

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**
(БГТУ им. В. Г. Шухова)

Методические указания к лабораторным работам по дисциплине
Схемотехника ЭВМ

специальность:

10.05.03 Информационная безопасность автоматизированных систем

специализация:

10.05.03-07 Обеспечение информационной безопасности распределённых
информационных систем

Квалификация

Специалист по защите информации

Форма обучения

очная

Институт энергетики, информационных технологий и управляющих систем
Кафедра программного обеспечения вычислительной техники и
автоматизированных систем

Белгород – 2017

СОДЕРЖАНИЕ

| | |
|-----------------------------------|----|
| Лабораторная работа 1 (Lr1) | 3 |
| Лабораторная работа 2 (Lr2) | 14 |
| Лабораторная работа 3 (Lr3) | 26 |
| Лабораторная работа 4 (Lr4) | 33 |
| Лабораторная работа 5 (Lr5) | 43 |
| Лабораторная работа 6 (Lr6) | 52 |
| Лабораторная работа 7 (Lr7) | 63 |
| Лабораторная работа 8 (Lr8) | 71 |

Лабораторная работа 1 (Lr1)

ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ И СХЕМЫ

ЦЕЛЬ РАБОТЫ

Ознакомление с основными характеристиками логических элементов и основами синтеза логических схем.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И РАСЧЕТНЫЕ ФОРМУЛЫ

1. ОПРЕДЕЛЕНИЯ КОМБИНАЦИОННЫХ И ПОСЛЕДОВАТЕЛЬНОСТНЫХ УСТРОЙСТВ

Устройства, реализующие функции алгебры логики, называют *логическими* или *цифровыми* и классифицируют по различным отличительным признакам. Так, по характеру информации на входах и выходах логические устройства подразделяют на устройства последовательного, параллельного и смешанного действия, а по схемному решению и характеру связи между входными и выходными переменными с учётом их изменения по тактам работы – на комбинационные и последовательностные.

В *комбинационных* устройствах значения (0 или 1) сигналов на выходах в каждый конкретный момент времени полностью определяются значениями (комбинацией, набором) действующих в данный момент цифровых входных сигналов. В *последовательностных* же устройствах значения выходных сигналов в n -такте определяются не только значениями входных сигналов в этом такте, но и зависят от внутренних состояний устройств, которые произошли в результате воздействия входных сигналов в предшествующие такты.

Данная работа посвящена изучению простейших комбинационных логических устройств, реализующих логические функции сложения, умножения и отрицания.

2. ОСНОВНЫЕ ЭЛЕМЕНТЫ АЛГЕБРЫ ЛОГИКИ

Анализ комбинационных устройств удобно проводить с помощью алгебры логики, оперирующей только с двумя понятиями: истинным (логическая 1) и ложным (логический 0). В результате, функции, отображающие информацию, принимают в каждый момент времени только значения 0 или 1. Такие функции называют *логическими*, а сигналы (входные и выходные переменные) – *двоичными* (бинарными).

Схемные элементы, при помощи которых осуществляется преобразование поступающих на их входы двоичных сигналов и непосредственное выполнение предусмотренных логических операций, называют *логическими* устройствами.

В общем случае логическое устройство может иметь n входов и m выходов. Рассматривая входные сигналы x_1, x_2, \dots, x_n в качестве аргументов, можно соответствующие выходные сигналы представлять в виде функции $y_i = f(x_0, x_1, x_2, \dots, x_n)$ с помощью операций алгебры логики.

Функции алгебры логики (ФАЛ), иногда называемые *переключаемыми функциями*, обычно представляют в алгебраической форме (в виде математического выражения), например $y_i = (x_0 \wedge x_1) \vee (x_1 \wedge x_2)$, или в виде таблиц истинности (комбинационных таблиц).

Таблица истинности содержит всевозможные комбинации (наборы) бинарных значений входных переменных с соответствующими им бинарными значениями выходных переменных; каждому набору входных сигналов соответствует определенное значение выходного сигнала – значение логической функции y_i . Максимальное число возможных различных наборов (строк) зависит от числа входных переменных n и равно 2^n .

В булевой алгебре выделяют три основные функции: конъюнкция, дизъюнкция, отрицание. Остальные функции являются производными от приведенных выше.

Основные логические операции состоят из следующих элементарных преобразований двоичных сигналов:

- *логическое сложение* или *дизъюнкция*, обозначаемое символом " \vee " (или "+") и называемое также операцией ИЛИ. При этом число аргументов (слагаемых x) может быть любым. Эта операция для функции двух переменных x_1 и x_2 описывается в виде логической формулы

$$y = x_1 \vee x_2 = x_1 + x_2.$$

Это значит, что y истинно (равно 1), если истинно хотя бы одно из слагаемых x_1 или x_2 . И только в случае, когда все слагаемые x равны 0, результат логического сложения y также равен 0.

Условное обозначение, таблица истинности и другие показатели этой логической функции приведены во втором столбце табл. 1.1;

- *логическое умножение* или *конъюнкция*, обозначаемое символом " \wedge " (или ".") и называемое также операцией И. При этом число аргументов (сомножителей x) может быть любым. Эта операция для функции двух переменных x_1 и x_2 описывается в виде логической формулы

$$y = x_1 \wedge x_2 = x_1 \cdot x_2 = x_1 x_2.$$

Это значит, что y истинно (равно 1), если истинны сомножители x_1 и x_2 . В случае, если хотя бы один из сомножителей равен 0, результат логического умножения y равен 0.

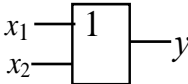
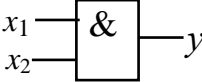
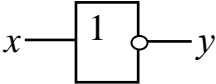
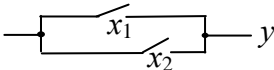
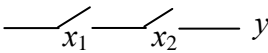
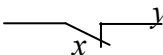
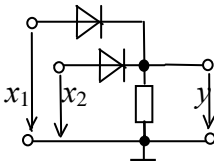
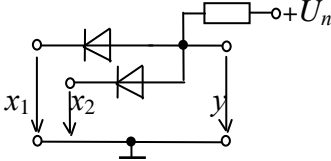
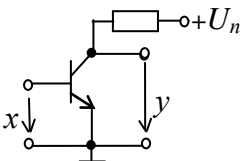
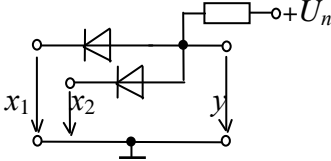
Условное обозначение, таблица истинности и другие показатели логической функции И приведены в третьем столбце табл. 1.1;

- *логическое отрицание* или *инверсия*, обозначаемое чёрточкой над переменной и называемое операцией НЕ. Эта операция записывается в виде

$$y = \bar{x}.$$

Это значит, что y истинно (равно 1), если x ложно (равно 0), и наоборот. Очевидно, что операция y выполняется над одной переменной x и её значение всегда противоположно этой переменной (см. четвертый столбец табл. 1.1).

Т а б л и ц а 1.1

| Формы отображения основных логических функций | | | | | | | |
|---|--|--|-------|---|---|---|-----|
| Наименование функции → | | Дизъюнкция | | Конъюнкция | | Инверсия | |
| Символическая я | | ∨ или + | | ∧ или • | | \bar{x} | |
| Буквенная | | ИЛИ | | И | | НЕ | |
| Условная графическая | |  | |  | |  | |
| Аналитическая | | $y = x_1 \vee x_2 = x_1 + x_2$ | | $y = x_1 \wedge x_2 = x_1 x_2$ | | $y = \bar{x}$ | |
| Табличная (истинности) | | x_1 | x_2 | y | x_1 | x_2 | y |
| | | 0 | 0 | 0 | 0 | 0 | 0 |
| | | 0 | 1 | 1 | 0 | 1 | 0 |
| | | 1 | 0 | 1 | 1 | 0 | 0 |
| | | 1 | 1 | 1 | 1 | 1 | 1 |
| Контактная | |  | | |  | | |
| | |  | | | | | |
| | |  | | |  | | |
| | |  | | | | | |
| | |  | | | | | |

Основные логические операции ИЛИ, И и НЕ позволяют аналитически описать, а логические элементы ИЛИ (*дизъюнктор*), И (*конъюнктор*) и НЕ (*инвертор*) – реализовать комбинационное устройство любой степени сложности, т. е. операции $y = x_1 + x_2$, $y = x_1 x_2$ и $y = \bar{x}$ обладают функциональной полнотой и составляет функционально полный набор.

В качестве примера рассмотрим функцию неравнозначности y двух переменных x_1 и x_2 , принимающая значение 1 при $x_1 \neq x_2$ и значение 0 при $x_1 = x_2 = 0$ или при $x_1 = x_2 = 1$, т. е. $y = \bar{x}_1 x_2 + x_1 \bar{x}_2$.

Операцию неравнозначности чаще называют *суммированием по модулю 2* и обозначают $y = x_1 \oplus x_2$.

Примеры контактной и простейшей схемной реализаций дизъюнктора, конъюнктора и инвертора приведены в предпоследней и последней строках табл. 1.1.

3. БАЗОВЫЕ ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ

Особое значение в цифровой электронике имеют универсальные (базовые) логические элементы, способные образовать функционально полный набор, с помощью которых можно реализовать синтез устройств любой сложности. При интегральной технологии удобство изготовления одного базового элемента

имеет решающее значение. Поэтому базовые логические устройства составляют основу большинства цифровых ИМС.

К универсальным логическим операциям (устройствам) относят две разновидности базовых элементов:

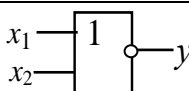
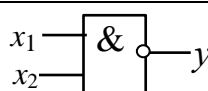
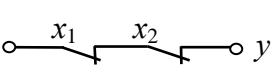
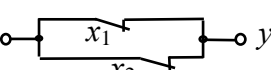
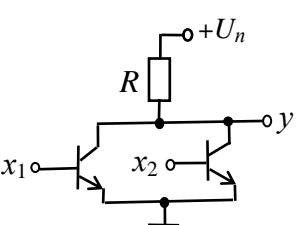
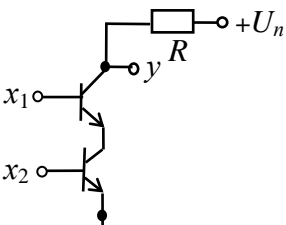
- *функцию Пирса*, обозначаемую символически вертикальной стрелкой \downarrow (стрелка Пирса) и отображающую операцию ИЛИ-НЕ. Для простейшей функции двух переменных x_1 и x_2 функция $y = 1$ тогда и только тогда, когда $x_1 = x_2 = 0$:

$$y = x_1 \downarrow x_2 = \overline{x_1 + x_2};$$

- *функцию Шеффера*, обозначаемую символически вертикальной черточкой $|$ (штрих Шеффера) и отображающую операцию И-НЕ. Для простейшей функции двух переменных x_1 и x_2 функция $y = 0$ тогда и только тогда, когда $x_1 = x_2 = 1$:

$$y = x_1 | x_2 = \overline{x_1 x_2}.$$

Т а б л и ц а 1.2

| Формы отображения базовых логических функций | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|---|-----|--|-------|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|-------|-------|-----|---|---|---|---|---|---|---|---|---|---|---|---|
| Наименование функции → | Функция Пирса | | Функция Шеффера | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Символическая | ↓ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Буквенная | ИЛИ-НЕ | | И-НЕ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Условная графическая |  | |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Аналитическая | $y = x_1 \downarrow x_2$ | | $y = x_1 x_2$ | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Табличная (истинности) | <table><tr><th>x_1</th><th>x_2</th><th>y</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | | x_1 | x_2 | y | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | <table><tr><th>x_1</th><th>x_2</th><th>y</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table> | x_1 | x_2 | y | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| x_1 | x_2 | y | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| x_1 | x_2 | y | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Контактная |  | |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Схемо-техническая |  | |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

При одних и тех же значениях аргументов обе функции отображают операцию инверсии. Важнейшие показатели функций Шеффера и Пирса представлены в табл. 1.2.

В последней строке табл. 1.2 приведены примеры построения двухвходовой схемы ИЛИ-НЕ, в которой к нагрузочному резистору R подключены коллекторы двух параллельно включенных биполярных транзисторов $p-n-p$ -типа, эмиттеры которых заземлены, и схемы И-НЕ, в которой последовательно включены два биполярных транзистора $p-n-p$ -типа (эмиттер нижнего транзистора подключен к земле) и нагрузочный резистор R .

4. ПРЕДСТАВЛЕНИЕ ЛОГИЧЕСКИХ ФУНКЦИЙ МАТЕМАТИЧЕСКИМИ ВЫРАЖЕНИЯМИ

Наиболее распространенным способом задания логических функций является табличная форма. Таблицы истинности позволяют полно и однозначно установить все существующие логические связи.

При табличном представлении логических функций их записывают в одной из канонических форм: совершенной дизъюнктивной нормальной форме (СДНФ) или совершенной конъюнктивной нормальной форме (СКНФ).

Математическое выражение логической функции в СДНФ получают из таблицы истинности следующим образом: для каждого набора аргументов, на котором функция равна 1, записывают элементарные произведения переменных, причем переменные, значения которых равны нулю, записывают с инверсией. Полученные произведения, называемые *конституентами единицы* или *минтермами*, суммируют.

Запишем логическую функцию y трех переменных a , b и c , представленной в виде табл. 1.3, в СДНФ:

Т а б л и ц а 1.3

| № | a | b | c | y |
|---|-----|-----|-----|-----|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 2 | 0 | 1 | 0 | 0 |
| 3 | 0 | 1 | 1 | 1 |
| 4 | 1 | 0 | 0 | 0 |
| 5 | 1 | 0 | 1 | 1 |
| 6 | 1 | 1 | 0 | 1 |
| 7 | 1 | 1 | 1 | 1 |

$$y(a, b, c) = \bar{a}bc + a\bar{b}c + abc\bar{c} + abc.$$

Совершенной конъюнктивной нормальной формой называют логическое произведение элементарных сумм, в каждую из которых аргумент или его отрицание входят один раз.

При этом для каждого набора аргументов таблицы истинности, на котором функция y равна 0, составляют элементарную сумму, причем переменные, значения которых равно 1, записывают с отрицанием. Полученные суммы, называемые

конституентами нуля или *макстермами*, объединяют операцией логического умножения.

Для функции (табл. 1.3) СКНФ

$$y(a, b, c) = (a + b + c)(a + b + \bar{c})(a + \bar{b} + c)(\bar{a} + b + c).$$

5. ПЕРЕХОД ОТ ЛОГИЧЕСКОЙ ФУНКЦИИ К ЛОГИЧЕСКОЙ СХЕМЕ

Для построения логической схемы необходимо логические элементы, предназначенные для выполнения логических операций, располагать, начиная от входа, в порядке, указанном в булевом выражении.

Построим структуру логического устройства, реализующего логическую функцию трех переменных

$$y = (a + b + c)(a + b + \bar{c})(\bar{a} + b + c)(\bar{a} + \bar{b} + c).$$

Слева располагаем входы a , b и c с ответвлениями на три инвертора, затем четыре элемента ИЛИ и, наконец, элемент И на выходе (рис. 1.1).

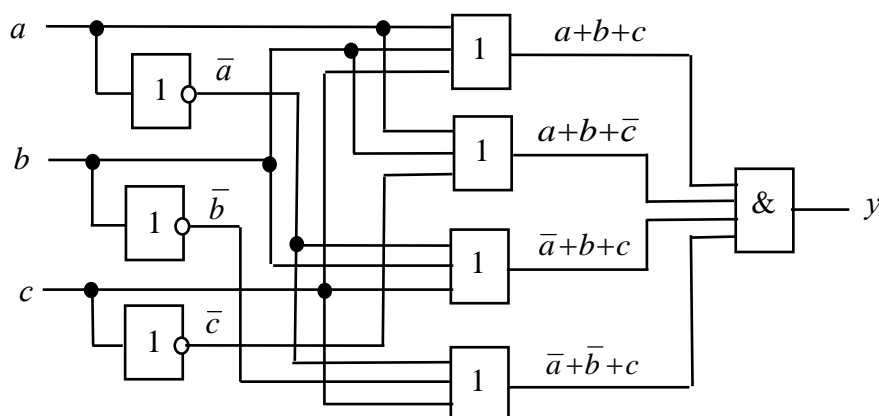


Рис. 1.1

Итак, любую логическую функцию можно реализовать непосредственно по выражениям, представленным в виде СДНФ или СКНФ. Однако, полученная таким образом схема, как правило, не оптимальна с точки зрения её практической реализации: она громоздка, содержит много логических элементов и возникают трудности в обеспечении её высокой надёжности.

Алгебра логики позволяет преобразовать формулы, описывающие сложные высказывания с целью их упрощения [10]. Это помогает в конечном итоге определить оптимальную структуру того или иного логического устройства, реализующего любую сложную функцию. Под оптимальной структурой принято понимать такое построение логического устройства, при котором число входящих в его состав элементов минимально.

УЧЕБНЫЕ ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ИХ ВЫПОЛНЕНИЮ

Задание 1. Запустить лабораторный комплекс Labworks и среду MS10 (щёлкнув мышью на команде **Эксперимент** меню комплекса Labworks). Открыть файл **1.2.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания *основных и базовых логических элементов* (см. рис. 1.2) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 1.2) на страницу отчёта.

Схема (рис. 1.2) собрана на двоичных основных [OR (ИЛИ), AND (И) и NOT (НЕ)] и универсальных (базовых) [NAND (И-НЕ) и XOR (ИЛИ-НЕ)] логических элементах, расположенных в библиотеке **Misc Digital/TIL** с уровнем высокого напряжения 5 В. В схему включены ключи 1, 2, ..., 9, пробники X1, X2, ..., X5 с пороговыми напряжениями 5 В, генератор прямоугольных сигналов E1 с амплитудой $E = 5$ В, длительностью импульса $t_u = 0,16$ с и периодом $T = 4$ с, и логический анализатор XLA1 (см. описание его настройки и работы в п. 2, Приложения 2).

Для удобства измерения сигналов выходы логических элементов подключены к входам 2, 4, 6, 8 и 10 анализатора XLA1. При моделировании происходит медленная развёртка временных диаграмм в окне анализатора. По достижению интервала времени, равном 70...80% ширины окна, следует посредством кнопки **Run/Stop** выключать процесс моделирования.

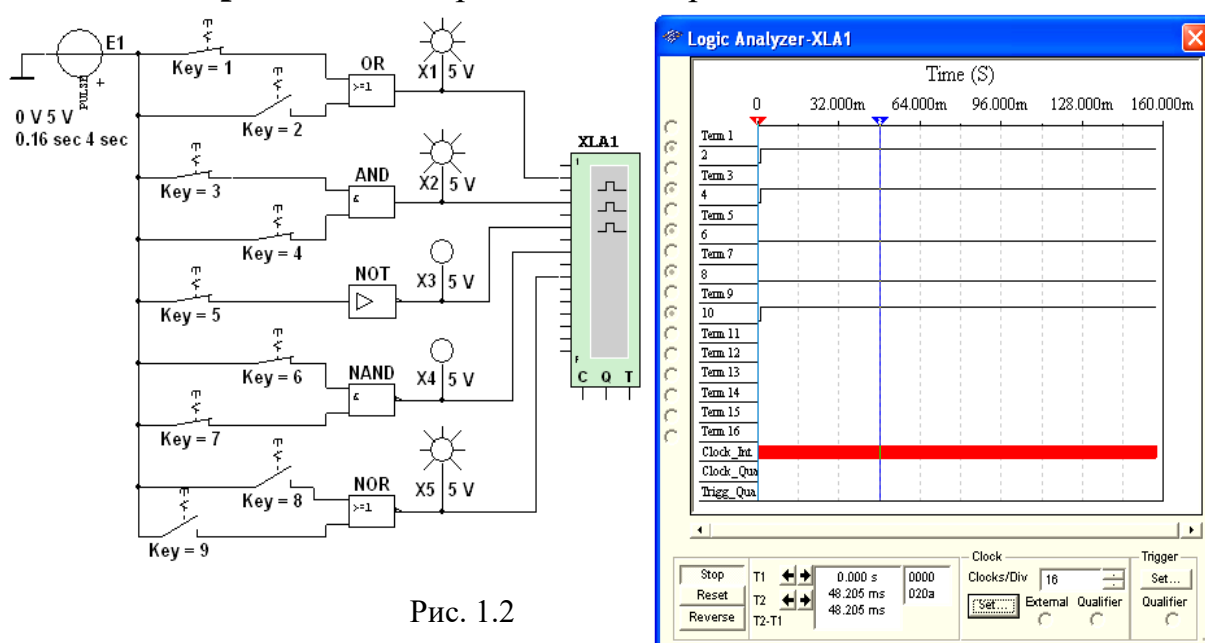


Рис. 1.2

Оперируя ключами 1, 2, ..., 9, сформировать все возможные комбинации аргументов x_1 и x_2 (00, 10, 01 и 11) на входе дизъюнктора (OR), конъюнктора (AND), штриха Шеффера (NAND) и стрелки Пирса (NOR) и записать значения выходных логических функций y_k (0 или 1) в табл. 1.4.

Заметим, что если ключ замкнут, то на этот вход элемента будет подана логическая единица (положительный потенциал 5 В), а при разомкнутом ключе – логический ноль. Поскольку инвертор (NOT) имеет один вход, то для формирования двух значений входного сигнала (логической единицы или логического нуля) достаточно одного ключа 5.

Значения функций исследуемых элементов можно контролировать с помощью пробников X1, X2, ..., X5: если выходной сигнал элемента равен логической единице, то включенный на выходе этого элемента пробник светится. Так, при положении ключей схемы (рис. 1.2) функции элементов OR, AND и NOR равны логической единице.

Т а б л и ц а 1.4

| Дизъюнктор [ИЛИ (OR)] | | | Конъюнктор [И (AND)] | | | Инвертор [НЕ (NOT)] | | Штрих Шеффера [И-НЕ (NAND)] | | | Стрелка Пирса [ИЛИ-НЕ (NOR)] | | |
|--------------------------|-------|-----|-------------------------|-------|-----|------------------------|-----|--------------------------------|-------|-----|---------------------------------|-------|-----|
| x_1 | x_2 | y | x_1 | x_2 | y | x | y | x_1 | x_2 | y | x_1 | x_2 | y |
| 0 | 0 | | 0 | 0 | | 0 | | 0 | 0 | | 0 | 0 | |
| 0 | 1 | | 0 | 1 | | | | 0 | 1 | | 0 | 1 | |
| 1 | 0 | | 1 | 0 | | 1 | | 1 | 0 | | 1 | 0 | |
| 1 | 1 | | 1 | 1 | | | | 1 | 1 | | 1 | 1 | |

Задание 2. "Перетащить" из библиотеки **Misc Digital\TIL** на рабочее поле среды MS10 необходимые логические элементы и **собрать** схему для реализации заданной в табл. 1.5 логической функции y с тремя аргументами a , b и c . **Скопировать** собранную логическую схему на страницу отчёта.

