТА 8. АРИФМЕТИКО-ЛОГИЧЕСКОЕ УСТРОЙСТВО

Цель работы: изучение способов управления и принципов построения арифметико-логических устройств как основных комбинационных узлов для построения процессоров вычислительных машин.

Определение: арифметико-логическим устройством называется схема, служащая для определения результатов выполнения математических или логических операций над одним или двумя двоичными кодами – операндами. АЛУ может быть как чисто комбинационной схемой, так и содержать в

своём составе запоминающие устройства. Если схема является комбинационной, то в её составе содержатся отдельные узлы, выполняющие все математические и логические операции, доступные АЛУ. Тогда для выбора результата, соответствующего коду операции, результаты которой требуются в текущий момент времени, выход нужного узла коммутируется на выход АЛУ.

Таблица 24

Логические функции АЛУ		
S	Логические функции M=1	Арифметико-логические функции M=0
0000	$\bar{A}$	$\bar{A} + C$
0001	$\overline{A \cup B}$	$\overline{A \cup B} + C$
0010	$\bar{A} \cap B$	$\bar{A} \cap B + C$
0011	0	C
0100	$\overline{A \cap B}$	$\overline{A \cap B} + C$
0101	$\bar{B}$	$\bar{B} + C$
0110	$A \oplus B$	$A \oplus B + C$
0111	$A \cap \bar{B}$	$A \cap \bar{B} + C$
1000	$\bar{A} \cup B$	$\bar{A} \cup B + C$
1001	$\overline{A \oplus B}$	$\overline{A \oplus B} + C$
1010	B	$B + C$
1011	$A \cap B$	$A \cap B + C$
1100	1	$1 + C$
1101	$A \cup \bar{B}$	$A \cup \bar{B} + C$
1110	$A \cup B$	$A \cup B + C$
1111	A	$A + C$

Ячейка АЛУ, предназначенная для выполнения 16 арифметических и 16 поразрядных логических операций с двумя 1-разрядными операндами, представлена на рис. 68.

Схема имеет информационные входы A, B, управляющие входы S3–S0 и M, входной перенос из предыдущего разряда C0. Код исполняемой операции подаётся на входы M, S3–S0. На выходе F формируется результат операции.

Состояние входа M определяет выполнение арифметических или логических операций. При выполнении математических операций на вход M должен быть подан уровень логического нуля. С подачей на вход M логической единицы, запрещается выполнение переносов, и на выходах АЛУ появляются результаты логических операций.

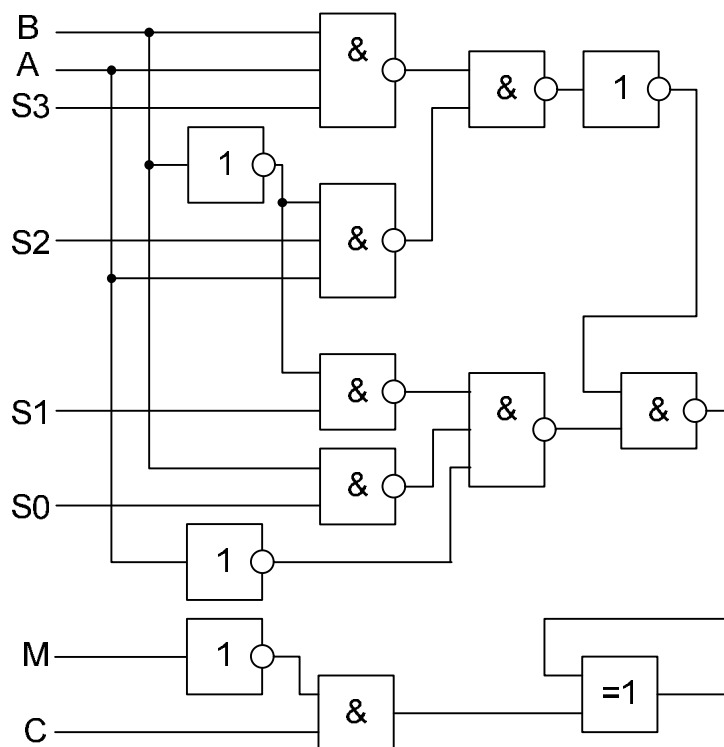


Рис. 68. Ячейка АЛУ

Описание схемы исследования

На рис. 69 приведена схема исследования четырех 1-разрядных ячеек АЛУ. Входные сигналы А и В для каждой ячейки подключены в лексикографической последовательности согласно табл. 25.

Таблица 25

Состояние входов для каждой ячейки АЛУ

Ячейка	А	В
0	0	0
1	0	1
2	1	0
3	1	1

Управляющий вход М опущен для упрощения схемы, в результате чего схема выполняет только логические операции.