Т а б л и ц а 1.5

| Вариант | Логическая функция |
|-----------------------|---|
| 1, 6, 11, 16, 21, 26 | $y = (\bar{a}b + \bar{c})(\bar{a} + \bar{b} + c)(a + b + c).$ |
| 2, 7, 12, 17, 22, 27 | $y = (a + b + \bar{c})(\bar{a} + \bar{b}c)(a + \bar{b} + \bar{c}).$ |
| 3, 8, 13, 18, 23, 28 | $y = (b + a\bar{c})(\bar{a} + bc)(a + \bar{b} + c).$ |
| 4, 9, 14, 19, 24, 29 | $y = (\bar{a}\bar{b} + \bar{c})(a + \bar{b} + c)(ab + \bar{c}).$ |
| 5, 10, 15, 20, 25, 30 | $y = (a + \bar{b}c)(\bar{a} + b + \bar{c})(ab + c).$ |

В качестве примера соберём схему для реализации логической функции $y = (ab + \bar{c})(\bar{a} + \bar{b} + c)(a + b + c).$

Анализ функции показывает, что для построения логической схемы нам потребуются три инвертора, три дизъюнктора, причем один дизъюнктор с двумя, а два – с тремя входами, и два конъюнктора, причём один с двумя, а другой с тремя входами.

"Перетащим" на рабочее поле среды MS10 необходимые модели логических элементов из библиотеки **Misc Digital\TIL**, располагая их, начиная с входа, а именно:

- три инвертора **NOT** (**NOT1**, **NOT2** и **NOT3**) для получения инверсий \bar{a} , \bar{b} и \bar{c} аргументов a , b и c ;
- конъюнктор **AND1** с двумя входами для реализации функции ab ;
- три дизъюнктора: **OR2** для реализации функции $y_1 = a + b + c$, **OR3** для реализации функции $y_2 = \bar{a} + \bar{b} + c$ и **OR1**, реализующий функцию $y_3 = ab + \bar{c}$, разместив их друг под другом (см. рис. 1.3).

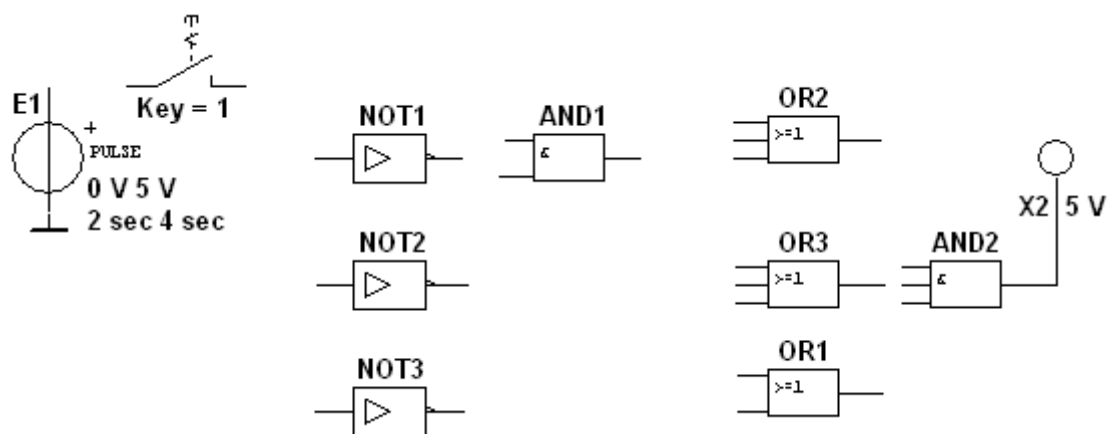


Рис. 1.3

Для выполнения функции логического умножения $y = y_1 y_2 y_3$ добавим в схему конъюнктор **AND2** с тремя входами, к выходу которого подключим логический пробник **X2** (уровень высокого напряжения 5 В) для сигнализации появления логической единицы на выходе схемы. "Перетащим" из соответствующих библиотек на рабочее поле источник прямоугольных сигналов **E1** и ключ **1**, расположив их на входе схемы.

Соединив "проводниками" входы и выходы элементов в соответствии с логическими выражениями составляющих заданной функции и записав в отчёте ожидаемые результаты выполнения операций на выходах элементов (рис. 1.4), приступим к моделированию, открыв файл **1.2.ms10**, размещённый в папке

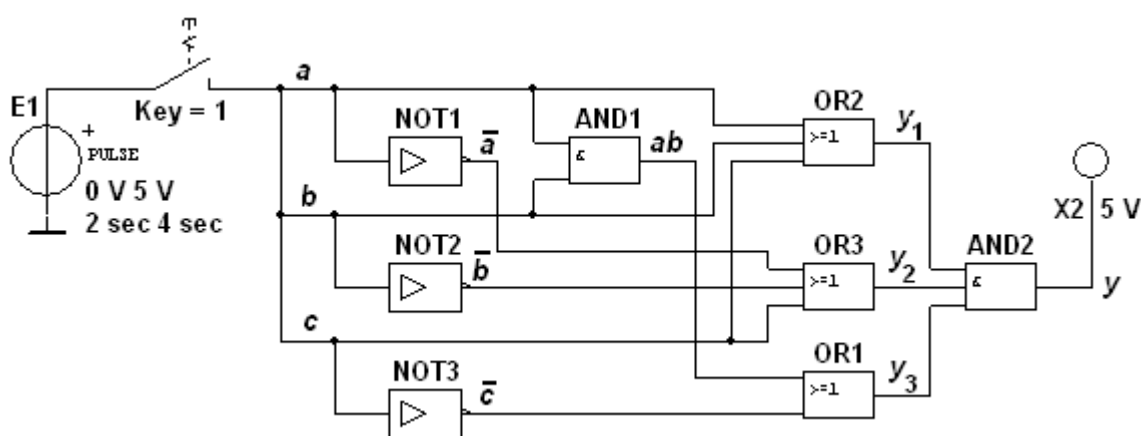


Рис. 1.4

Circuit Design Suite 10.0 среды MS10.

С этой целью вначале щелкнем мышью на кнопке **Run/Stop**, затем нажмём управляющую ключом клавишу с цифрой **1** клавиатуры. Если соединения элементов выполнены правильно, то пробник **X2** засветится. При выключении ключа **1** пробник гаснет и т. д. По окончании моделирования щёлкнем мышью на кнопке **Run/Stop**.

Примечания. 1. Основным измерительным прибором для проверки цифровых электронных схем является логический пробник. После двойного щелчка мышью на его изображении в открывшемся окне нужно задать уровень высокого напряжения, например, 5 В (см. рис 1.4), при котором он светится.

Если пробник не светится, то это обычно означает, что уровень проверяемого напряжения находится в промежутке между высоким и низким. Поиск неисправностей нужно начинать с проверки подачи сигналов высокого уровня генератором сигналов на входы элементов, затем проверить правильность выполнения ими логических функций в схеме и проконтролировать появление сигналов на выходах.

2. Таблицы истинности для рассмотренных библиотечных логических элементов можно вызвать нажатием клавиши помощи **F1** после выделения на схеме соответствующего элемента.

СОДЕРЖАНИЕ ОТЧЕТА

1. Наименование и цель работы.
2. Перечень приборов, использованных в экспериментах, с их краткими характеристиками.
3. Изображения электрической схемы для испытания логических элементов и собранной схемы для реализации заданной логической функции.
4. Таблицы истинности, отображающие работу исследуемых логических элементов.
5. Выводы по работе.

ТЕСТОВЫЕ ЗАДАНИЯ К РАБОТЕ 1

1. Укажите **признаки**, характеризующие основные логические элементы.

- ☐ На входах логических элементов аналоговые сигналы, а на выходах – цифровые
- ☐ Операции логического сложения, логического умножения и инверсия не составляют функционально полный набор
- ☐ Используя основные логические операции И, ИЛИ и НЕ, можно аналитически выразить любую сложную логическую функцию
- ☐ Минимальный логический базис составляют операции ИЛИ и НЕ или И и НЕ
- ☐ Входные и выходные сигналы логических элементов могут принимать только два значения: логическую 1 и логический 0
- ☐ Операция логического сложения совпадает с операцией обычного сложения

2. Укажите **выражение** логической функции двух переменных x_1 и x_2 , реализуемой элементом "Стрелка Пирса".

- ☐ $y = \bar{x}_1 x_2 + x_1 \bar{x}_2$ ☐ $y = \overline{x_1 x_2}$ ☐ $y = \overline{x_1 + x_2}$
☐ $y = x_1 \oplus x_2$ ☐ $y = x_1 + x_2$ ☐ $y = x_1 x_2$

3. Укажите **выражение** логической функции двух переменных x_1 и x_2 , реализуемой элементом "Штрих Шеффера".

- ☐ $y = \bar{x}_1 x_2 + x_1 \bar{x}_2$ ☐ $y = \overline{x_1 x_2}$ ☐ $y = x_1 \oplus x_2$
☐ $y = \overline{x_1 + x_2}$ ☐ $y = x_1 + x_2$ ☐ $y = x_1 x_2$

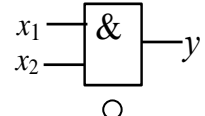
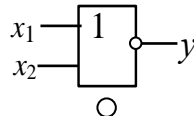
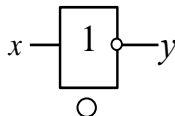
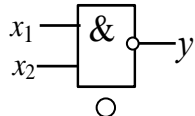
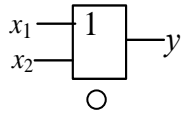
4. Укажите **выражение** логической функции трех переменных a , b и c , записанной в совершенной дизъюнктивной нормальной форме (СДНФ).

$$\bigcirc y(a, b, c) = \bar{a}bc + a\bar{b}c + ab\bar{c} + abc$$

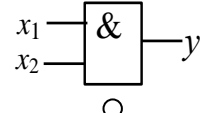
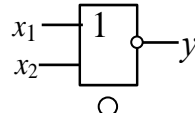
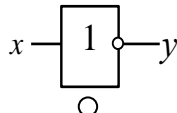
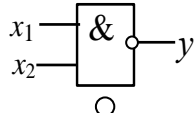
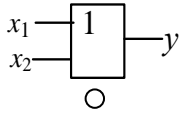
$$\bigcirc y(a, b, c) = (a + b + c)(a + b + \bar{c})(a + \bar{b} + c)(\bar{a} + b + c)$$

$$\bigcirc y(a, b, c) = (\bar{a}b + c + a\bar{b}c)(ab\bar{c} + \bar{a}b + \bar{c}a)$$

5. Укажите элемент ИЛИ-НЕ.



6. Укажите элемент И.



7. Укажите значение функции $y = (ab + \bar{c})(\bar{a} + \bar{b})$, если $a = b = c = 1$.

☐ 1

☐ 0

Лабораторная работа 2 (Lr2)

ИССЛЕДОВАНИЕ УЗЛОВ КОМБИНАЦИОННОГО ТИПА

ЦЕЛЬ РАБОТЫ

Ознакомление с основными характеристиками и испытание интегральных преобразователей кодов (дешифратора, шифратора, демультиплексора и мультиплексора).

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И РАСЧЕТНЫЕ ФОРМУЛЫ

Кодом называют систему символов для представления информации в форме, удобной для обработки, хранения и передачи. В цифровой технике для записи кодовых символов, или просто кода, используют две цифры: 0 и 1. *Преобразователи кодов* служат для перевода одной формы бинарного числа (кодовой комбинации) в другую, например, преобразование двоично-десятичного кода в семисегментный код индикатора. Входные и выходные коды преобразователей связаны между собой. Эту связь задают логическими функциями или в виде таблицы переключений. Рассмотрим наиболее распространённые в цифровой технике виды преобразователей кодов.

1. ДЕШИФРАТОР

Дешифратор (DC) или *декодер* – комбинационная схема с n входами и $m = 2^n$ выходами ($m > n$), преобразующая двоичный входной n -код (кодовое слово) в унитарный. На одном из m выходов дешифратора появляется логическая 1, а именно на том, номер которого соответствует поданному на вход двоичному коду.

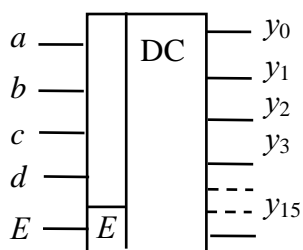


Рис. 2.1

На всех остальных выходах дешифратора выходные сигналы равны нулю. Дешифратор используют, когда нужно обращаться к различным цифровым устройствам по адресу, представленному двоичным кодом.

Условное изображение дешифратора 4x16 (читаемого "четыре в шестнадцать") на схемах дано на рис. 2.1.

Дешифратор содержит число выходов, равное числу

комбинаций входных переменных: от $y_0 = \bar{a}\bar{b}\bar{c}\bar{d}$ до $y_{15} = abcd$ при $n = 4$ и $m = 2^n = 16$.

Применяются также неполные дешифраторы с меньшим числом выходов (10 или 12 при четырех переменных на входе, тогда ряд комбинаций на входе не используется).

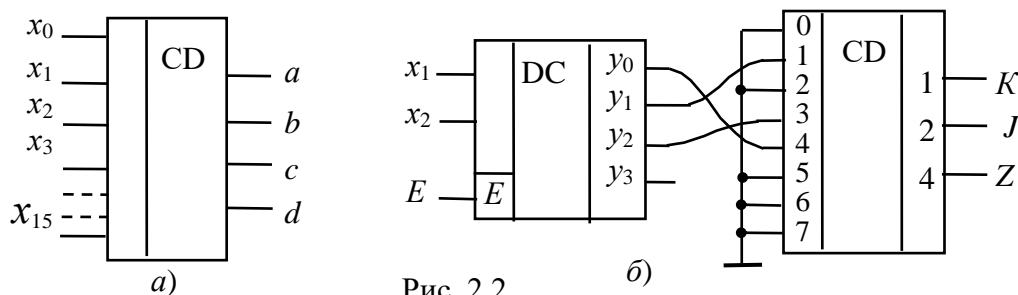
Каждый выход полного дешифратора реализует конъюнкцию входных переменных (код адреса) или их инверсий: при наборе $\bar{a}\bar{b}\bar{c}\bar{d}$ (0000) $y_0 = 1$, при $\bar{a}bcd$ (0111) $y_7 = 1$, при $abcd$ (1111) $y_{15} = 1$ и т. д.

Дешифраторы часто имеют *разрешающий* (управляющий, стробирующий) вход E . При $E = 1$ дешифратор функционирует как обычно, при $E = 0$ на всех выходах устанавливается 0 независимо от поступающего кода адреса.

Дешифраторы широко используют во многих устройствах, в том числе в качестве преобразователей двоичного кода в десятичный.

2. ШИФРАТОР

Шифратор (CD) или *кодер* выполняет функцию, обратную функции дешифратора. Условное изображение шифратора 16x4 (16 в 4) на схемах показано на рис. 2.2, а. Классический шифратор имеет n входов и m выходов ($m < n$), и при подаче сигнала 1 на один из входов (и не более) на выходе кодера появляется двоичный код номера возбужденного выхода. Число входов и выходов такого шифратора связано соотношением $n = 2^m$.



Области использования шифраторов – отображение в виде двоичного кода номера нажатой кнопки или положения многопозиционного переключателя, а также номера устройства, подавшего сигнал на обслуживание в микропроцессорных системах. Шифраторы входят в состав микросхем контроллеров прерываний, например КР580ВН59.

Для решения многих конкретных задач необходимо синтезировать преобразователи различных кодов. В качестве примера на рис. 2.2, б представлена схема кодового преобразователя, состоящая из пары декодер DC – кодер CD, реализующая логику работы ($y = a + b\bar{c} + \bar{b}c$) некоторого трёхцветного светофора K , J и Z , управляемого двухразрядным двоичным кодом X . При этом вначале дешифруется каждая комбинация исходного кода, в результате чего на соответствующем выходе декодера появляется логическая 1. Затем этот логический сигнал, значение которого определено номером выхода декодера, подаётся на кодер и на его выходах устанавливается преобразованный код.

Число входов дешифратора DC равно двум (x_1 и x_2), число выходов – трём (числу выходов преобразователя) y_0 , y_1 и y_2 . Соединения дешифратора и шифратора выполнены в соответствии с заданной логической функцией y . Часть выходов декодера и входов кодера не используется.

Эффективно стыкуются друг с другом декодер и кодер, построенные на элементах И-НЕ: первый имеет инверсные выходы, а второй – инверсные входы. Если некоторым входным комбинациям соответствует одна и та же выходная, то соответствующие выходы декодера объединяют на элементе ИЛИ и выход последнего подают на нужный вход кодера.

Проектирование кодовой преобразовательной схемы на паре декодер-кодер оказывается в среднем более выгодным и по числу корпусов, и по быстродействию, чем при проектировании из готовых базовых логических микросхем И-НЕ и ИЛИ-НЕ. Однако потребляемая мощность в этом случае может оказаться больше, чем у схемы из отдельных элементов. Затраты времени инженера на логическое проектирование по схеме декодер-кодер неизмеримо меньше, чем затраты на проектирование преобразователя из россыпи.

3. МУЛЬТИПЛЕКСОР

Мультиплексор (MS) – это функциональный узел, осуществляющий подключение (*коммутацию*) одного из нескольких входов к выходу y . На выход такого устройства передаётся логический уровень того информационного разряда, номер которого в двоичном коде задан на адресных входах x_1 и x_2 . Условное изображение мультиплексора на четыре входа и возможный вариант его структурной схемы показаны на рис. 2.3, а и б.

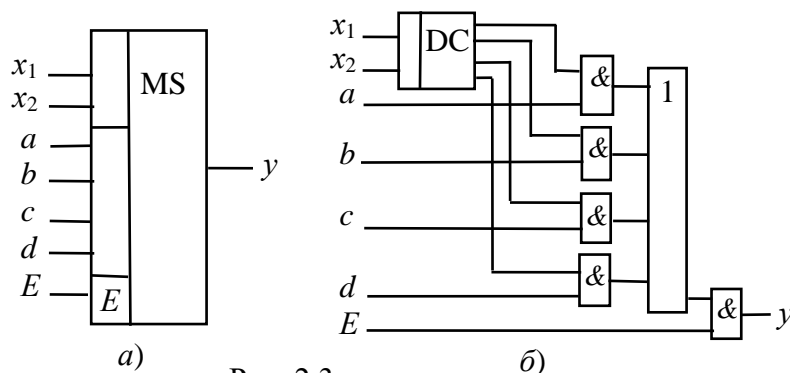


Рис. 2.3

При $x_1 = 0$ и $x_2 = 0$, $y = a$; при $x_1 = 0$ и $x_2 = 1$, $y = b$; при $x_1 = 1$ и $x_2 = 0$, $y = c$ и при $x_1 = 1$ и $x_2 = 1$, $y = d$.

Функционирование мультиплексора описывается выражением

$$y = a\bar{x}_1\bar{x}_2 + b\bar{x}_1x_2 + cx_1\bar{x}_2 + dx_1x_2.$$

Вход E – разрешающий: при $E = 1$ мультиплексор работает как обычно, при $E = 0$ выход узла находится в неактивном состоянии, мультиплексор заперт. Серийные узлы выпускаются с числом адресных входов $n = 2, 3$ и 4 при возможном числе 2^n коммутируемых входов. При необходимости коммутировать большее количество входов используют несколько мультиплексоров. Мультиплексоры находят широкое применение в устройствах отображения информации в различных устройствах управления.

Так как мультиплексор может пропустить на выход сигнал с любого информационного входа, адрес которого установлен на соответствующих адресных входах, то на основе мультиплексоров реализуют логические функции, подавая на информационные входы логические 1 или 0 в соответствии с таблицей переключений, а на адресные входы – аргументы функции.

4. ДЕМУЛЬТИПЛЕКСОР

Демультиплексор (DMS) выполняет функцию, обратную функции мультиплексора, т. е. производит коммутацию одного входного сигнала на 2^n выходов, где n – число адресных входов x_i . Он осуществляет преобразование информации из последовательной формы (последовательно-параллельной) в параллельную. Демультиплексор имеет один информационный вход D и несколько выходов, причем вход подключается к выходу y_i , имеющему заданный адрес.

В качестве примера на рис. 2.4, а дано условное графическое обозначение демультиплексора, имеющего четыре выхода, закон функционирования которого задан (табл. 2.1). Пользуясь табл. 2.1, запишем переключательные функции для выхода устройства:

$$y_0 = D\bar{x}_1\bar{x}_2; y_1 = D\bar{x}_1x_2; y_2 = Dx_1\bar{x}_2; y_3 = Dx_1x_2.$$

Функциональная схема демультиплексора, реализующая эти выражения, приведена на рис. 2.4, б.

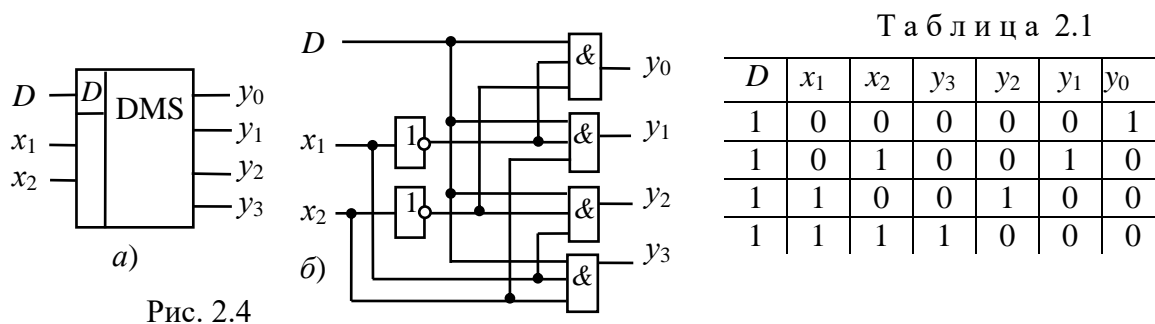


Рис. 2.4

Если общее число выходов разрабатываемого устройства превышает имеющиеся в выпускаемых интегральных микросхемах, то используют параллельное подключение

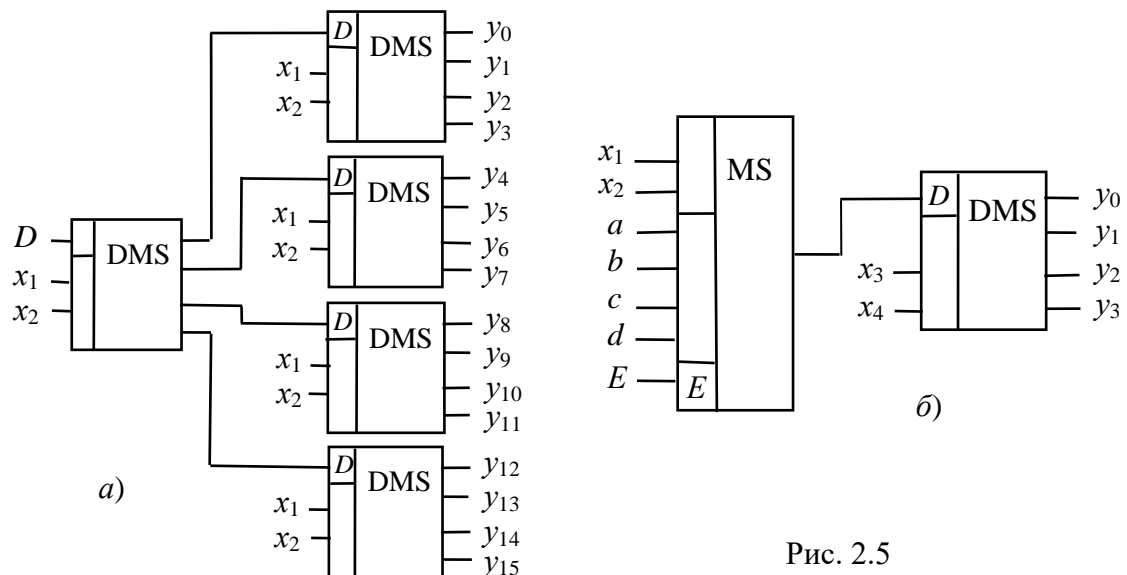


Рис. 2.5

нескольких схем. На рис. 2.5, а показано демультиплексорное дерево, построенное на мультиплексорах с четырьмя выходами. Объединяя мультиплексор с демультиплексором, получают комбинационное устройство, в

котором по заданным адресам один из входов подключается к одному из его выходов (рис. 2.5, б).

УЧЕБНЫЕ ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ИХ ВЫПОЛНЕНИЮ

Задание 1. Запустить лабораторный комплекс Labworks и среду MS10 (щёлкнув мышью на команде **Эксперимент** меню комплекса Labworks). Открыть файл **2.6.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания **дешифратора DC** (рис. 2.6) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 2.6) на страницу отчёта.

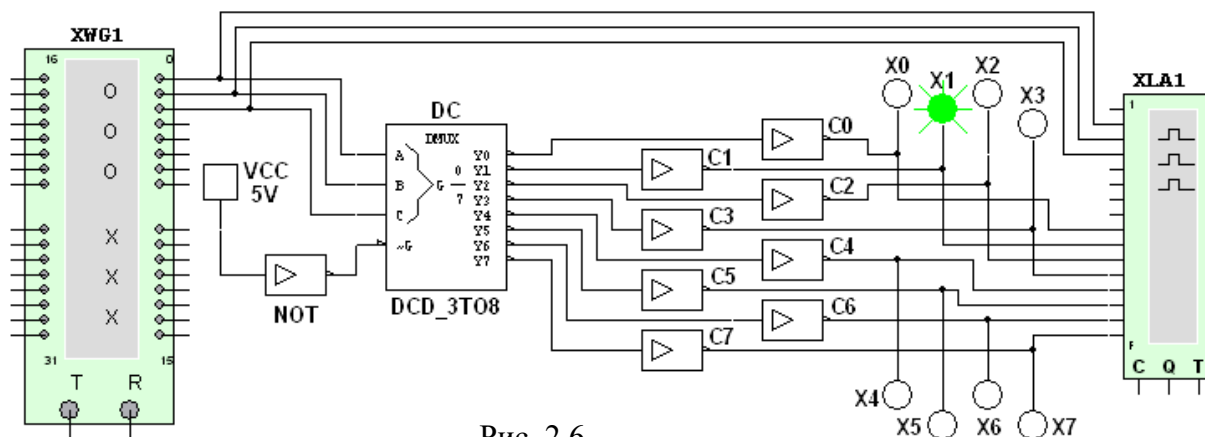


Рис. 2.6

Схема (рис. 2.6) содержит:

- интегральный **дешифратор DC** (*decoder*) 3x8, имеющий 3 информационных входа **A**, **B** и **C** (для кода 4–2–1), 8 выходов (**Y0**, ..., **Y7**) и преобразующий позиционный 3-разрядный двоичный код в *унитарный* "1 из 8": в выходной 8-разрядной кодовой комбинации только одна позиция занята единицей, а все остальные – нулевыми (см. рис. 2.7, справа). В зависимости от входного двоичного кода, например 001, на выходе **DC** появляется сигнал 1 только на одной (второй, см, рис. 2.6) из 8-ми выходных линий, к которым подключены пробники **X0**, ..., **X7**.

Данный тип шифратора относится к шифраторам с разным уровнем входных и выходных сигналов: активные входные уровни соответствуют уровню логической 1, а активные выходные сигналы – уровню логического 0. Для получения активных выходных уровней, равных 1, к выходам дешифратора подключено восемь инверторов **C0**, ..., **C7**;

- логический генератор слова **XWG1** ($f_2 = 500$ кГц) с записанными логическими словами в его ячейки памяти, которые эквивалентны десятичным числам от 0 до 7 (см. рис. 2.7, слева);

- логический анализатор **XLA1**, на экран которого выводятся временные диаграммы как трёх входных (**A**, **B**, **C**), так и восьми (**Y0**, **Y1**, ..., **Y7**) выходных сигналов при пошаговом режиме **Step** генератора **XWG1**;

- источник **VCC**, напряжение 5 В с выхода которого подано на инвертор **NOT**. Логический 0 с инвертора подается на управляющий вход дешифратора **DC**: при $\bar{G} = 0$ дешифратор находится в активном состоянии.

Запустить программу моделирования дешифратора. Щёлкая мышью на кнопке **Step** генератора **XWG1**, последовательно **подавать** на вход дешифратора логические слова. **Убедиться**, что при подаче на вход дешифратора каждой новой двоичной кодовой комбинации засвечивается только один пробник, который "распознаёт" свой входной код.

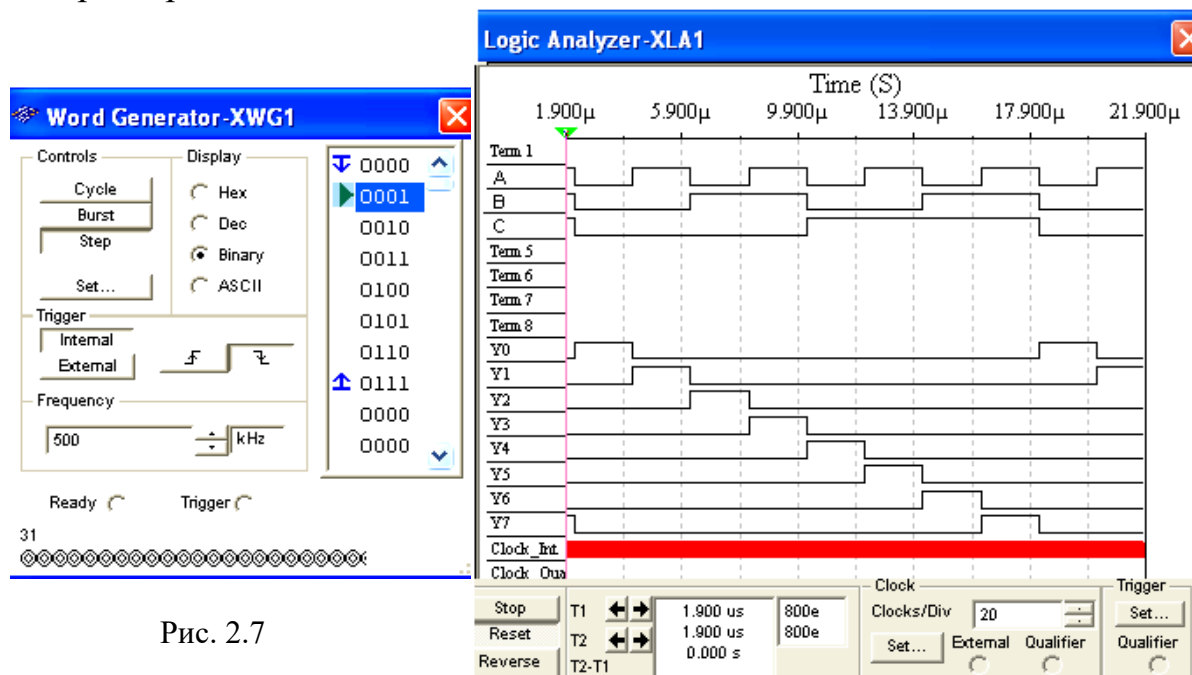


Рис. 2.7

Скопировать временные диаграммы входных и выходных сигналов дешифратора на страницу отчёта. По результатам моделирования **составить** и **заполнить** таблицу переключений (функций $Y_i = (A_i B_i C_i; G_i)$) на выходах дешифратора **DC** 3x8.

Задание 2. Открыть файл **2.8.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания *шифратора* **CD** (рис. 2.8) и **установить** в диалоговых окнах компонентов их параметры или режимы работы.

Скопировать схему (рис. 2.8) на страницу отчёта.

Интегральный *шифратор* **CD** 8x3 (из 8 в 3) имеет 8 входов **D0**, **D1**, ..., **D7**, подключённых к выходам **Y0**, **Y1**, ..., **Y7** дешифратора **DC**, и три инверсных выхода **A0**, **A1**, **A2**, к которым через инверторы **C0**, **C1**, **C2** подключены логические пробники **X0**, **X1**, **X2** и семисегментный индикатор **Ind**. Содержимое ячеек памяти генератора слова **XWG1**: 000, 001, ..., 111 (см. рис. 2.7, слева).

Запустить программу моделирования шифратора. Щёлкая мышью на кнопке **Step** генератора **XWG1**, последовательно **подавать** на вход дешифратора логические слова. **Убедиться**, что при подаче с выхода **DC** на вход шифратора **CD** 8-разрядной последовательности, в которой только одна позиция занята

единицей, а остальные – нулями, на выходе шифратора формируются 3-разрядные двоичные коды **A0A1A2**, где **A0 = A**, **A1 = B** и **A2 = C**, соответствующие двоичным кодовым комбинациям на входе дешифратора **DC**.

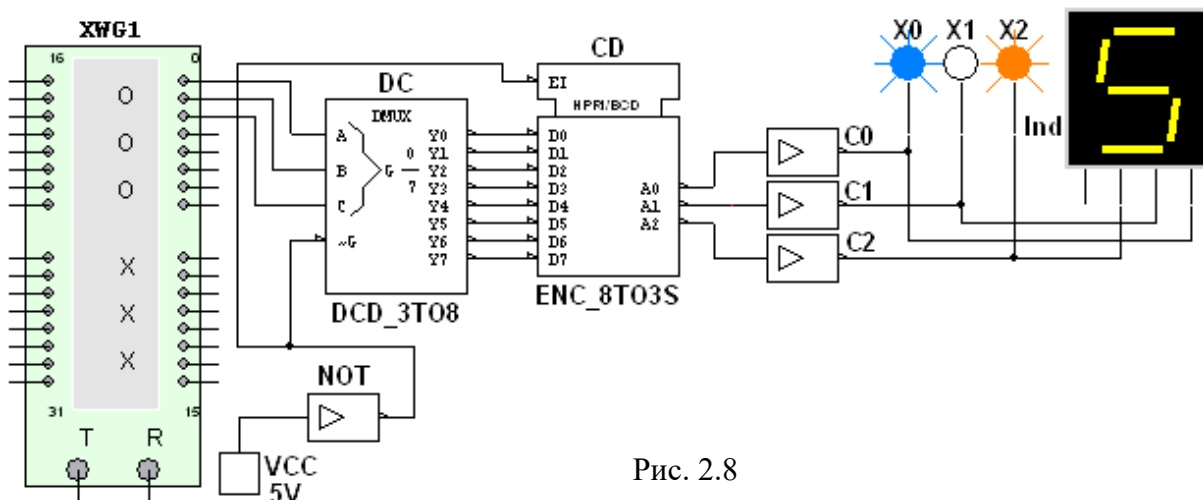


Рис. 2.8

По
рез
ул
та
та
м
де
ли
ро
ва
ни

я (по засвечиванию логических пробников **X0**, **X1**, **X2** и показаниям индикатора **Ind**) **составить** и **заполнить** таблицу переключений на выходе шифратора **CD 8x3**.

Преобразовать схему дешифратора **DC 3x8** и шифратора **CD 8x3** (см. рис. 2.8) в схему **DC 2x4** и шифратора **CD 4x2**, отсоединив провод **C**, подходящий к дешифратору, и провод **A2** с выхода шифратора, и **составить** таблицы переключений дешифратора **2x4** и шифратора **4x2**.

Задание 3. Открыть файл **2.9.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания **демультиплексора DMS** (рис. 2.9) и **установить** в диалоговых окнах компонентов их параметры или режимы работы.

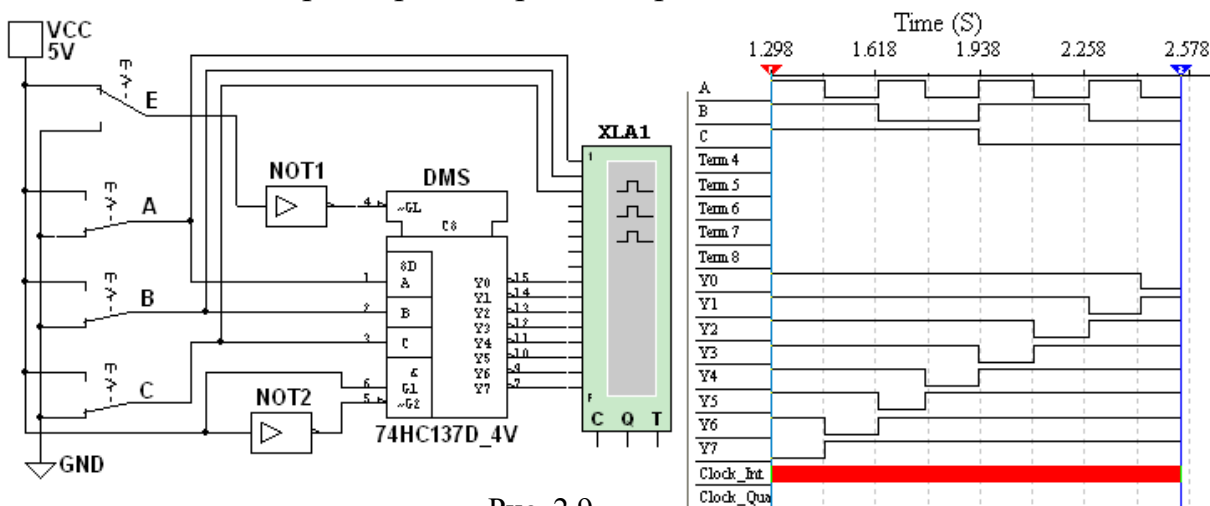


Рис. 2.9

Демультиплексор DMS 1x8 (из 1 в 8) (рис. 2.9) имеет один информационный вход (с активными высоким **G1** и низким **G2** уровнями), три адресных **A**, **B**, **C** входа, разрешающий **GL** вход с активным низким уровнем и восемь **Y0**, **Y1**, ..., **Y7** инверсных выходов, соединённых с входами логического анализатора **XLA1**. На вход анализатора также подаются сигналы с адресных входов **A**, **B**,

С. С помощью ключей **A**, **B** и **C** можно сформировать восемь трёхразрядных двоичных адресных слов. При последовательной подаче формируемых ключами адресных слов от 111 до 000 на экран анализатора **XLA1** при моделировании выводятся 8-разрядные кодовые последовательности с одним активным (низким) уровнем.

Для обеспечения медленного перемещения лучей на экране анализатора **XLA1** **установить** частоту его таймера $f_a = 500$ Гц и число импульсов, приходящихся на одно деление, **Clocs/div** = 80.

Задать код ключей 111 и **щелкнуть** мышью на кнопке **Run/Stop**. Кривые адресных и выходных логических сигналов медленно разворачиваются во времени на экране анализатора.

Остановить (щелчком мыши на кнопке **Stop**) процесс моделирования при приближении лучей анализатора к линии разметки экрана.

Повторять перечисленные выше операции для спадающих счётных комбинаций адресных сигналов (с 110 до 000) до тех пор, пока не будет записан процесс моделирования при адресном слове 000 (см. рис. 2.9, справа).

Убедиться, что для каждой комбинации адресных сигналов демультиплексор формирует логический 0 на одном из восьми выходов, номер которого соответствует определенному кодовому слову на входе, т. е. демультиплексор подобен коммутатору, посредством которого поток цифровой информации разделяется на 8 выходных потоков.

Скопировать схему (рис. 2.9) и временные диаграммы входных и выходных сигналов на страницу отчёта.

Если адресные входы **A**, **B** и **C** принять в качестве информационных входов, а вход **G1** (**G2**) в качестве входа разрешения работы, то мультиплексор превратится в дешифратор.

Задание 4 (выполняется факультативно). **Открыть** файл **2.10.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания *демультиплексора* **DMS** 1x16 (из 1 в 16) (рис. 2.10) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 2.10) в отчёт.

С целью автоматизации процесса моделирования к входу демультиплексора **DMS** подключен логический генератор **XWG1** с записанными в его ячейки памяти адресными кодами от 0000 до 1111, а для визуализации сигналов на выходах включены 16 логических пробников **X1**, **X2**, ..., **X16** и логический анализатор **XLA2**.

Запустить программу моделирования демультиплексора **DMS 1x16**. Последовательно **подавать** (щелкая мышью на кнопке **Step** генератора **XWG1**) на вход демультиплексора логические слова, начиная с комбинации 0000 адресного сигнала и заканчивая комбинацией 1111, и **наблюдать** за изменениями выходных сигналов по показаниям индикаторов и в окне анализатора **XLA2**.

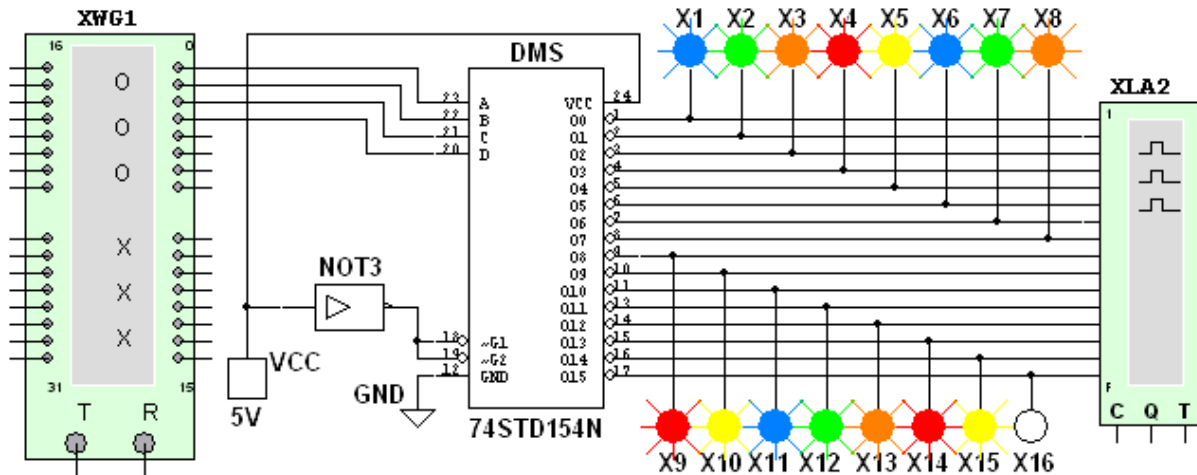


Рис. 2.10

В исследуемой модели демультиплексора соответствующий активный выход имеет низкий логический уровень (рис. 2.11), поэтому пробник на этом выходе

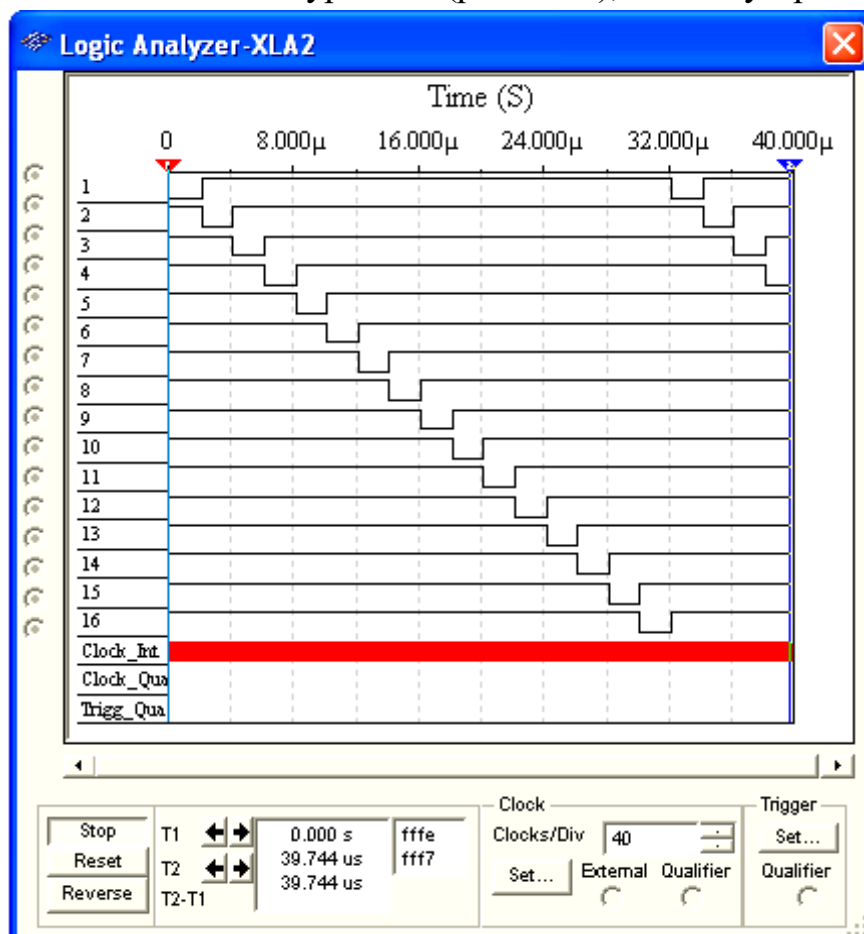


Рис. 2.11

не светится. Так, при подаче последней кодовой комбинации 1111 на вход демультиплексора не светится пробник **X16**, так как активным является выход **15** (см. рис. 2.10).

Скопировать на страницу отчёта временные диаграммы выходных сигналов демультиплексора **DMS 1x16**.

Примечание. Демультиплексоры как таковые промышленностью не выпускаются, поскольку режим мультиплексирования может быть реализован как частный случай в других устройствах – в дешифраторах.

Задание 5. Открыть файл **2.12.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему (рис. 2.12) для испытания *мультиплексора MS 8x1* (из 8 в 1) и **установить** в диалоговых окнах компонентов их параметры или режимы работы.