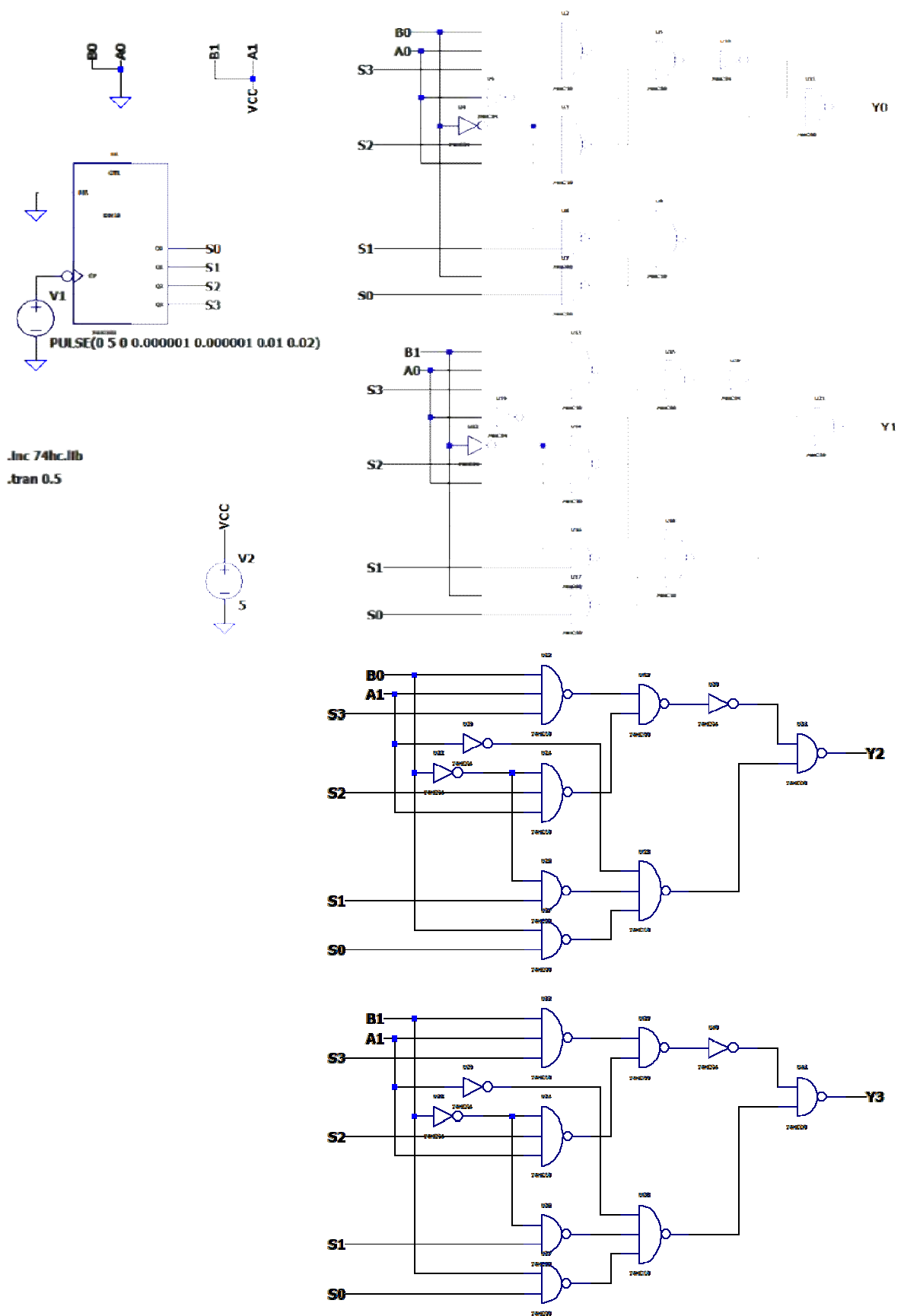


Рис. 69. Схема исследования АЛУ

На рис. 70. приведен результат моделирования схемы, представленной на рис. 69.