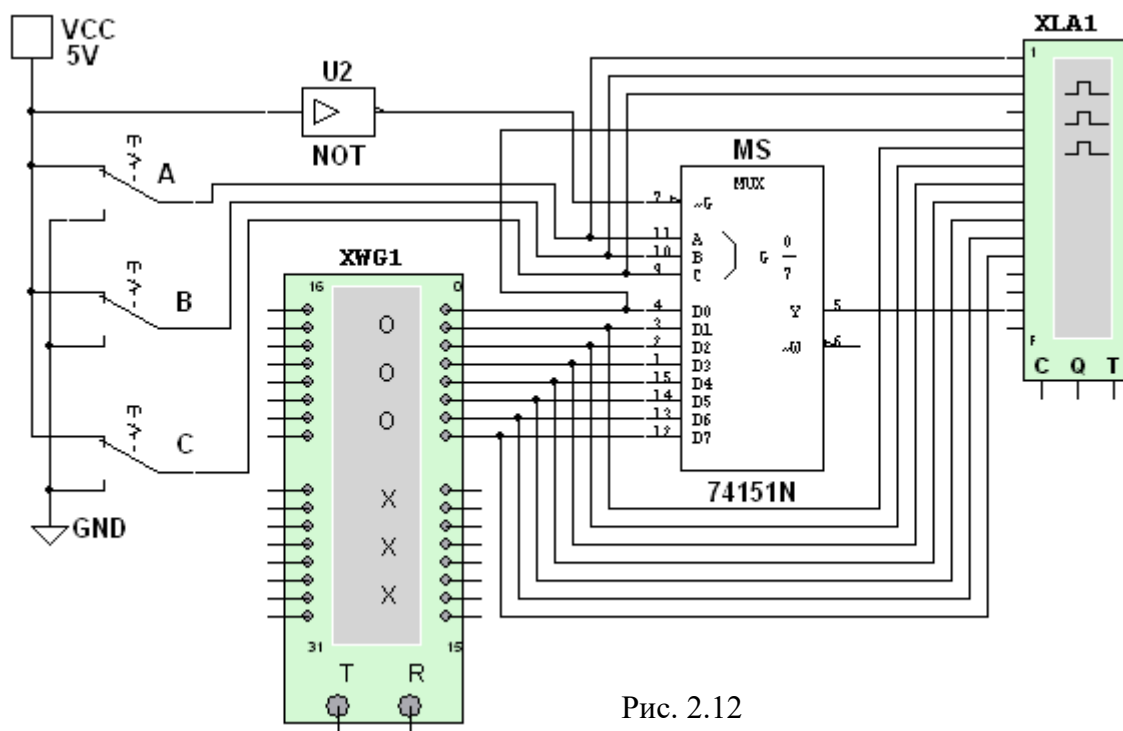


Рис. 2.12

Скопировать схему (рис. 2.12) на страницу отчёта.

Мультиплексор **MS** с разрешающим входом **G** осуществляет передачу сигнала с каждого информационного входа **D0**, **D1**, ..., **D7**, заданного 3-разрядным кодом **ABC** – адресом выбираемого входа, на единственный выход **Y**. Разрядность (3) управляющего сигнала определяет количество входов ($2^3 = 8$), с которых мультиплексор может принимать информацию. Если предположить, что к входам **D0**, **D1**, ..., **D7** мультиплексора **MS** присоединено 8 источников цифровых сигналов – генераторов последовательных двоичных слов, то байты от любого из них можно передавать на выход **Y**.

Для иллюстрации работы мультиплексора **MS** запишем в ячейки памяти генератора **XWG1** произвольные 8-разрядные кодовые слова (рис. 2.13, слева), а с помощью ключей **A**, **B**, **C** сформируем управляющий сигнал 111. Последовательно щёлкая мышью на кнопке **Step** генератора **XWG1** и при **G** = 1, поступающие на вход **D7** мультиплексора байты (сигнал 01001110) с 8-го

разряда (на рис. 2.13, слева 8-й разряд показан стрелкой) логических слов генератора **XWG1** передаются на выход **Y** и на вход анализатора (см. рис. 2.13, справа).

Если ключ **A** установить в нижнее положение (сформировав, тем самым, адресный код 011), то с входа **D3** на выход **Y** мультиплексора будут поступать байты 4-го разряда логических слов, записанных в ячейки памяти генератора **XWG1**, и т. д.

Записать в первые восемь ячеек памяти генератора **XWG1** произвольные 8-разрядные кодовые слова, **здать** частоту $f_2 = 500$ кГц и режим **Step** его работы (см. рис. 2.13, слева).

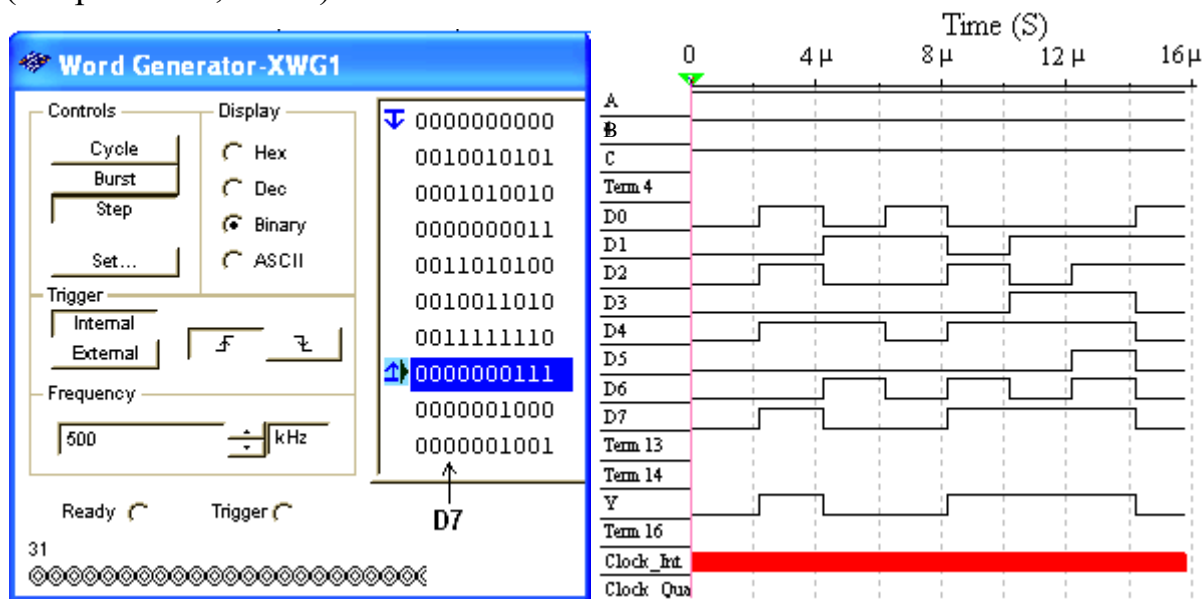


Рис. 2.13

Задать частоту $f_a = 20$ МГц таймера логического анализатора **XLA1** и количество импульсов таймера **Clock/div** = 20, приходящихся на одно деление.

Установить с помощью ключей **A**, **B** и **C** адресный код (самостоятельно или по указанию преподавателя), например 100_2 (4_{10}) и **запустить** программу моделирования мультиплексора. **Получить** и **скопировать** временные диаграммы входных сигналов **D0**, **D1**, ..., **D7** и выходного сигнала **Y** мультиплексора на страницу отчёта.

Примечание. Таблицы переключений на выходах для рассмотренных библиотечных преобразователей кодов можно вызвать нажатием клавиши помощи **F1** после выделения на схеме соответствующего преобразователя.

СОДЕРЖАНИЕ ОТЧЁТА

1. Наименование и цель работы.
2. Перечень приборов, использованных в экспериментах, с их краткими характеристиками.
3. Изображения электрических схем для испытания дешифратора, шифратора, демультиплексора и мультиплексора.

4. Копии временных диаграмм и таблицы переключений, отображающие работу исследуемых преобразователей кодов.
5. Выводы по работе.

ТЕСТОВЫЕ ЗАДАНИЯ К РАБОТЕ 2

1. Укажите задачи:

- а) Для демультимплексирования данных и адресной логики в запоминающих устройствах, а также для преобразования двоично-десятичного кода в десятичный с целью управления индикаторными и печатающими устройствами;
- б) Для преобразования десятичных чисел в двоичные или в двоично-десятичный код, например, в микрокалькуляторах, в которых нажатие десятичных клавиш вызывает генерацию соответствующих двоичных кодов;
- в) Для хранения и преобразования многоразрядных двоичных чисел;
- г) Для коммутации в заданном порядке сигналов, поступающих с нескольких входных шин на одну выходную;
- д) Для распределения в требуемой последовательности по нескольким выходам сигналов с одного информационного входа, в частности, для передачи информации по одной линии от нескольких установленных на ней датчиков,

при решении которых используется:

1. Шифратор: ☐ а) ☐ б) ☐ в) ☐ г) ☐ д)
2. Дешифратор: ☐ а) ☐ б) ☐ в) ☐ г) ☐ д)
3. Мультиплексор: ☐ а) ☐ б) ☐ в) ☐ г) ☐ д)
4. Демультимплексор: ☐ а) ☐ б) ☐ в) ☐ г) ☐ д)

2. Укажите, с **какого разряда** бинарного слова генератора логического слова XWG будет передаваться информация на выход мультиплексора 8х3 при адресном коде 100 на его входе?

- ☐ 1 ☐ 3 ☐ 5 ☐ 7 ☐ 9

3. Укажите число **выводов** дешифратора при трёх информационных входах.

- ☐ 2 ☐ 4 ☐ 6 ☐ 8 ☐ 16

4. Укажите назначение **стробирующих** входов в преобразователях кодов.

- ☐ Для синхронизации работы преобразователей
- ☐ Для увеличения числа коммутируемых информационных входов, а также для блокирования работы преобразователей
- ☐ Для увеличения числа адресных входов

5. Укажите, в каком **преобразователе** выбор входа по его номеру (адресу) осуществляется с помощью двоичного кода?

- ☐ В шифраторе ☐ В дешифраторе ☐ В мультиплексоре ☐ В демультимплексоре

6. Укажите **число выводов** у шифратора при четырёх информационных входах.

- ☐ 16 ☐ 8 ☐ 4 ☐ 2 ☐ 1

7. Укажите, какой из приведенных преобразователей кодов выпускается промышленностью только в **составе других устройств**?

- ☐ Шифратор ☐ Дешифратор ☐ Демультимплексор ☐ Мультиплексор

Лабораторная работа 3 (Lr3)

ЦИФРОВОЙ КОМПАРАТОР

ЦЕЛЬ РАБОТЫ

Ознакомление с основными характеристиками и испытание интегрального цифрового компаратора.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И РАСЧЕТНЫЕ ФОРМУЛЫ

Цифровой компаратор предназначен для сравнения двух многоразрядных двоичных чисел. В простейшем случае требуется лишь установить факт равенства бинарных чисел A и B одинаковой разрядности. При n -разрядных числах компаратор состоит из n сумматоров по модулю 2, выходы которых подключены к элементу ИЛИ. Только при совпадении значений всех разрядов чисел A и B на выходах всех сумматоров будет 0. Если же числа отличаются хотя бы в одном разряде, то на выходе соответствующего сумматора и, следовательно, на общем выходе будет 1.

Операция поразрядного сравнения заключается в выработке признака равенства (равнозначности) или неравенства (неравнозначности) двух сравниваемых двоичных чисел. Два числа равны при равенстве цифр в одноименных разрядах: $a_i = b_i$, где a_i – цифра в i -м разряде одного числа, b_i – цифра в i -м разряде другого числа. Равенство $a_i = b_i$ имеет место при $a_i = 1, b_i = 1$ или при $a_i = 0, b_i = 0$. Поэтому логическая функция, выражающая это равенство, равна единице, если единице равно произведение этих цифр или произведение их инверсных значений, т. е.

$$y = a_i b_i + \bar{a}_i \bar{b}_i,$$

а логическая функция, описывающая компаратор для n -разрядных чисел, имеет вид

$$y = (a_1 b_1 + \bar{a}_1 \bar{b}_1)(a_2 b_2 + \bar{a}_2 \bar{b}_2) \dots (a_n b_n + \bar{a}_n \bar{b}_n).$$

Для построения компаратора только на элементах И-НЕ запишем её в другой форме, воспользовавшись формулой де Моргана,

$$\bar{y} = \overline{(a_1 b_1 + \bar{a}_1 \bar{b}_1)} \overline{(a_2 b_2 + \bar{a}_2 \bar{b}_2)} \dots \overline{(a_n b_n + \bar{a}_n \bar{b}_n)}.$$

Схема, реализующая это выражение, приведена на рис. 3.1, а.

Если необходимо, чтобы при равенстве кодов на выходе компаратора была логическая 1, то к выходу схемы (рис. 3.1, а) следует присоединить инвертор.

В некоторых компараторах находит применение узел сравнения чисел с определением знака неравенства, т. е. $A > B$ или $A < B$. Устройство компаратора в этом случае получается более сложным. Число входов его равно $2n$, а число выходов три: $Y_>$ при $A > B$, $Y_=$ при $A = B$, $Y_<$ при $A < B$.

Компараторы выполняют в виде отдельных микросхем. Так, например, микросхема К564ИП2 позволяет сравнивать два четырёхразрядных числа с определением знака неравенства. Условное обозначение такой микросхемы приведено на рис. 3.1, б.

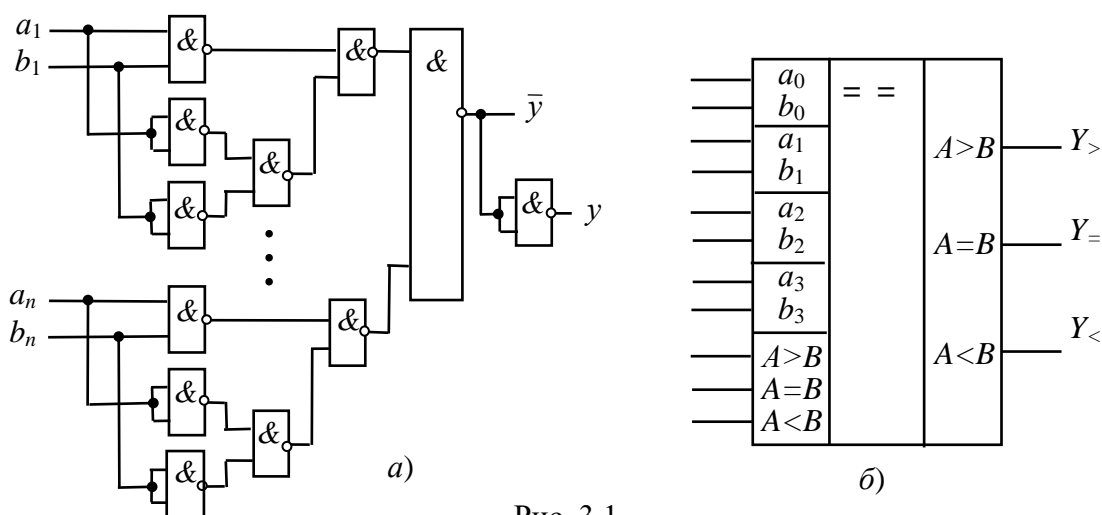


Рис. 3.1

Данный тип компаратора обладает свойством наращиваемости. Для сравнения, например, 8-разрядных чисел применяют два 4-разрядных компаратора. Для этой цели как в микросхеме К564ИП2, так и в некоторых других марках отечественных и зарубежных производителей, предусмотрены три дополнительных входа: $A > B$, $A = B$ и $A < B$, к которым подводятся соответствующие выходы микросхемы, выполняющей сравнение младших разрядов.

УЧЕБНЫЕ ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ИХ ВЫПОЛНЕНИЮ

Задание 1. Запустить лабораторный комплекс Labworks и среду MS10 (щёлкнув мышью на команде **Эксперимент** меню комплекса Labworks). Открыть файл **3.2.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или собрать на рабочем поле среды MS10 схему для испытания *цифрового компаратора* (рис. 3.2) и установить в диалоговых окнах компонентов их параметры или режимы работы. Скопировать схему (рис. 3.2) на страницу отчёта.

Цифровой 4-разрядный компаратор **Comp4** (рис. 3.2) выполняет сравнение четырёх старших разрядов 8-разрядных бинарных чисел **A** и **B** с учётом результатов сравнения младших разрядов, подаваемых на входы **AGTB** ($A > B$), **AEQT** ($A = B$) и **ALTB** ($A < B$) с соответствующих выводов первой микросхемы компаратора. На входы **A3**, **A2**, **A1**, **A0** и **B3**, **B2**, **B1**, **B0** микросхемы **Comp4** поступают с генератора слова **XWG1** сигналы четырёх старших разрядов чисел **A** и **B**.

Сигналы сравнения 8-разрядных чисел с определением их равенства $A = B$ или неравенства $A > B$, $A < B$ подаются на выходы **OAGTB** ($A > B$), **OAEQT** ($A = B$) и **OALTB** ($A < B$). К этим выходам подключены входы логического анализатора **XLA1** и логические пробники **X1**, **X2** и **X3**.

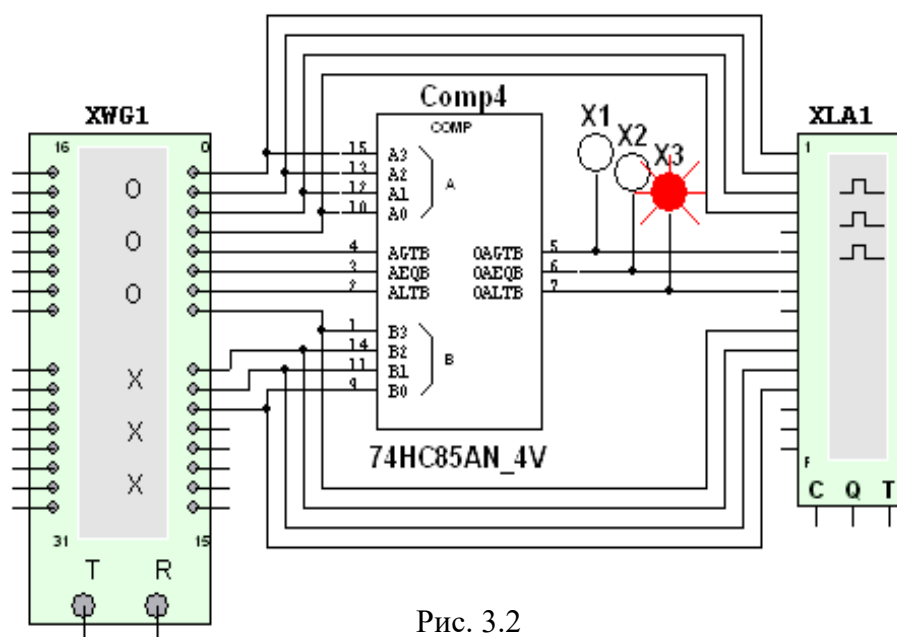


Рис. 3.2

При сравнении многоразрядных двоичных чисел используется следующий алгоритм. Сначала сравниваются значения старших разрядов. Если они различны, то эти разряды и определяют результат сравнения. Если они равны, то необходимо сравнить следующие за ними младшие разряды и т. д.

Компаратор **74HC85AN_4V** реализует указанный алгоритм: соответствующие логические функции приведены в таблице истинности (табл. 3.1), выводимой на экран дисплея после выделения изображения компаратора на схеме (рис. 3.2) и нажатия клавиши помощи F1 клавиатуры.

Таблица 3.1

| COMPARING INPUTS (Сравниваемые 4-разрядные двоичные числа) | | | | CASCADING INPUTS (Результат сравнения младших разрядов) | | | OUTPUTS (Выходы) | | |
|---|--------|--------|--------|--|-----|-----|---------------------|-----|-----|
| A3, B3 | A2, B2 | A1, B1 | A0, B0 | A>B | A=B | A<B | A>B | A=B | A<B |
| A3>B3 | X | X | X | X | X | X | 1 | 0 | 0 |
| A3<B3 | X | X | X | X | X | X | 0 | 0 | 1 |
| A3=B3 | A2>B2 | X | X | X | X | X | 1 | 0 | 0 |
| A3=B3 | A2<B2 | X | X | X | X | X | 0 | 0 | 1 |
| A3=B3 | A2=B2 | A1>B1 | X | X | X | X | 1 | 0 | 0 |
| A3=B3 | A2=B2 | A1<B1 | X | X | X | X | 0 | 0 | 1 |
| A3=B3 | A2=B2 | A1=B1 | A0>B0 | X | X | X | 1 | 0 | 0 |
| A3=B3 | A2=B2 | A1=B1 | A0<B0 | X | X | X | 0 | 0 | 1 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | 1 | 0 | 0 | 1 | 0 | 0 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | 0 | 0 | 1 | 0 | 0 | 1 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | 0 | 1 | 0 | 0 | 1 | 0 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | X | 1 | X | 0 | 1 | 0 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | 1 | 0 | 1 | 0 | 0 | 0 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | 0 | 0 | 0 | 1 | 0 | 1 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | 0 | 1 | 1 | 0 | 1 | 1 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | 1 | 1 | 0 | 1 | 1 | 0 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | 1 | 1 | 1 | 1 | 1 | 1 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | 1 | 0 | 1 | 1 | 0 | 1 |
| A3=B3 | A2=B2 | A1=B1 | A0=B0 | 0 | 0 | 0 | 0 | 0 | 0 |

Примечание. X – любое состояние

Задание 2. Получить временные диаграммы входных и выходных сигналов на экране анализатора **XLA1** при пошаговой подаче на входы компаратора сигналов с выходов генератора слова **XWG1** ($f_c = 500$ кГц).

Для этого:

- щёлкнуть мышью на изображении генератора **XWG1** (см. рис. 3.2) и записать в его первые ячейки памяти 10 произвольных (или заданных преподавателем) 11-разрядных кодовых последовательностей, причём в первые четыре разряда записать (справа налево) значения (1 или 0) числа **A**, т. е. A3A2A1A0, в следующие три разряда – трёхразрядные двоичные числа (**A > B**, **A = B** и **A < B** с одним высоким уровнем, равным 1, остальные 0) с выходов предыдущей микросхемы сравнения и, наконец, в последние четыре разряда – значения B3B2B1B0 числа **B**;
- щёлкнуть мышью на изображении логического анализатора **XLA1** и установить в его окне частоту $f_a = 10$ МГц таймера, уровень высокого напряжения $U_m = 4$ В и число импульсов таймера, приходящихся на одно деление, **Clocks/div** = 20;
- запустить программу моделирования компаратора;
- последовательно щёлкая мышью на кнопке **Step** генератора **XWG1**, получить временные диаграммы входных и выходных сигналов на экране анализатора **XLA1**.

В качестве примера на рис. 3.3 показано содержание запрограммированных ячеек памяти генератора бинарного слова **XWG1**, а на рис. 3.4 – временные диаграммы входных и выходных ($Y_>$, $Y_=>$ и $Y_<$) сигналов, характеризующих работу компаратора.

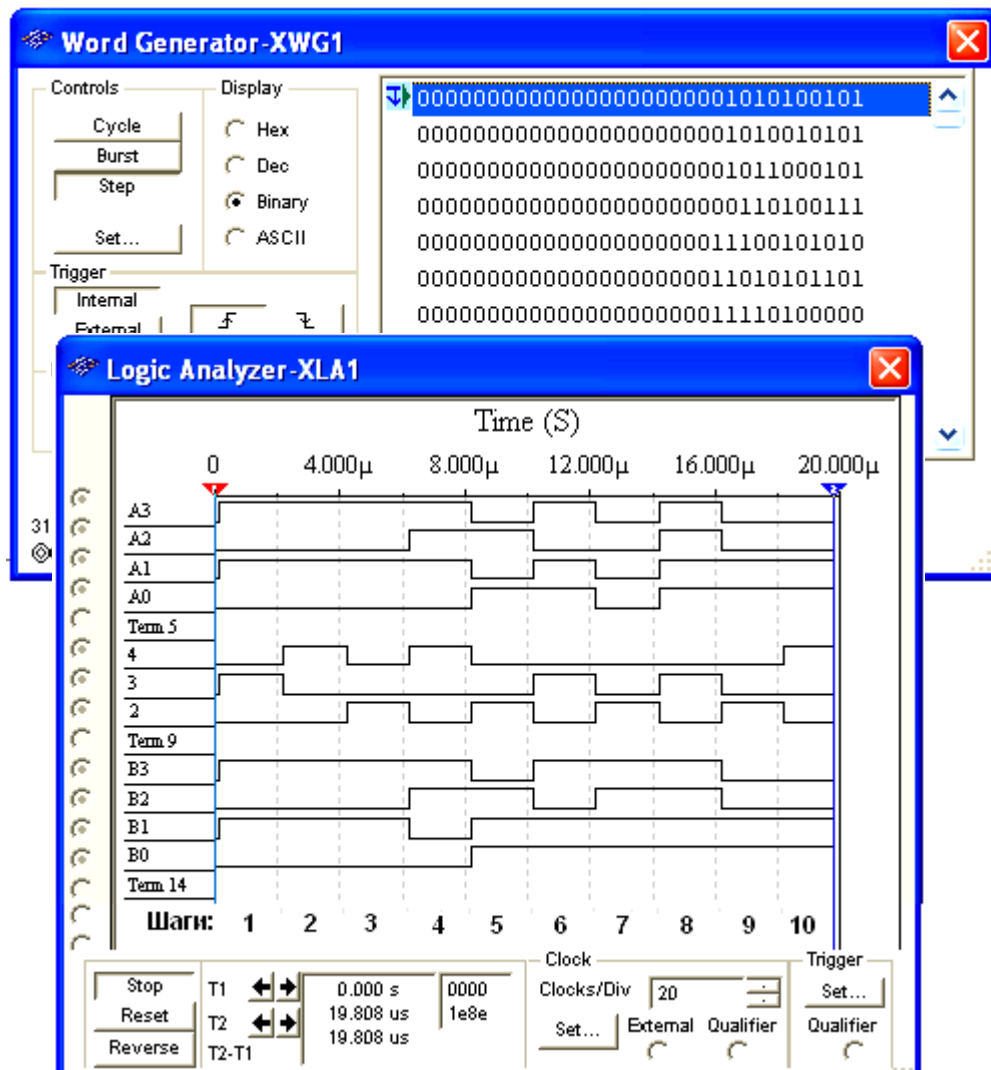


Рис. 3.4

Только при равенстве всех разрядов двоичных чисел, в том числе четырёх младших разрядов (при коде 010 с предыдущей микросхемы) и четырёх старших разрядов: $A = B = 1010$ (см. шаг 1 на рис. 3.4); $A = B = 0101$ (шаг 5) и $A = B = 1111$ (шаг 8) на выходе $Y_=>$ компаратора формируются логические единицы. На втором шаге при $A = B = 1010$ выходной сигнал $Y_> = 1$, так как на компаратор подан код 001 с предыдущей микросхемы, а на третьем шаге выходной сигнал $Y_< = 1$, так как подан код 100. При равенстве четырёх младших разрядов (код 010) на четвёртом шаге $Y_> = 1$, так как число $A = 1110$ больше числа $B = 1100$, а на пятом – сигнал $Y_< = 1$, так как число $A = 0101$ меньше числа $B = 0111$, и т. д.

Задание 3. Скопировать на страницу отчёта диалоговое окно генератора **XWG1** и окно анализатора **XLA1** с временными диаграммами входных и выходных сигналов.

Руководствуясь таблицей истинности (см. табл. 3.1), **дать пояснения** результатам сравнения двух бинарных чисел для всех записанных в ячейки памяти генератора **XWG1** комбинаций бинарных последовательностей.

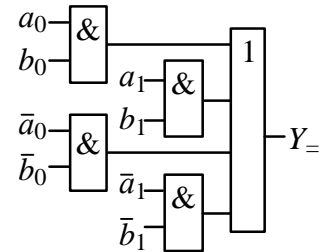
СОДЕРЖАНИЕ ОТЧЕТА

1. Наименование и цель работы.
2. Перечень приборов, использованных в экспериментах, с их краткими характеристиками.
3. Изображение электрической схемы для испытания цифрового компаратора.
4. Копии диалогового окна генератора слова **XWG1** с записанными комбинациями двоичных последовательностей в его ячейки памяти и окно логического анализатора **XLA1** с временными диаграммами входных и выходных сигналов исследуемого компаратора.
5. Выводы по работе.

ТЕСТОВЫЕ ЗАДАНИЯ К РАБОТЕ 3

1. Укажите: а) можно ли установить **факт равенства** двухразрядных бинарных чисел **A** и **B** с помощью приведенного устройства сравнения; б) какой **уровень** сигнала установится на его выходе при равенстве чисел **A** и **B**?

- а) ☐ Да ☐ Нет
б) ☐ 0 ☐ 1



2. Укажите, какую **функцию** выполняет цифровой компаратор?

- ☐ Суммирование по модулю 2 всех разрядов с целью выяснения чётности числа
☐ Сравнение двух бинарных чисел **A** и **B** одинаковой разрядности с целью определения равенства **A = B** или неравенства **A < B** и **A > B**
☐ Хранение и преобразование многоразрядных чисел
☐ Сравнение пилообразного сигнала с образцовым

3. Укажите **логическую функцию**, выражающую равенство i -х разрядов двоичных чисел.

- ☐ $y = a_i b_i + \bar{a}_i \bar{b}_i$ ☐ $y = \overline{a_i b_i}$ ☐ $y = \overline{a_i + b_i}$ ☐ $y = \overline{a_i b_i + a_i \bar{b}_i}$

4. Укажите, к какому **типу** цифровых устройств относят компараторы?

- ☐ К последовательным
☐ К комбинационным

5. Укажите **число активных** логических сигналов, формирующихся на выходе компаратора при сравнении многоразрядных двоичных чисел.

- ☐ Число активных выходных сигналов равно числу разрядов сравниваемых бинарных чисел.
☐ 4
☐ 2
☐ 1

6. Укажите, чем определяется **число входов** цифрового компаратора?

- ☐ Компараторы всегда имеют четыре входа
☐ Число входов зависит от степени декомпозиции сравнивающего устройства и равно числу элементов сравнения одноразрядных слов
☐ Число входов определяется разрядностью сравниваемых бинарных чисел

7. Укажите, можно ли **построить** устройство сравнения требуемой разрядности, используя цифровые компараторы с ограниченной разрядностью (например, четырёхразрядные)?

- ☐ Да ☐ Нет

Лабораторная работа 4 (Lr4)

ТРИГГЕРЫ

ЦЕЛЬ РАБОТЫ

Ознакомление с основными характеристиками и испытание интегральных триггеров RS , D , T и JK .

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И РАСЧЁТНЫЕ ФОРМУЛЫ

Триггер – это устройство последовательностного типа с двумя устойчивыми состояниями равновесия, предназначенное для записи и хранения информации. Под действием входных сигналов триггер может переключаться из одного устойчивого состояния в другое. При этом напряжение на его выходе скачкообразно изменяется с низкого уровня на высокий или наоборот.

По способу записи информации триггеры делят на *асинхронные*, которые переключаются в момент подачи входного сигнала, и *синхронные* (тактируемые), которые переключаются только при подаче синхронизирующих импульсов, а момент переключения связан с определённым уровнем синхросигнала (*статические* триггеры) или с моментом перепада напряжения на тактируемом входе (*динамические* триггеры).

Как правило, триггер имеет два выхода: прямой Q и инверсный \bar{Q} . Число входов зависит от структуры и функций, выполняемых триггером. Например, асинхронные RS -триггеры имеют два входа: вход S установки в *единичное* состояние прямого выхода Q и вход R установки в *нулевое* состояние выхода Q . Синхронные триггеры для занесения в них информации, помимо информационных входов S (J) и R (K), имеют синхронизирующий C или счётный T вход, а триггеры задержки – информационный вход D .

Наибольшее распространение в цифровых устройствах получили триггеры RS , D , T и JK .

1. АСИНХРОННЫЙ И СИНХРОННЫЙ RS -ТРИГГЕРЫ

Простейшим триггером является *асинхронный* RS -триггер, условное графическое изображение которого представлено на рис. 4.1, а, а принцип его работы поясняется таблицей истинности (табл. 4.1). Триггер имеет два отдельных информационных входа: R и S и два выхода: Q и \bar{Q} . Независимым является один (прямой) выход Q , так как инверсный сигнал \bar{Q} можно получить с помощью внешнего инвертора.

Рассмотрим табл. 4.1. Обозначим Q^t сигнал на выходе триггера до поступления сигнала 1 на его вход S . При подаче сигналов $S = 1$ и $R = 0$ триггер переходит в состояние $Q^{t+1} = 1$. При поступлении сигналов $R = 1$ и $S = 0$ на выходе устанавливается $Q^{t+1} = 0$. При отсутствии новых команд состояние триггера не изменяется: триггер сохраняет информацию о последней из поступивших команд. Естественно, что комбинация сигналов $S = 1$ и $R = 1$ относится к

запрещённым, так как при её подаче на входы триггера на его выходе Q^{t+1} устанавливается либо 1, либо 0.

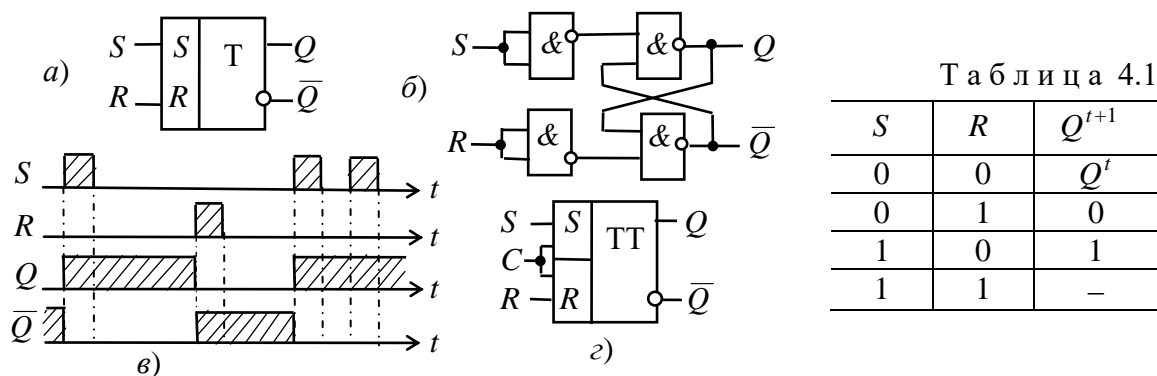


Рис. 32.1

На основании табл. 4.1 запишем аналитическое выражение функционирования RS-триггера:

$$Q^{t+1} = S + Q^t \bar{R}.$$

На рис. 4.1, в изображена временная диаграмма, иллюстрирующая его работу. В момент, когда подаётся сигнал $S = 1$, триггер переходит в состояние $Q = 1$. При отсутствии входных сигналов состояние триггера не изменяется, а в момент подачи сигнала $R = 1$ триггер переключается в состояние $Q = 0$, в котором пребывает до поступления нового единичного сигнала на S-вход.

RS-триггер может быть построен на различных логических элементах. На рис. 31.1, б показана схема реализации RS-триггера на базовых элементах И-НЕ, в которой использована положительная обратная связь (ПОС) с выходов триггера на входы логических элементов. Именно наличие ПОС отличает триггер от ранее рассмотренных комбинационных логических устройств: посредством сигналов ПОС в триггере фиксируется его предшествующее состояние.

Асинхронный RS-триггер можно преобразовать в *синхронный*, если добавить третий синхронизирующий вход C (рис. 4.1, г), соединенный, например, с нижними, предварительно разделёнными, входами двух левых элементов И-НЕ (см. рис. 4.1, б).

Вход C обеспечивает функционирование RS-триггера по закону

$$Q^{t+1} = Q^t (\bar{C} + \bar{R}) + CS.$$

Переключение синхронного RS-триггера в состояние $Q = 1$ происходит при $S = 1$ (или в состояние $Q = 0$ при $R = 1$) в момент прихода синхроимпульса C. При $C = 0$ информация с S- и R-входов на триггер не передается.

2. T-ТРИГГЕР

Триггер со счетным запуском (*T-триггер*) должен переключаться каждым импульсом, подаваемым на единственный счётный вход T (рис. 4.2, а). Функционирование T-триггера определяется уравнением

$$Q^{t+1} = Q^t \bar{T} + \bar{Q}^t T.$$

Он может быть реализован, например, на базе двух синхронных RS -триггеров (рис. 4.2, б). С появлением фронта тактового импульса триггер T_1 первой ступени переключается в состояние, противоположное состоянию триггера T_2 . Но это не вызывает изменение сигналов на выходах Q и \bar{Q} , так как за счёт инвертора на тактовый вход C триггера T_2 в данный момент подан логический 0. Только на срезе счетного импульса на входе T_1 переключится триггер T_2 и произойдёт изменение сигналов на выходах Q и \bar{Q} , а также на S - и R -входах первой ступени.

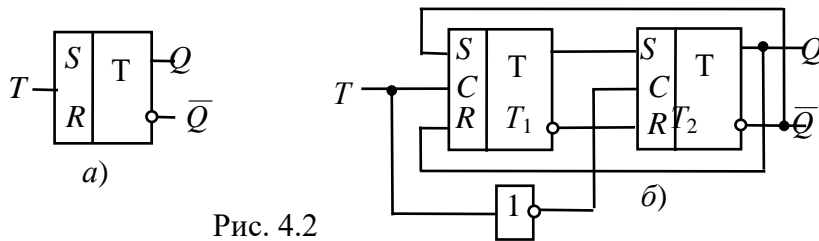


Рис. 4.2

3. D-ТРИГГЕР

Триггер задержки (D -триггер) может быть только синхронным, так как имеет один информационный D -вход, информация с которого переписывается на выход триггера только по тактовому сигналу, подаваемому на C -вход. Условное изображение D -триггера приведено на рис. 4.3, а. Реализовать его можно на различных логических элементах, в том числе, на основе синхронного RS -триггера, дополненного инвертором (рис. 4.3, б). Из анализа табл. 4.2 переключательной функции D -триггера

$$Q^{t+1} = \bar{C}^t Q^t + C^t Q^t$$

следует, что при отсутствии синхроимпульса ($C = 0$) состояние триггера остается неизменным. При условии же $C = 1$ триггер передает на выход сигнал, поступивший на его вход D в предыдущем такте, т. е. выходной сигнал Q^{t+1} изменяется с задержкой на один период импульсов синхронизации.

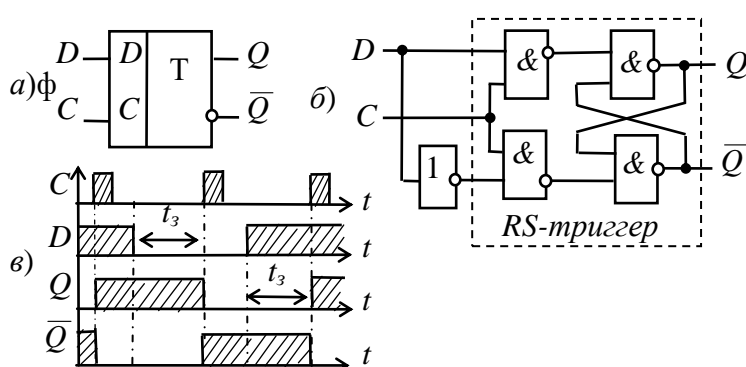


Рис. 4.3

Таблица 4.2

| C^t | D^t | Q^t | Q^{t+1} |
|-------|-------|-------|-----------|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

Из анализа временной диаграммы D -триггера (рис. 4.3, в) также следует, что выходной сигнал Q триггера повторяет состояние D -входа с поступлением очередного тактового импульса на вход C с задержкой t_3 относительно сменившегося логического состояния на D -входе.

4. JK-ТРИГГЕР

JK-триггеры обычно выполняют тактируемыми. JK-триггер имеет информационные входы J и K , которые по своему воздействию на устройство аналогичны входам S и R синхронного RS-триггера: при $J = 1$ и $K = 0$ триггер по тактовому импульсу C устанавливается в состояние $Q = 1$; при $J = 0$ и $K = 1$ – переключается в состояние $Q = 0$, а при $J = 0$ и $K = 0$ – хранит ранее принятую информацию.

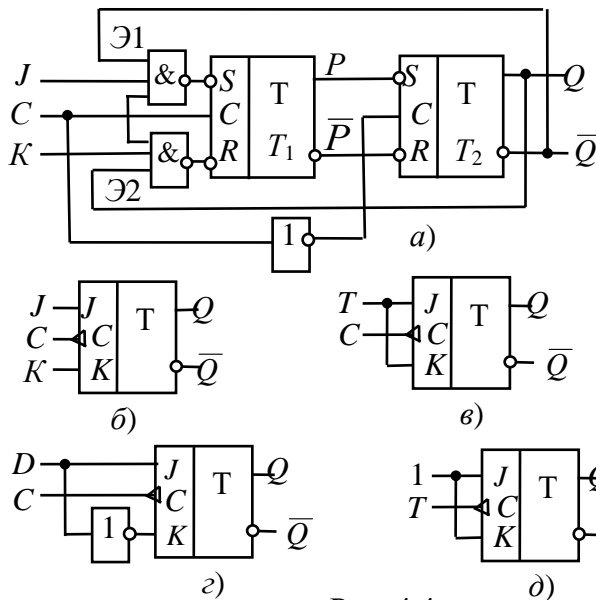


Рис. 4.4

Т а б л и ц а 4.3

| J^t | K^t | Q^t | Q^{t+1} |
|-------|-------|-------|-----------|
| 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 |

В отличие от синхронного RS-триггера одновременное присутствие логических единиц на информационных входах не является для JK-триггера запрещенной комбинацией; при $J = 1$ и $K = 1$ триггер работает в счетном режиме, т. е. переключается каждым тактовым импульсом на входе C .

На рис. 4.4, а изображена одна из функциональных схем JK-триггера. Она отличается от схемы Т-триггера (см. рис. 4.2, б) двумя трёхвходовыми элементами И-НЕ Э1 и Э2 входной логики первой ступени JK-триггера. Переключающий вход C – динамический (рис. 4.4, б): переключение JK-триггера происходит в момент перепада синхроимпульса с уровня $C = 1$ на уровень $C = 0$, т. е. при срезе.

При $J = 0$ и $K = 0$ на выходе элементов Э1 и Э2 устанавливаются логические единицы, которые для триггеров с инверсными входами являются пассивными сигналами: триггер T_1 и, следовательно, JK-триггер в целом сохраняют прежнее состояние (см. рис. 4.4, а). Логическая 1 на одном из входов элемента И-НЕ не определяет 1 на его выходе и комбинация $J = 1$, $K = 1$ никак не влияет на входную логику первой ступени, поэтому схемы Т- и JK-триггеров (см. рис. 4.2, б и рис. 4.4, а) принципиально не отличаются: оба работают в счетном режиме.

Только при комбинации сигналов $J = 1$, $C = 1$ и $\bar{Q} = 1$ на входе элемента Э1 триггер T_1 переключится в состояние $P = 1$. Аналогично логический 0 будет на выходе элемента Э2, когда $K = 1$, $C = 1$ и $Q = 1$.

Таким образом, комбинация $J = 1, K = 0$ обуславливает по тактовому импульсу $C = 1$ переключение JK -триггера в целом в состояние $Q = 1$, а комбинация $J = 0, K = 1$ – в состояние $Q = 0$.

Из анализа табл. 4.3 переключательной функции JK -триггера

$$Q^{t+1} = \bar{K}^t Q^t + J^t \bar{Q}^t$$

следует, что состояние триггера определяется не только уровнями сигналов на информационных входах J и K , но и состоянием Q^t , в котором ранее находился JK -триггер. Так, при комбинации $J = 0, K = 0$ триггер сохраняет предыдущее состояние ($Q^{t+1} = Q^t$); комбинация $J = 1, K = 1$ приводит к тому, что тактовым импульсом триггер переключается в состояние, противоположное предыдущему: $Q^{t+1} = \bar{Q}^t$. Комбинации $J = 1, K = 0$ и $J = 0, K = 1$ дают разрешение триггеру переключиться соответственно в состояния $Q = 1$ и $Q = 0$.

На основе JK -триггера (рис. 4.4, б) могут быть выполнены синхронный (рис. 4.4, в) и асинхронный (рис. 4.4, г) T -триггеры, D -триггер (рис. 4.4, д) и синхронный RS -триггер (рис. 4.4, е).

При проектировании сложных логических схем (микросхем) необходимы триггеры различных типов, которые можно было бы выполнить на основе одного универсального триггера и использовать его в разных режимах работы и модификациях. В интегральной схемотехнике наибольшее распространение получили D - и JK -триггеры.

УЧЕБНЫЕ ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ИХ ВЫПОЛНЕНИЮ

Задание 1. Запустить лабораторный комплекс Labworks и среду MS10 (щёлкнув мышью на команде **Эксперимент** меню комплекса Labworks). Открыть файл **4.5.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания *асинхронного RS-триггера* (рис. 4.5) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 4.5) на страницу отчёта.

Схема (рис. 4.5) собрана на четырёх логических элементах И-НЕ (**NAND**). На входы S и R элементов **NAND1** и **NAND2** через ключи **1** и **2** подаются логические сигналы 1 или 0 от источника прямоугольных импульсов **E1** с амплитудой 5 В. К выходам Q и \bar{Q} элементов **NAND3** и **NAND4**, т. е. к выходам триггера, как и к его входам S и R , подключены пробники **X1, X2, X3** и **X4** с пороговым напряжением 5 В.

Воспользовавшись порядком засвечивания разноцветных пробников и задавая коды (00, 01, 10) состояния ключей **1** и **2** (входных сигналов), **составить** таблицу истинности RS -триггера. Например, сформировав с помощью ключей сигналы $S = 1$ и $R = 0$ и подав их на вход триггера, получите на его выходе сигналы $Q = 1$ и $\bar{Q} = 0$ (см. рис. 4.5). Убедитесь, что при запрещённом коде 11

входных сигналов, на выходе *RS*-триггера могут засветиться оба пробника, или оба не светятся.

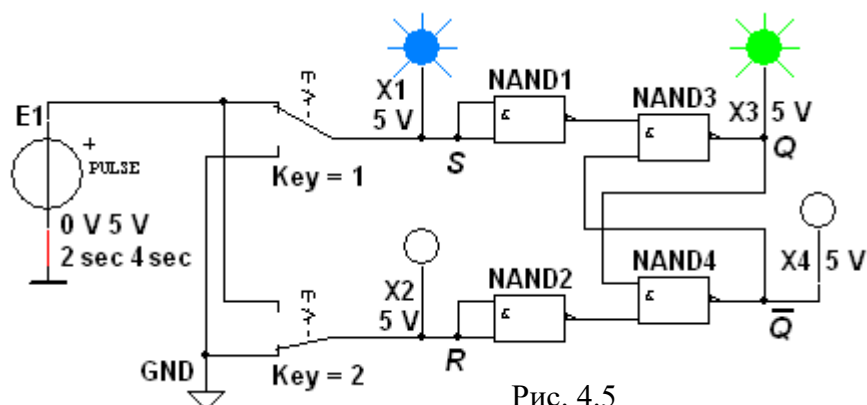


Рис. 4.5

Задание 2. Подключить к входам триггера логический генератор (генератор слова) **XWG1** (рис. 4.6), запрограммировав его первые три ячейки кодами 00, 10 и 01 и соединив входы и выходы триггера с входами логического анализатора **XLA2**.