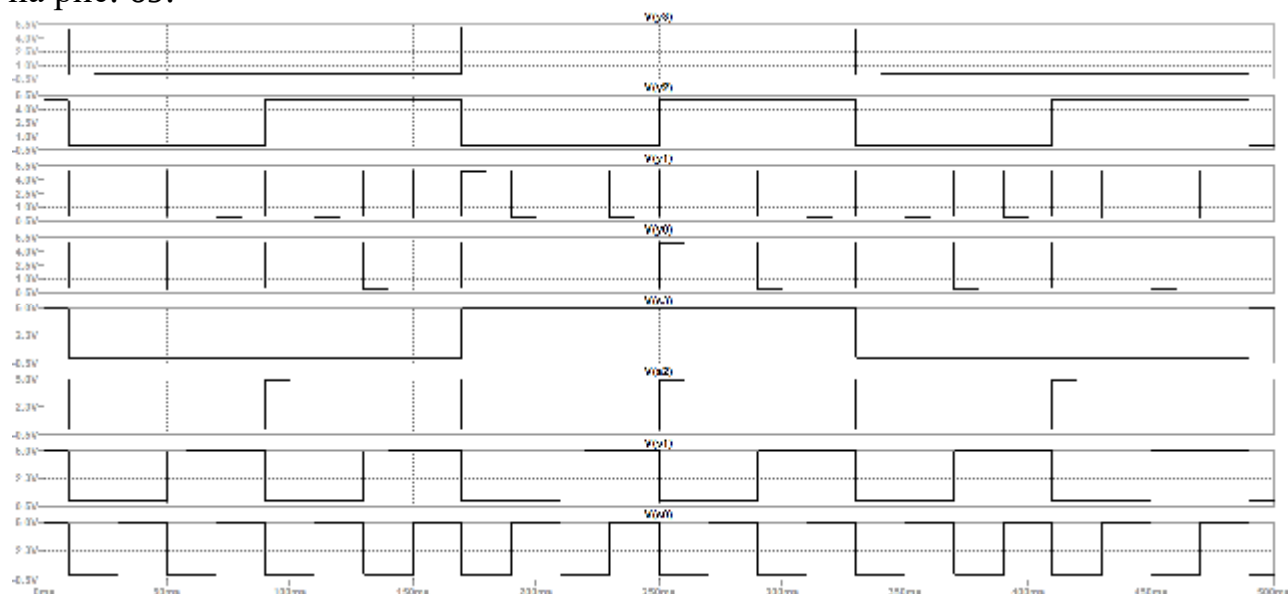


Рис. 70. Результат моделирования схемы АЛУ

Содержание работы

1. Набрать схему исследования АЛУ (см. рис. 69). Входные слова (операнды) задать согласно табл. 25.

2. Запустить процесс моделирования работы схемы. Проанализировать сигналы S0...S3 и F0...F3.

3. Сравнить экспериментально полученные результаты с теоретическими, вычисленными по табл. 24.

4. Заполнить табл. 26 для всех состояний управляющих сигналов S3...S0. Убедиться, что полученные функции соответствуют табл.24 для логических операций.

Таблица 26

Анализ моделирования ячейки АЛУ

S3 S2 S1 S0	A	B	Y	Функция
0 0 0 0	0	0	Y0 = 1	$\bar{A}$
	0	1	Y1 = 1	
	1	0	Y2 = 0	
	1	1	Y3 = 0	
0 0 0 1	0	0	...	...
	0	1		
	1	0		
	1	1		

Содержание отчёта

Отчёт должен содержать результаты исследования ячейки АЛУ по следующим пунктам.

1. Схема исследования ячейки АЛУ.
2. Результат моделирования работы АЛУ при всех возможных наборах входных сигналов А, В.
3. Таблица выходов ячеек при различных наборах входных сигналов А и В для всех наборов управляющих сигналов S3...S0.
4. Выводы по результатам выполнения работы.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Вставская, Е.В. Микропроцессорные средства систем управления: конспект лекций / Е.В. Вставская, В.И. Константинов – Челябинск: Издательский центр ЮУрГУ, 2010. – 91 с.
2. Вставская, Е.В. Вычислительные машины, системы и сети Ч. 1: Элементарные операционные узлы ЭВМ: конспект лекций / Е. В. Вставская, В. И. Константинов – Челябинск: Издательский центр ЮУрГУ, 2010. – 77 с.
3. Гудилин, А. Е. Архитектура ЭВМ, узлы и подсистемы: учебное пособие / А. Е. Гудилин – Челябинск: Издательский центр ЮУрГУ, 2004. – 85 с.
4. Угрюмов, Е. П. Цифровая схемотехника: учеб. пособие для вузов по направлению 230100 "Информатика и вычисл. техника" / Е. П. Угрюмов. - 3-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2010. – 797 с.
5. Зубчук, В. И. Справочник по цифровой схемотехнике / В. И. Зубчук, В. П. Сигорский, А. Н. Шкуро. – Киев: Тэхника, 1990. – 446 с.
6. http://ltwiki.org/?title=Components_Library_and_Circuits