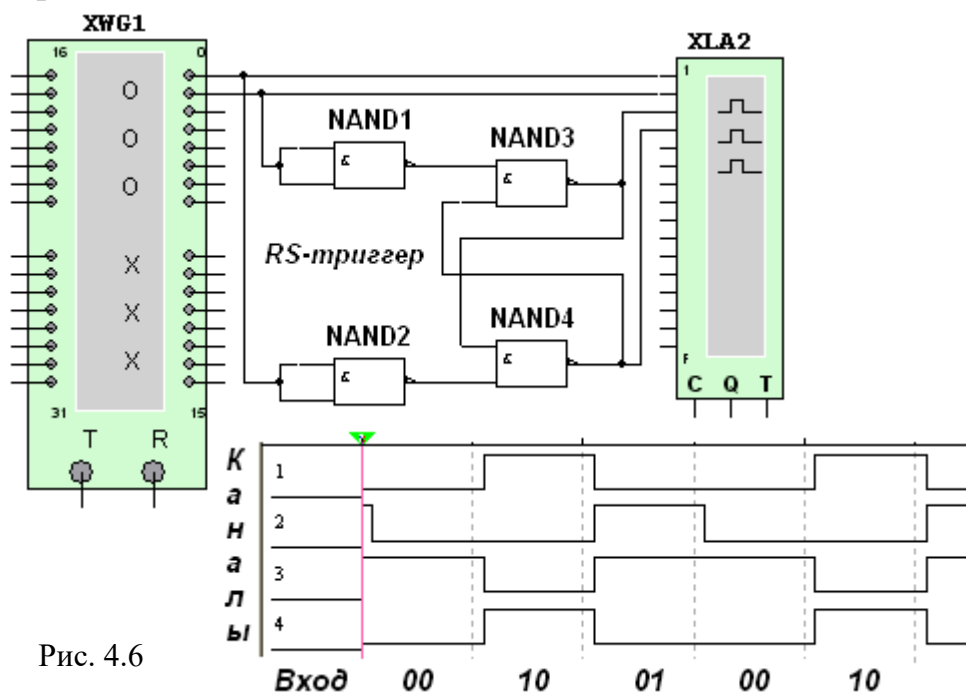


Рис. 4.6

В диалоговом окне генератора слова **XWG1** задать частоту $f_z = 10$ кГц и два цикла моделирования сигналов (в режиме **Burst**), а в окне анализатора **XLA2** – частоту $f_a = 0,1$ МГц таймера, уровень высокого напряжения $U_m = 5$ В, число импульсов **Clocks/div** = 8 таймера, приходящихся на одно деление.

Получить на экране анализатора **XLA2** временную диаграмму состояний *RS*-триггера (см. рис. 4.6, внизу). **Скопировать** схему испытания и временную диаграмму состояния *RS*-триггера на страницу отчёта.

Задание 3. Открыть файл **4.7.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания *триггеров* **JK**, **T** и **D** (рис. 4.7) и **установить** в диалоговых окнах

компонентов их параметры или режимы работы. **Скопировать** схему (рис. 4.7) на страницу отчёта.

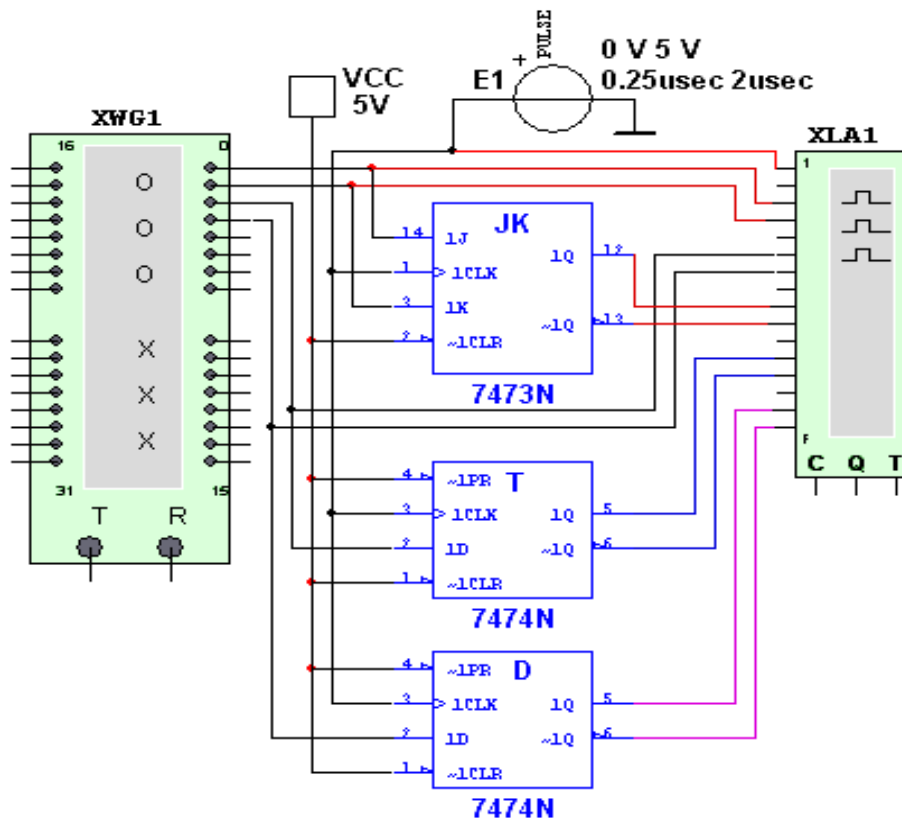


Рис. 4.7

В схему (рис. 4.7) включены: генератор **XWG1** (частота $f_2 = 500$ кГц); логический анализатор **XLA1**; триггеры в интегральном исполнении: универсальный **JK**, счётный **T** и задержки **D**.

На $\overline{1CLR}$ - и $\overline{1PK}$ -входы триггеров подаётся постоянное напряжение 5 В (имитирующее сигнал 1) источника **VCC**, а на $\overline{1C}$ -входы триггеров и на вход 20 анализатора **XLA1** поступают тактовые импульсы с амплитудой 5 В и частотой 500 кГц, сформированные генератором **E1**.

С выходов 1 и 2 генератора **XWG1** сигналы подаются на управляющие входы **1J** и **1K** **JK**-триггера, с выхода 3 – на вход **1D** **T**-триггера, а с выхода 4 – на вход **1D** **D**-триггера.

Для формирования выходных сигналов генератор **XWG1** нужно **запрограммировать**, т. е. ввести в ячейки памяти кодовые комбинации из единиц и нулей согласно варианту (табл. 4.4).

В качестве примера введём в первые восемь ячеек памяти генератора четырехразрядные кодовые комбинации (см. рис. 4.8, а):

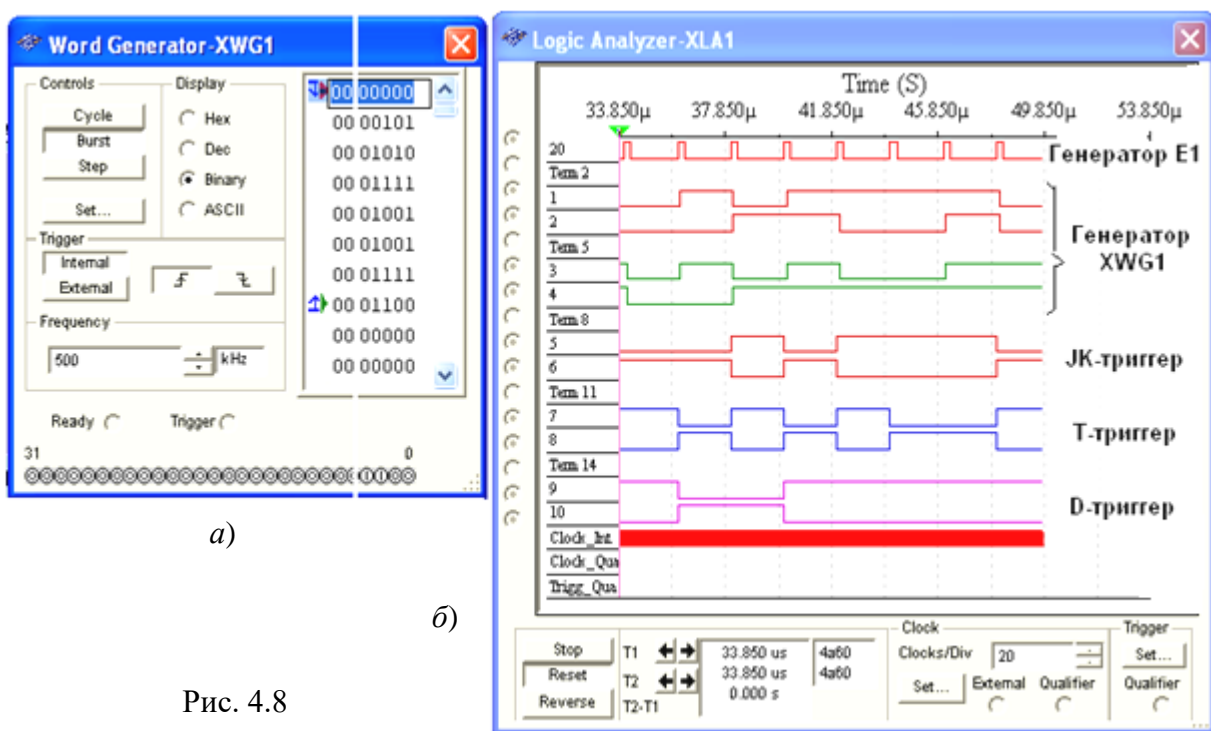
0000, 0101, 1010, 1111, 1001, 1001, 1111, 1100.

При моделировании генератор последовательно и циклично выводит содержимое каждой ячейки памяти (от начальной до конечной) на выходы 1, 2, 3 и 4, формируя на них следующие коды сигналов: 01011110, 00110010, 01010011 и 00111111 (см. сигналы на каналах 1, 2, 3 и 4 логического анализатора

XLA1 (рис. 4.8, б)). Перед моделированием **выделите** в окне генератора **XWG1** ячейку с адресом 0 начала счёта и вывода сигналов.

Т а б л и ц а 4.4

| Вариант | Содержимое ячеек памяти генератора слова XWG1 |
|-----------------------|--|
| 1, 6, 11, 16, 21, 26 | 0000, 1010, 1111, 1001, 1001, 1101, 1100, 0000 |
| 2, 7, 12, 17, 22, 27 | 0000, 1100, 1010, 1011, 1001, 1111, 1110, 0000 |
| 3, 8, 13, 18, 23, 28 | 0000, 1010, 1011, 1001, 1001, 1111, 1101, 0000 |
| 4, 9, 14, 19, 24, 29 | 0000, 1111, 1101, 1001, 1011, 1011, 1100, 0000 |
| 5, 10, 15, 20, 25, 30 | 0000, 1011, 1101, 1001, 1100, 1111, 1010, 0000 |



Провести моделирование работы триггеров в режимах **Step** или **Burst** генератора **XWG1**, **скопировать** в отчёт временные диаграммы, **составить** и **заполнить** таблицы истинности работы триггеров **JK**, **T** и **D** при заданном в табл. 4.4 варианте входных кодовых комбинаций. В частности, **описать** состояния **JK**-триггера с приходом тактового сигнала $C = 1$, когда сигналы $J = 1$ и $K = 1$, а $Q = 0$ или $Q = 1$.

Примечание. Таблицы истинности для рассмотренных библиотечных триггеров можно вызвать нажатием клавиши помощи **F1** после выделения на схеме триггера.

СОДЕРЖАНИЕ ОТЧЁТА

1. Наименование и цель работы.
2. Перечень приборов, использованных в экспериментах, с их краткими характеристиками.

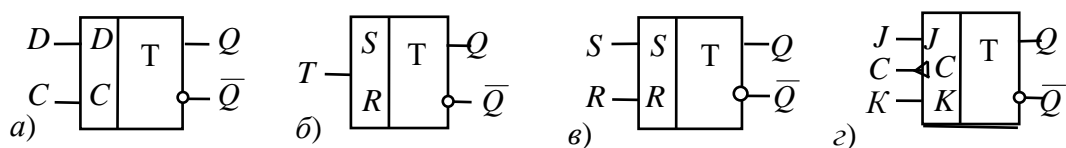
- Изображения электрических схем для испытания триггеров RS , JK , T и D с помощью логических пробников и логического анализатора **XLA1**.
- Копии временных диаграмм и таблицы истинности, отображающие работу исследуемых триггеров.
- Выводы по работе.

ТЕСТОВЫЕ ЗАДАНИЯ К РАБОТЕ 4

- Укажите, какая **комбинация** логических сигналов является запрещённой для асинхронного RS -триггера?

☐ 01 ☐ 11 ☐ 10 ☐ 00

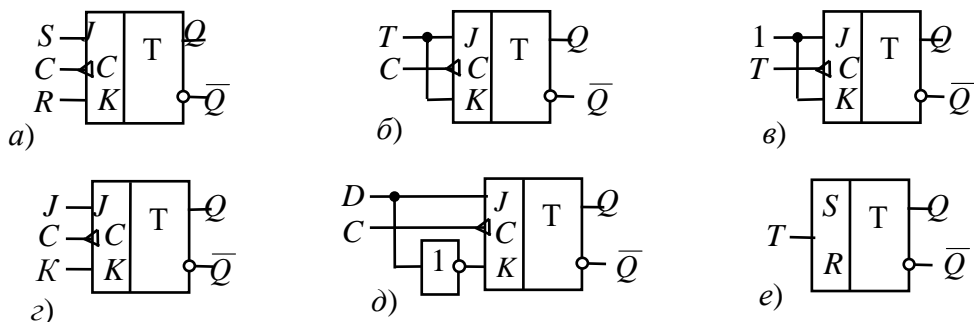
- Укажите **условное графическое обозначение**:



- JK -триггера: ☐ а) ☐ б) ☐ в) ☐ г)

- RS -триггера: ☐ а) ☐ б) ☐ в) ☐ г)

- Укажите **условное графическое обозначение**:



- Синхронного T -триггера, выполненного на основе JK -триггера:

а) б) в) г) д) е)
☐ ☐ ☐ ☐ ☐ ☐

- D -триггера, выполненного на основе JK -триггера:

а) б) в) г) д) е)
☐ ☐ ☐ ☐ ☐ ☐

- Укажите, нашли ли широкое применение **асинхронные** D -триггеры?

☐ Да ☐ Нет

- Укажите, как **функционирует** JK -триггер при комбинации $J = 1$, $K = 1$ на входе?

- ☐ Триггер находится в режиме хранения
- ☐ Триггер работает в счётном режиме
- ☐ Такая комбинация сигналов на входе является запрещённой

- Укажите **время запаздывания** выходного сигнала по отношению к моменту подачи на C -вход D -триггера синхроимпульса при тактовой частоте $f = 10$ кГц ($D^t = 1$, $Q^t = 0$).

☐ 1 с ☐ 0,1 с ☐ 10 мс ☐ 0,1 мс

7. Укажите значение **сигнала на выходе JK-триггера** при комбинации $J = 1, K = 0$ на входе и $Q = 1$ после окончания действия синхроимпульса.

☐ 0 ☐ 1 ☐ Неопределённость: 0 или 1

8. Укажите **аналитическое выражение**, описывающее работу:

а) $Q^{t+1} = Q^t \bar{T} + \bar{Q}^t T$; б) $Q^{t+1} = S + Q^t \bar{R}$;
 в) $Q^{t+1} = \bar{C}^t Q^t + C^t \bar{Q}^t$; г) $Q^{t+1} = \bar{K}^t Q^t + J^t \bar{Q}^t$.

1. RS-триггера: ☐ а) ☐ б) ☐ в) ☐ г)

2. JK- триггера: ☐ а) ☐ б) ☐ в) ☐ г)

3. T-триггера: ☐ а) ☐ б) ☐ в) ☐ г)

4. D-триггера: ☐ а) ☐ б) ☐ в) ☐ г)

9. Укажите, чем отличается **динамическое управление** триггерами от статического управления?

☐ Принципиальных отличий нет: сигналы, поступающие на информационные входы всех модификаций триггеров, действуют в момент их поступления

☐ У триггеров с динамическим управлением сигналы на информационных входах должны оставаться неизменными на всём интервале действия активного логического сигнала синхронизации ($C = 1$)

☐ При динамическом управлении запоминание сигналов, действующих на информационных входах триггера, происходит в момент изменения значения сигнала на входе синхронизации

☐ У триггеров с динамическим управлением отсутствуют прямые или инверсные входы, реагирующие на перепады сигналов на входах

10. Укажите **уровни напряжения** интегральных микросхем триггеров серии ТТЛ, принимаемые за логическую 1 и логический 0 при напряжении питания $U_n = 5$ В.

☐ $2,4 \text{ В} < U^1 < 5 \text{ В}; 0 < U^0 < 0,4 \text{ В}$ ☐ $4,0 \text{ В} < U^1 < 5 \text{ В}; 0 < U^0 < 2,4 \text{ В}$

☐ $3,5 \text{ В} < U^1 < 5 \text{ В}; 0 < U^0 < 0,2 \text{ В}$ ☐ $2,4 \text{ В} < U^1 < 5 \text{ В}; 0 < U^0 < 1,4 \text{ В}$

11. Укажите, к какому **типу** триггеров относят T-триггеры?

☐ К асинхронным

☐ К синхронным

Лабораторная работа 5 (Lr5)

РЕГИСТРЫ

ЦЕЛЬ РАБОТЫ

Ознакомление с устройством и функционированием регистров и регистровой памяти; испытание интегрального универсального регистра сдвига.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И РАСЧЁТНЫЕ ФОРМУЛЫ

1. КЛАССИФИКАЦИЯ РЕГИСТРОВ

Регистр – это последовательностное устройство, предназначенное для записи, хранения и (или) сдвига информации, которая поступает и хранится в регистре в виде n -разрядных двоичных чисел. В общем случае регистр может выдавать информацию в последовательной или параллельной форме, *преобразовывать* прямой код числа в обратный (когда единицы заменяются нулями, а нули – единицами), и наоборот, а также выполнять логическое сложение и логическое умножение двоичных чисел.

В зависимости от способа ввода и вывода разрядов числа различают регистры параллельные, последовательные и параллельно-последовательные. В *параллельном* регистре ввод и вывод всех разрядов кодового числа осуществляется одновременно, в *последовательном* – разряды числа вводятся и выводятся последовательно, а в *параллельно-последовательном* регистре ввод числа производится в параллельной форме, а вывод – в последовательной, и наоборот. Преобразование параллельного кода в последовательный и наоборот – очень актуальная задача, так как передача цифровой информации в сетях передачи данных осуществляется в последовательном коде, а обработка её в микропроцессорах вычислительных устройств – в параллельном.

Регистр, в котором можно осуществить сдвиг числа, называют *сдвигающим* (сдвиговым), причем сдвиг может быть или в одну сторону (в сторону младшего разряда – *прямой* (правый) *сдвиг*, или в сторону старшего разряда – *обратный* (левый) *сдвиг*, или в обе стороны (*реверсивный сдвигающий* регистр). В этом смысле последовательный и параллельно-последовательный регистры относят к сдвиговым.

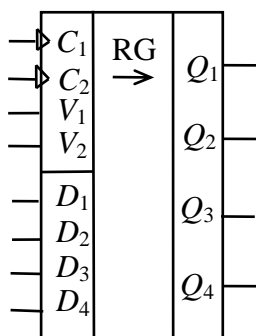


Рис. 5.1

Отечественная промышленность выпускает многие типы регистров в виде микросхем. В качестве примера на рис. 5.1 приведено изображение четырехразрядного регистра (микрохема серии К155). При $V_2 = 0$ разряды числа вводят последовательно в регистр через вход V_1 ; синхроимпульсы, поступающие на вход C_1 , обеспечивают сдвиг вправо разрядов числа; регистр работает как сдвигающий. В микросхеме (см. рис. 5.1) предусмотрен также параллельный ввод всех разрядов числа по синхроимпульсу на входе C_2 с

входов D_1, \dots, D_4 при $V_2 = 1$. В данном случае регистр работает как параллельный.

Если выводы последнего триггера сдвигающего регистра соединить с входами первого, то получится *кольцевой* регистр сдвига, называемый *кольцевым счётчиком*. Его коэффициент пересчёта равен числу разрядов n : единица, записанная в один из разрядов, периодически появляется на выходе счётчика после того, как пройдут n сдвигающих синхроимпульсов.

2. ПАРАЛЛЕЛЬНЫЙ РЕГИСТР НА RS-ТРИГГЕРАХ

Любой регистр состоит из связанных между собой триггеров с динамическим или статическим управлением и логических элементов, причем количество триггеров равно количеству разрядов в записываемом числе. Синтез регистра сводится к выбору типа триггеров и логических элементов И, НЕ, ИЛИ для реализации заданных операций.

Рассмотрим работу параллельного регистра на RS -триггерах (рис. 5.2). Ввод (запись) числа осуществляется в два такта. Во избежание ошибочной записи числа $x_1x_2...x_n$ в первом такте все триггеры регистра обнуляются. Для этого на шину "0" подается логический 0. Во втором такте по сигналу 1 на шине "П" ("Приём") через конъюнкторы одновременно записывается в соответствующие разряды регистра двоичное число $x_1x_2...x_n$. Вывод (считывание) числа $y_1y_2...y_n$ в прямом коде происходит по сигналу 1 на шине " B_{np} ", а в обратном – по

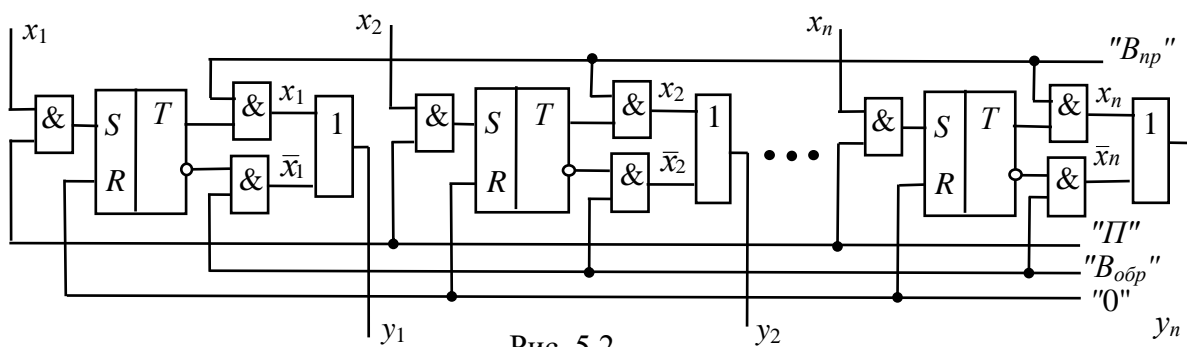


Рис. 5.2

сигналу 1 на шине " $B_{обp}$ ".

Объединив в одной микросхеме несколько регистров и добавив на входе дешифратор DCW , а на выходе мультиплексор MS , получают *регистровую (сверхоперативную) память* (рис. 5.3). Входы D_i четырёх или восьми регистров, как правило, 4-разрядных, подключают к общей входной шине данных DIN . Вход загрузки требуемого регистра выбирается дешифратором записи DCW на основании поступающего на его вход адреса записи WA , т. е. кода номера загружаемого регистра. *Запись* данных, присутствующих на шине DIN , происходит в момент поступления сигнала разрешения записи WE .

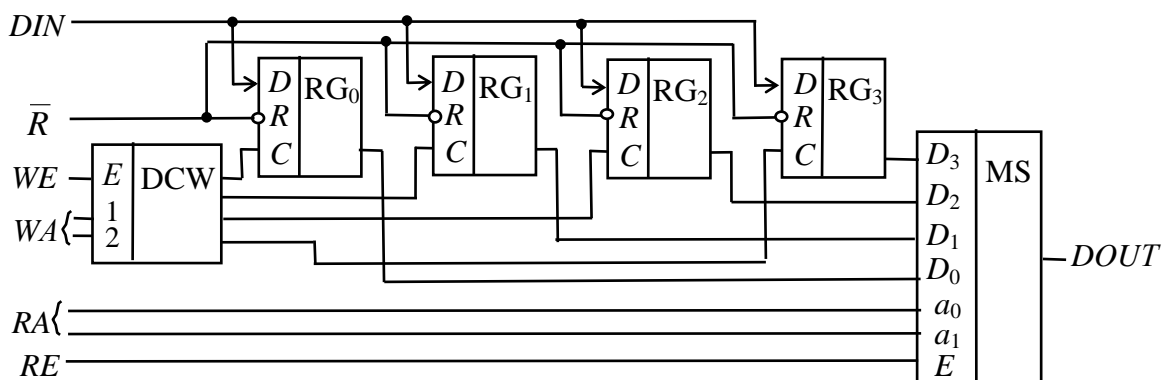


Рис. 5.3

Выходы регистров мультиплексором MS подключаются к выходной шине *DOUT*. Номер регистра, с которого происходит чтение, определяется посредством кода адреса чтения *RA*. Разрешение выдачи данных в шину *DOUT* происходит по сигналу *RE*. Поскольку дешифрация адреса записи и адреса чтения производится двумя независимыми узлами, имеющими автономные адресные входы *WA* и *RA*, в регистровую память можно одновременно записывать бинарное число в один из регистров и считывать число из другого. Описанная структура использована в кристаллах отечественных микросхем К155РП1, ИР11 и ИР12 серий К561 и К564.

Микросхемы регистровой памяти легко наращиваются по разрядности и допускают наращивание по числу регистров. Они разработаны для построения блоков *регистров общего назначения* (РОН), предназначенных для временного хранения исходных данных и промежуточных результатов расчёта в микропроцессорах.

УЧЕБНЫЕ ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ИХ ВЫПОЛНЕНИЮ

Задание 1. Запустить лабораторный комплекс Labworks и среду MS10 (щёлкнув мышью на команде **Эксперимент** меню комплекса Labworks). Открыть файл **5.4.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания *универсального регистра сдвига* (рис. 5.4) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 5.4)

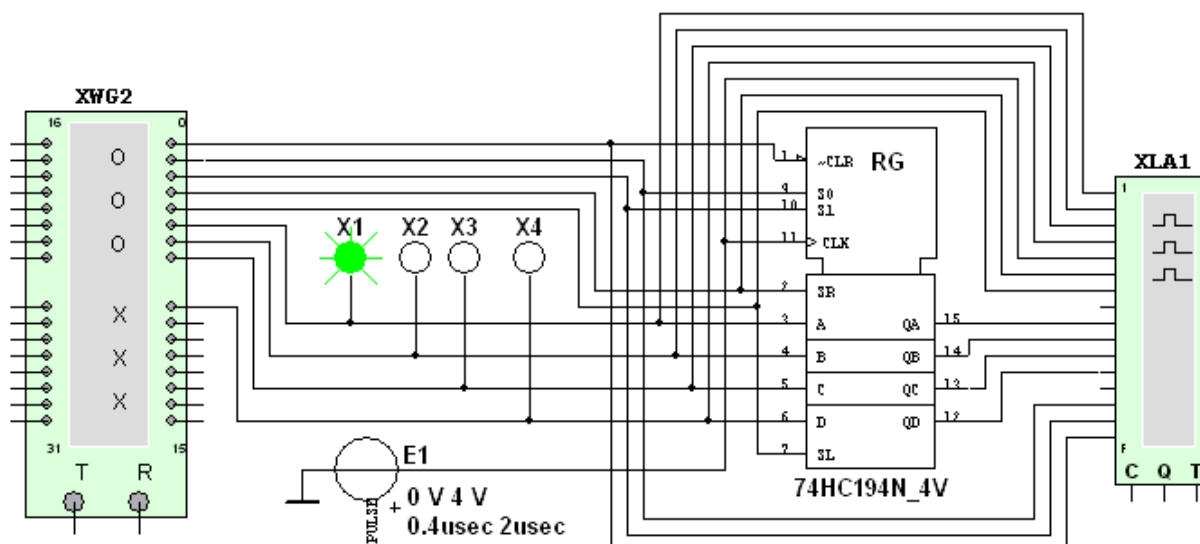


Рис. 5.4

на страницу отчёта.

Универсальный 4-разрядный регистр сдвига **74HC194N_4V** (отечественные аналоги-микросхемы К230ИР2, КМ155ИР1, К176ИР3) способен сдвигать информацию и вправо, и влево, возможна как параллельная, так и последовательная запись данных. Регистр имеет параллельные входы (**A, B, C, D**), параллельные выходы (**QA, QB, QC, QD**), последовательные входы (**SR, SL**),

цепь прямой очистки регистра по входу $\overline{\text{CLR}}$ и управляющие входы (**S0** и **S1**) – входы задания режима:

- **S0** = 1, **S1** = 1 – запись данных в регистр по входам **A**, **B**, **C**, **D**;
- **S0** = 1, **S1** = 0 – сдвиг данных влево в направлении от **QA** к **QD**;
- **S0** = 0, **S1** = 1 – сдвиг данных вправо в направлении от **QD** к **QA**;
- **S0** = 0, **S1** = 0 – входы регистра недоступны (блокировка).

Задание 2. Составить план исследования параллельного регистра сдвига, заполнив ячейки памяти генератора слова **XWG1** на основе правил функционирования регистра **74HC194_4V**, отражённых в табл. 5.1.

Т а б л и ц а 5.1

| Входы | | | | | | | | | | Выходы | | | |
|-------------------------|-------------------------|-------|----|--------------|----|-------------------|---|---|---|-----------------|-----------------|-----------------|-----------------|
| Сброс | Старт | Режим | | Послед. вход | | Параллельный вход | | | | | | | |
| $\overline{\text{CLR}}$ | $\overline{\text{CLX}}$ | S0 | S1 | SR | SL | A | B | C | D | QA | QB | QC | QD |
| 0 | x | x | x | x | x | x | x | x | x | 0 | 0 | 0 | 0 |
| 1 | 0 | x | x | x | x | x | x | x | x | QA ₀ | QB ₀ | QC ₀ | QD ₀ |
| 1 | ↑ | 1 | 1 | x | x | A | B | C | D | A | B | C | D |
| 1 | ↑ | 1 | 0 | 1 | x | x | x | x | x | 1 | QA _n | QB _n | QC _n |
| 1 | ↑ | 1 | 0 | 0 | x | x | x | x | x | 0 | QA _n | QB _n | QC _n |
| 1 | ↑ | 0 | 1 | x | 1 | x | x | x | x | QB _n | QC _n | QD _n | 1 |
| 1 | ↑ | 0 | 1 | x | 0 | x | x | x | x | QB _n | QC _n | QD _n | 0 |
| 1 | x | 0 | 0 | x | x | x | x | x | x | QA ₀ | QB ₀ | QC ₀ | QD ₀ |

Примечание. 0 – низкий уровень; 1 – высокий уровень; x – любое состояние; ↑ - положительный перепад (с низкого уровня на высокий); QA₀, QB₀, QC₀, QD₀ – стационарные уровни A, B, C, D до установки указанных состояний на входах; QA_n, QB_n, QC_n, QD_n – соответственно уровни A, B, C, D перед началом прохождения фронта самого последнего тактового импульса.

Запустить программу моделирования параллельного регистра, **скопировать** в отчёт программу (см. рис. 5.5) и временные диаграммы сигналов на входах и выходах регистра (см. рис. 5.6).

Ввод (запись) и вывод (считывание) информации производится параллельным кодом. Ввод обеспечивается тактовым импульсом, с приходом очередного тактового импульса записанная информация обновляется. Считывание информации происходит в прямом коде в интервале между синхроимпульсами, когда триггеры находятся в режиме хранения.

Руководствуясь схемой соединения генератора **XWG1** с регистром (см. рис. 5.4), при записи чисел в ячейки памяти генератора в младший разряд 9-разрядных чисел нужно заносить значение сигнала $\overline{\text{CLR}}$: логический 0 для очистки регистра или логическая 1 – разрешение записи числа, сдвига данных и др.; в следующие два разряда – значения (1 или 0) сигналов **S0** и **S1**, определяющих режим работы регистра; в два следующих – вводить значения сигналов **SR** и **SL**, определяющих направление сдвига записанной информации в направлении от **QA** к **QB**, **QC**, а затем к **QD** после каждого положительного перепада импульса на тактовом входе $\overline{\text{CLK}}$ или наоборот от **QD** к **QA**. В

старшие разряды нужно занести произвольные (или по указанию преподавателя) значения 4-разрядных чисел **DCBA**, которые передаются на соответствующие выходы.

В качестве примера на рис. 5.5 приведена запись 9-разрядных кодовых комбинаций в 15 ячеек памяти генератора **XWG1**, а на рис. 5.6 – реализация программы моделирования параллельного регистра в виде временных диаграмм сигналов (выводимых в окне анализатора **XLA1**) на его входах и выходах при шаговом (**Step**) режиме работы генератора **XWG1**.

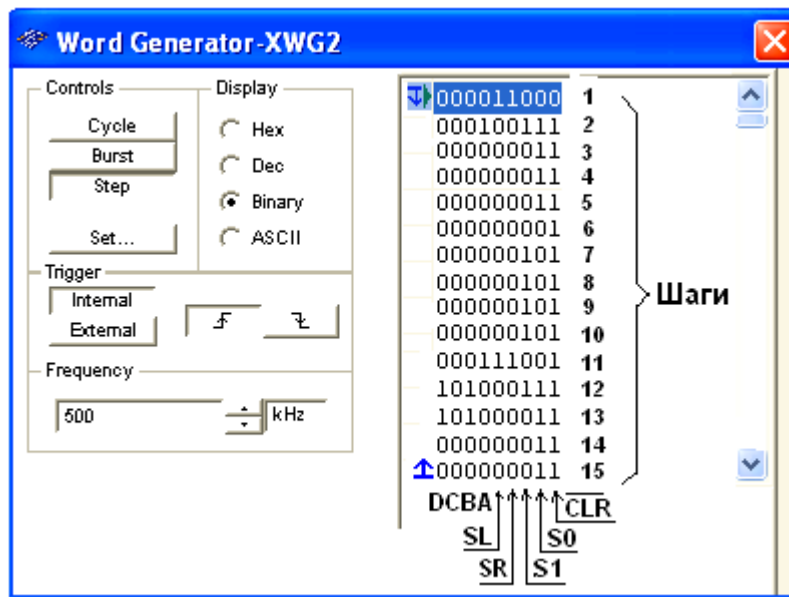


Рис. 5.5

На первом шаге (первом такте работы генератора и регистра) при подаче сигнала $\overline{\text{CLR}} = 0$ (см. первую строку табл. 5.1 и рис. 5.6), на всех входах и выходах регистра установились нулевые значения. На втором шаге при $\overline{\text{CLR}} = 1$, $\text{SR} = 0$, $\text{SL} = 0$ (разрешение записи числа в регистр), $\text{S0} = 1$ и $\text{S1} = 1$ (запрещение сдвига данных во время синхронной параллельной записи числа в регистр, см. третью строку табл. 5.1) происходит загрузка 4-разрядного двоичного числа **DCBA** = 0001 в регистр.

При задании направления сдвига данных влево ($S0 = 1$ и $S1 = 0$, такт или шаг 3) сигнал 0001 выводится на выходы: $QD = 0$, $QC = 0$, $QB = 0$ и $QA = 1$. С приходом очередного тактового импульса (шаги 4, 5 и 6) происходит перезапись (сдвиг) содержимого триггера каждого разряда в соседний разряд (от разряда **A** к разряду **D**) без изменения порядка следования единиц и нулей. По окончании шестого тактового импульса на выходе устанавливается число 1000 (см. рис. 5.6). Если выполнить ещё один шаг при $S0 = 1$ и $S1 = 0$, то занесенная в регистр информация будет полностью из него выведена. Если при работе регистра в режиме сдвига влево (см. шаги 3, ..., 6 на рис. 5.6) в ячейки памяти генератора внести $SL = 1$, то сигнал 1 будет формироваться на выходе **QA** и сдвигаться влево от **QA** к **QD** при каждом тактовом импульсе. В результате, после шестого импульса на выходе, установится сигнал 1111.

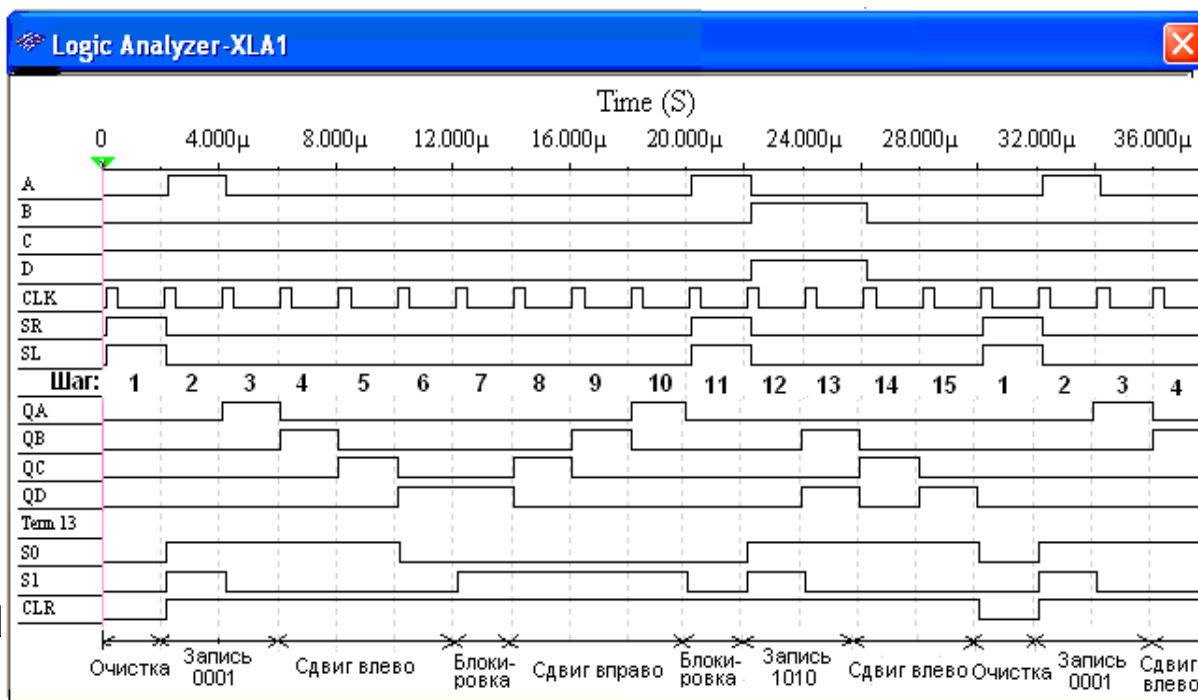


Рис. 5.6

Режим блокировки реализуется при подаче на оба управляющих входа сигналов низкого уровня, т. е. $S0 = S1 = 0$ при $CLR = 1$ (см. шаг 7 и шаг 11 на рис. 5.6). В режиме блокировки данные в регистре не сдвигаются ни вправо, ни влево, а остаются на своих прежних позициях. При установке сигналов $S0 = 0$ и $S1 = 1$ с приходом 8, 9 и 10 тактовых импульсов происходит сдвиг сигнала 1000 вправо и его полный вывод из регистра. Если при работе регистра в режиме сдвига вправо (см. шаги 8, ..., 10, на рис. 5,6) в ячейки памяти генератора внести $SR = 1$, то сигнал 1 будет формироваться на выходе **QD** и сдвигаться вправо от **QD** к **QA** при каждом тактовом импульсе. И, как следствие, после десятого импульса на выходе установится сигнал 1111.

При установке $S0 = S1 = 0$ с приходом 11-го импульса происходит блокировка выходов, на следующем шаге выполняется параллельная запись числа **DCBA** = 1010 в регистр, далее сдвиг данных влево и т. д.

Задание 3. Открыть файл **5.7.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания *последовательного регистра сдвига* (рис. 5.7) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 5.7) в отчёт.

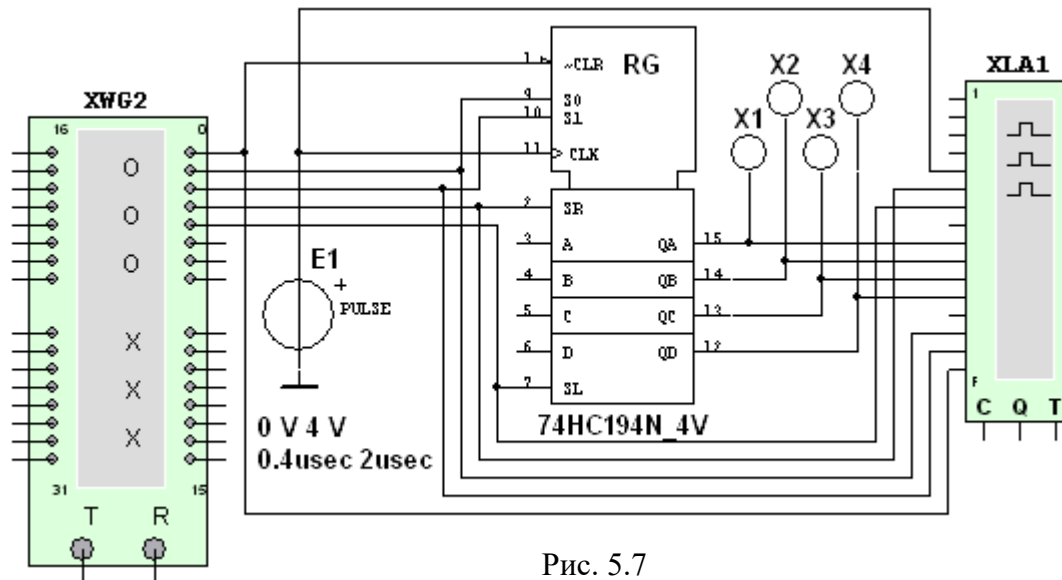


Рис. 5.7

Чтобы микросхема **74HC194N_4V** работала в качестве *последовательного регистра сдвига влево*, нужно подать на управляющий вход **S0** высокий уровень напряжения, а на вход **S1** – низкий уровень, т. е. установить **S0** = 1 и **S1** = 0, и подавать в последовательной форме на вход **SR** данные, например, 1, 0, 1 и 0, которые записываются в разряд **A** и передаются на выход **QA** (рис. 5.8). Регистр последовательно сдвигает влево эти сигналы от **QA** к **QD**, на выходе **QD** они теряются (см. шаги 3, ..., 9 на рис. 5.8).

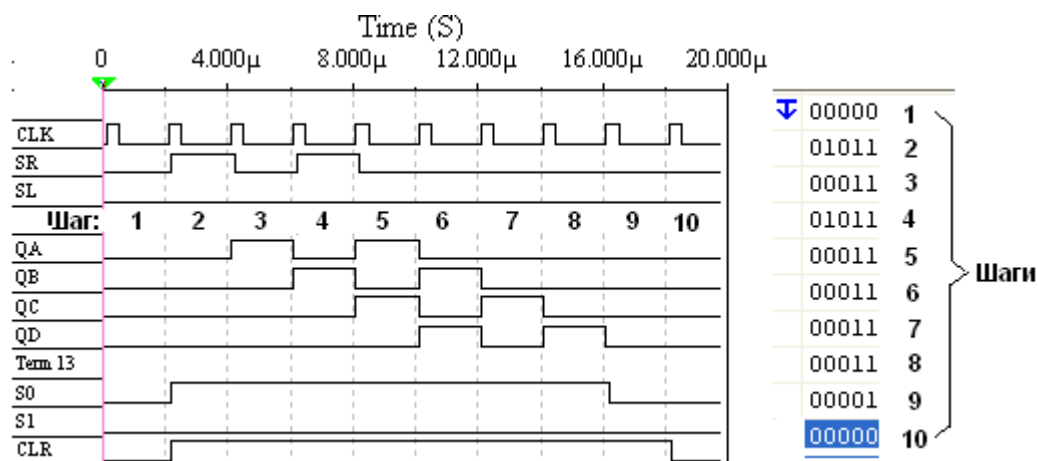


Рис. 5.8

При установке **S1** = 0 и **S1** = 1 и подаче на вход **SL** данных в последовательной форме, например, 1, 0, 0 и 1, которые записываются в разряд **D** (и передаются на выход **QD**), микросхема работает в режиме *последовательного регистра сдвига вправо* (без кольцевого перемещения данных): сигналы 1, 0, 0 и 1

сдвигаются по направлению к разряду **A**, на выходе **QA** они теряются (см. шаги 3, ..., 9, рис. 5.9).

Задание 4. Составить план исследования последовательного регистра **74HC194_4V**, заполнив ячейки памяти генератора **XWG1** произвольными (или по заданию преподавателя) 4-разрядными кодовыми комбинациями, вводимыми последовательно сперва в регистр **A**, а затем в регистр **D**.

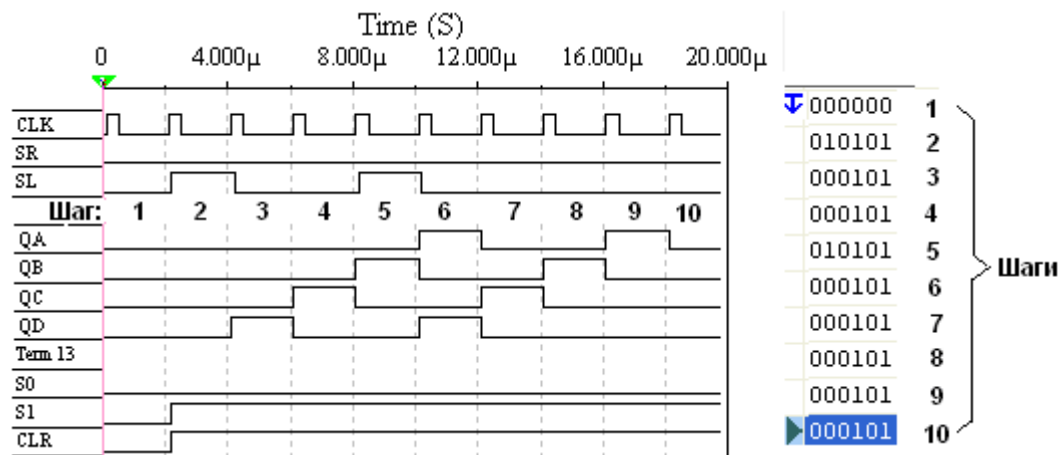


Рис. 5.9

Запустить программу моделирования последовательного регистра, **скопировать** в отчёт временные диаграммы сигналов на входах и выходах регистра при сдвиге данных влево (см. рис. 5.8) и вправо (см. рис. 5.9).

СОДЕРЖАНИЕ ОТЧЁТА

1. Наименование и цель работы.
2. Перечень приборов, использованных в экспериментах, с их краткими характеристиками.
3. Изображения электрических схем для испытания параллельного и последовательного регистров.
4. Копии временных диаграмм, отображающих работу исследуемых регистров.
5. Выводы по работе.

ТЕСТОВЫЕ ЗАДАНИЯ К РАБОТЕ 5

1. Укажите **функции**, которые в общем случае может выполнять регистр.

- ☐ Обнуление (очистку) хранимой информации, запись входной информации в последовательном или в параллельном коде
- ☐ Суммирование по модулю 2 всех разрядов бинарных чисел с целью выяснения чётности числа
- ☐ Сравнение двух бинарных чисел одинаковой разрядности с целью определения их равенства или неравенства
- ☐ Преобразование информации путём её сдвига под воздействием тактовых импульсов
- ☐ Хранение информации, её сдвиг вправо и влево, выдачу хранимой информации в последовательном или в параллельном коде
- ☐ Преобразование десятичных чисел в двоичные или в двоично-десятичные

2. В параллельном регистре с приходом каждого тактового импульса информация на выходах поразрядно сдвигается в направлении от выхода **QD** к выходу **QA**. Укажите, как называют такой регистр?

- ☐ Регистр прямого сдвига
- ☐ Регистр обратного сдвига
- ☐ Реверсивный регистр
- ☐ Регистр хранения.

3. Укажите, какие регистры выполняют со **статическим** управлением?

- ☐ Последовательные
- ☐ Параллельные
- ☐ Последовательно-параллельные
- ☐ Параллельно-последовательные

4. Укажите, при каких **уровнях сигналов** на управляющих входах **S0** и **S1** информационные входы реверсивного регистра **74HC194_4V** недоступны?

- ☐ **S0** = 0, **S1** = 0
- ☐ **S0** = 0, **S1** = 1
- ☐ **S0** = 1, **S1** = 0
- ☐ **S0** = 1, **S1** = 1

5. Укажите, в какой **разряд** вводится информация последовательного регистра **74HC194_4V** при **S0** = 1, **S1** = 0 на управляющих входах и сигналах **SR** = 1 и $\overline{\text{CLR}}$ = 1?

- ☐ В разряд **D**
- ☐ В разряд **C**
- ☐ В разряд **B**
- ☐ В разряд **A**

6. Укажите, при **каких уровнях** управляющих сигналов **S0** и **S1** разрешена запись информации в параллельный регистр **74HC194_4V**?

- ☐ **S0** = 0, **S1** = 0
- ☐ **S0** = 0, **S1** = 1
- ☐ **S0** = 1, **S1** = 0
- ☐ **S0** = 1, **S1** = 1

7. Укажите, разрешено ли последовательное **перемещение** сигналов в триггерной подсистеме параллельного регистра **74HC194_4V** во время записи информации?

- ☐ Да
- ☐ Нет

8. Укажите, сколько **входов** имеет последовательный регистр с динамическим управлением?

- ☐ Один информационный вход
- ☐ Два: один информационный вход и вход для тактовых импульсов (импульсов сдвига)
- ☐ Три: один информационный, вход для тактовых импульсов и установочный вход
- ☐ Четыре: два информационных входа, вход для тактовых импульсов и установочный вход

9. Укажите, чем отличается **динамическое управление** регистрами от статического управления?

☐ Принципиальных отличий нет: сигналы, поступающие на информационные входы всех модификаций регистров, действуют в момент их поступления

☐ У регистров с динамическим управлением сигналы на информационных входах должны оставаться неизменными на всём интервале действия активного логического сигнала синхронизации ($C = 1$)

☐ При динамическом управлении запоминание сигналов, действующих на информационных входах регистра, происходит во входных ёмкостях МДП-транзисторов в момент изменения значения сигнала на входе синхронизации, а в статических регистрах, построенных, например на *RS*-триггерах, сигналы действуют в момент их поступления на информационные входы

Лабораторная работа 6 (Lr6)

СЧЁТЧИКИ ЦЕЛЬ РАБОТЫ

Ознакомление с устройством и функционированием счётчиков и испытание синхронного суммирующего, реверсивного и десятичного счётчиков.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И РАСЧЁТНЫЕ ФОРМУЛЫ

1. КЛАССИФИКАЦИЯ СЧЁТЧИКОВ

Счётчик предназначен для счёта поступающих на его вход импульсов, в интервале между которыми он должен хранить информацию об их количестве. Поэтому счётчик состоит из запоминающих ячеек – триггеров обычно *D*- или *JK*-типа. Между собой ячейки счётчика соединяют таким образом, чтобы каждому числу импульсов соответствовали состояния 1 или 0 определенных ячеек. При этом совокупность единиц и нулей на выходах *n* ячеек, называемых *разрядами* счетчика, представляет собой *n*-разрядное двоичное число, которое однозначно определяет количество прошедших через входы импульсов.

Каждый разряд счётчика может находиться в двух состояниях. Число устойчивых состояний, которое может принимать данный счётчик, называют *коэффициентом пересчёта* $K_{сч}$.

Если с каждым входным импульсом "записанное" в счётчике число увеличивается, то такой счётчик является *суммирующим*, если же оно уменьшается, то – *вычитающим*. Счётчик, работающий как на сложение, так и на вычитание, называют *реверсивным*.

Счётчики, у которых под воздействием входного импульса переключение соответствующих разрядов происходит последовательно друг за другом, называют *асинхронными*, а когда переключение происходит одновременно – *синхронными*. Максимальное число *N*, которое может быть записано в счётчике, равно $(2^n - 1)$, где *n* – число разрядов счётчика.

По способу кодирования последовательных состояний различают *двоичные счётчики* с коэффициентами пересчёта (обнуления) $K_{сч} = 2^n$, у которых порядок смены состояний триггеров соответствует последовательности двоичных чисел, и *недвоичные*, у которых $K_{сч} < 2^n$ (например, десятичные с коэффициентом $K_{сч} = 10$ или делители частоты с коэффициентом деления $K_{сч} \neq 2^n$).

Счётчики входят в состав разнообразных цифровых устройств: электронных часов, делителей частоты, распределителей импульсов, вычислительных и управляющих устройств. Выпускаемые промышленностью интегральные счётчики представляют собой схемы средней интеграции (например, микросхемы серий K155, K176 и др.); среди них многоразрядные бинарные счётчики на сложение и реверсивные счётчики с установочными входами *R* и *S* для всех разрядов, с постоянными и произвольными коэффициентами пересчёта.

2. СЧЁТЧИК С НЕПОСРЕДСТВЕННЫМИ СВЯЗЯМИ

Условное изображение трехразрядного *суммирующего* счётчика показано на рис. 6.1, а, на котором символом R обозначен вход общего сброса, символами Q_1 , Q_2 и Q_3 – выходы счетчика, CR – выход переноса единицы. Суммирующий вход счётчика обозначается $+1$, вычитающий -1 . Это счетные входы. У асинхронных счётчиков эти входы помечены специальными символами: \triangleright или \triangleleft , указывающими полярность перепада входного сигнала: 1/0 или 0/1, при которой происходит переключение триггеров счётчика.

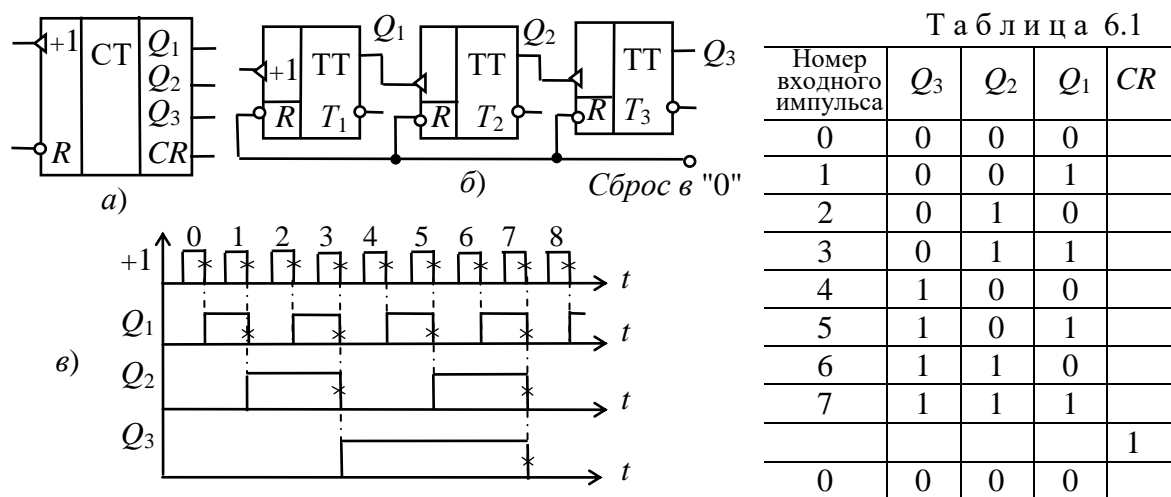


Рис. 6.1

Для переключения триггеров в счётчиках используют следующие связи: непосредственную, тракт последовательного переноса, тракт параллельного переноса. Схема счётчика с непосредственными связями показана на рис. 6.1, б. Первый триггер счётчика T_1 образует младший разряд. Он пересчитывает входные импульсы по модулю 2, а состояние его выхода воспринимается следующим T_2 триггером как входные сигналы и снова пересчитываются на 2 и т. д.

Полное представление о состояниях счётчика (рис. 6.1, б), в зависимости от числа поданных на вход импульсов, даёт переключательная таблица (табл. 6.1) и временные диаграммы (рис. 6.1, в), где изображены последовательность входных импульсов (на входе $+1$), а также состояния триггеров – первого (Q_1), второго (Q_2) и третьего (Q_3). Фронты импульсов на диаграммах показаны идеальными: потенциал, соответствующий логическому 0, считается равным нулю, переключающие перепады для наглядности помечены крестиками.

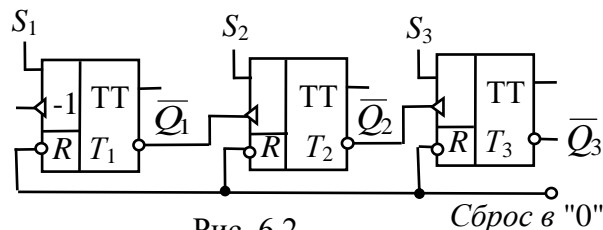
Рассмотрим воздействие на счётчик, к примеру, шестого (обозначенного на диаграмме цифрой 5) импульса. По его спаду триггер T_1 устанавливается в 0, перепад 1/0 на его выходе Q_1 переключает в 1 триггер T_2 , а триггер T_3 остается в прежнем (единичном) состоянии, так как перепад 0/1 на выходе Q_2 не является для него переключающим.

Из диаграммы видно, что частота импульсов на выходе каждого триггера вдвое меньше частоты импульсов на его входе. В момент, предшествующий

переключению очередного разряда, все предыдущие разряды счётчика находятся в состоянии 1. Восьмой импульс для трехразрядного счётчика (см. табл. 6.1) является импульсом переполнения: им все триггеры устанавливаются в 0 (счётчик "обнуляется").

Если в счётчике используются триггеры, переключающиеся перепадом 0/1, то вход последующего триггера нужно соединить с инверсным выходом предыдущего, на котором формируется этот перепад, когда по основному выходу триггер переключается из 1 в 0.

Схема *вычитающего* счётчика приведена на рис. 6.2, в которой по входам S в разряды счётчика заносят двоичное число, из которого нужно вычесть число, представляемое количеством входных импульсов. Пусть, например, в счётчик (рис. 6.2) занесено число $5_{10} = 101_2$. Первым входным импульсом триггер T_1 переключится из 1 в 0 (по основному выходу); при этом на инверсном выходе \overline{Q}_1 возникает перепад 0/1, которым триггер T_2 переключиться не может; в счётчике останется число $100_2 = 4_{10}$.



Второй входной импульс устанавливает триггер T_1 в состояние 1, на выходе \overline{Q}_1 появляется перепад 1/0, который переключает T_2 в состояние 1, а формирующийся при этом на \overline{Q}_2 перепад 1/0 переключает T_3 в состояние 0. В счётчике остается число $011_2 = 3_{10}$. Аналогично можно рассмотреть действие последующих входных импульсов.

В счётчике с непосредственной связью переключение триггеров, вызванное срезом входного сигнала, происходит один за другим, *последовательно*, и задержка распространения n -разрядного счётчика, оцениваемая задержкой самого худшего случая – сменой всех 1 на все 0, – в n раз больше задержки одного T -триггера. Если разрядов много, то большая задержка может оказаться серьёзным недостатком такого счётчика. Из-за невозможности выполнить смену состояния всего счётчика в единый момент времени, счётчики с непосредственной связью бывают только *асинхронными*, т. е. сигналом, переключающим их, является сам входной сигнал.

3. СУММИРУЮЩИЙ СИНХРОННЫЙ СЧЁТЧИК

В *синхронном* счётчике переключающиеся разряды переходят в новое состояние одновременно (синхронно). Для того чтобы на входы всех разрядов каждый счётный импульс поступал одновременно, а переключение разрядов происходило в нужной последовательности, в схему добавляют логические

цепи, которые обеспечивают переключение одних разрядов, а другие удерживают от переключения.

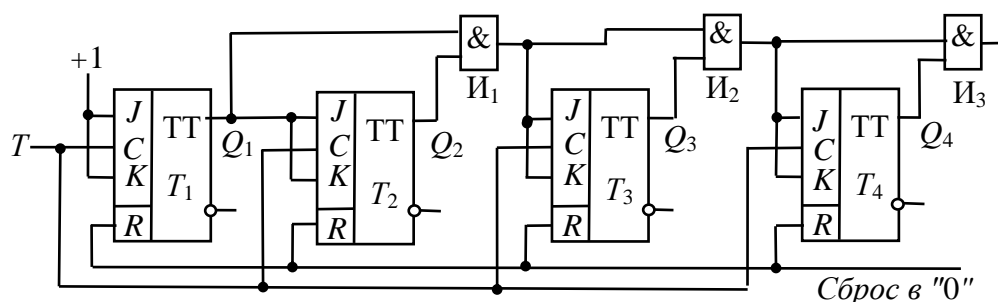


Рис. 6.3

В схеме (рис. 6.3) четырехразрядного синхронного счётчика на JK -триггерах на тактовые входы C всех триггеров счётные импульсы поступают одновременно с входа T . Информационные входы J и K каждого триггера объединены. Триггер T_1 переключается каждым счётным импульсом, так как на его входы J и K постоянно подаётся 1. Остальные триггеры переключаются счётными импульсами при следующих условиях:

T_2 – при $Q_1 = 1$; T_3 – при $Q_1 = 1$ и $Q_2 = 1$; T_4 – при $Q_1 = 1$, $Q_2 = 1$ и $Q_3 = 1$.

Чтобы обеспечить указанные условия переключения триггеров, в схему (рис. 6.3) добавлены конъюнкторы $И1$, $И2$ и $И3$. На информационный вход каждого из триггеров T_2 , T_3 и T_4 подаётся конъюнкция сигналов с основных выходов предыдущих триггеров. Разрешающая переключение единица поступит на вход соответствующего триггера, если все предыдущие триггеры находятся в состоянии 1, и по счётному сигналу он переключается.

4. РЕВЕРСИВНЫЙ СИНХРОННЫЙ СЧЁТЧИК

Реверсивный счётчик, фрагмент которого изображен на рис. 6.4, работает как на сложение, так и на вычитание. Для перехода от сложения к вычитанию и обратно изменяют подключение входа последующего триггера к выходам предыдущего.

На объединённые входы J и K каждого триггера подаётся через дизъюнкторы конъюнкция сигналов с выходов предыдущих триггеров: основные выходы предыдущих триггеров присоединяются через конъюнкторы верхнего ряда (при сложении), а инверсные выводы – через конъюнкторы нижнего ряда (при вычитании). При сложении подают 1 на шину сложения, которой вводятся в действие конъюнкторы верхнего ряда; при этом на шине вычитания присутствует 0, вследствие чего конъюнкторы нижнего ряда выключены. Вычитание осуществляется при подаче 1 на шину вычитания и 0 на шину сложения. Счётные импульсы поступают на вход T .

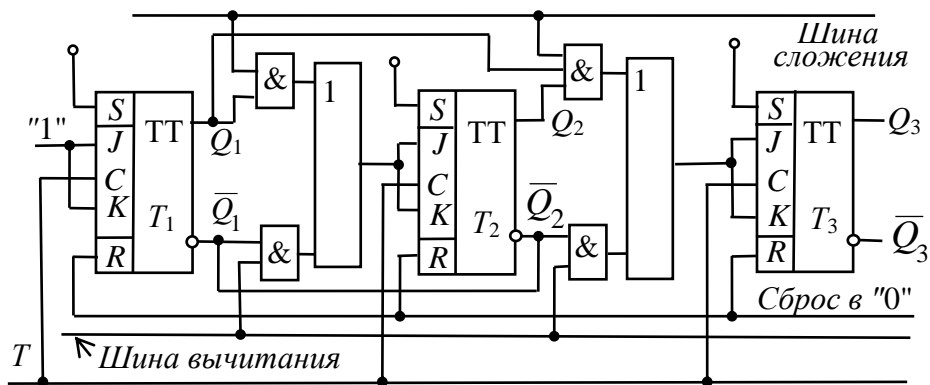


Рис. 6.4

Как отмечалось в п. 2, каждый триггер переключается по тактовому входу C при $J = K = 1$, что имеет место, когда на выходах всех предыдущих триггеров (на основных – при сложении, на инверсных – при вычитании) будут единицы. Функционирование счетчика при сложении и вычитании описано в п. 2 и в п. 3.

5. ДЕСЯТИЧНЫЙ СЧЁТЧИК

Наибольшее распространение среди двоичных счётчиков, у которых коэффициент пересчёта $K_{сч} < 2^n$, имеют десятичные счётчики, у которых $K_{сч} = 10$. При проектировании двоичного счётчика вначале определяют количество его разрядов n так, чтобы 2^n было бóльшим ближайшим к $K_{сч}$ числом. Затем тем или иным способом (например, принудительной установкой некоторых разрядов счётчика в 1) исключают избыточные состояния счётчика, число которых равно $2^n - K_{сч}$.

Так, для получения $K_{сч} = 10$ одноразрядный счётчик должен содержать четыре триггера, а избыточные состояния $2^n - K_{сч} = 16 - 10 = 6$ исключают тем или иным способом. При проектировании десятичного счётчика чаще используют двоично-десятичное кодирование чисел. В этой системе, например, число 375 записывается как 0011 0111 0101, где сохранены позиции десятичных разрядов: $0011_2 = 3_{10}$, $0111_2 = 7_{10}$, $0101_2 = 5_{10}$.

В связи с этим десятичный счётчик должен состоять из последовательно соединенных декад, информация о каждом из девяти импульсов накапливается в декаде, а десятым импульсом она обнуляется, и единица переносится в следующую декаду. Каждая декада работает в натуральном двоичном коде с весами двоичных разрядов, начиная со старшего, соответственно равными 8, 4, 2, 1, т. е. декада работает в коде 8-4-2-1. Если к выводам декад подключить индикаторы, то они будут показывать записанные числа в декадах в привычном десятичном коде. Десятичные счетчики выполняются и с другими весами разрядов, например, в коде 4-2-2-1.

На рис. 6.5 изображена функциональная схема десятичного счётчика с параллельным переносом на JK -триггерах с встроенными логическими элементами, реализующая переключательные функции:

$$J_1 = 1; J_2 = Q_1 \bar{Q}_4; J_3 = Q_1 \bar{Q}_2; J_4 = Q_1 Q_2 Q_3;$$

$$K_1 = 1; K_2 = Q_1; K_3 = Q_1 Q_2; K_4 = Q_1.$$

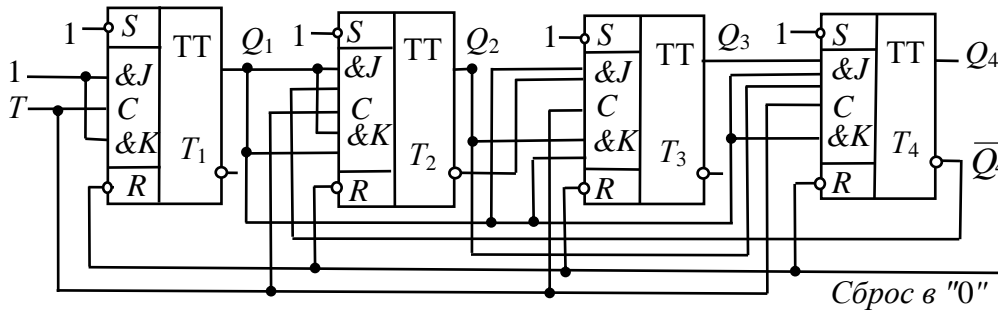


Рис. 6.5

Рассмотрим работу схемы. Пусть по тактовому входу T на триггер T_1 поступило семь импульсов и показание счётчика 0111. При этом на входах K триггеров T_1 , T_2 и T_3 будут логические единицы. Восьмой импульс вызовет переключение всех триггеров счетчика, т. е. в нём будет записан код 1000. Девятый импульс вызовет переключение только первого триггера, так как остальные триггеры заблокированы по входу J уровнями логического нуля с триггеров T_1 , T_2 и T_3 соответственно. Показание счётчика будет 1001. Десятый входной импульс вызовет переключение триггеров T_1 и T_4 , так как триггеры T_2 и T_3 заблокированы по входу J уровнями 0 с выходов соответствующих триггеров. Счетчик зафиксирует двоичный код 0000, т. е. установится в исходное состояние.

Уменьшение числа устойчивых состояний в счётчике прямого счёта достигнуто за счёт введения обратных связей, посредством которых сигнал с какого-либо старшего разряда поступает в младшие, обеспечивая при этом изменение естественной последовательности двоичных чисел при подсчёте входных импульсов. Этим способом можно строить счётчики с заданным коэффициентом пересчёта.

УЧЕБНЫЕ ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ИХ ВЫПОЛНЕНИЮ

Задание 1. Запустить лабораторный комплекс Labworks и среду MS10 (щёлкнув мышью на команде **Эксперимент** меню комплекса Labworks). Открыть файл **6.6.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания **синхронного двоичного счётчика** (рис. 6.6) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 6.6) на страницу отчёта.

В библиотеке программной среды MS10 имеются 4-, 8- и 12-разрядные счётчики различных типов. Среди них: счётчики асинхронные (**SN7493**, **SN74393**), синхронные (**74NC161**, **SN74163**), реверсивные (**SN74191**), двоично-десятичные (**SN7493**, **SN74160**) и др.

В схему (рис. 6.6) включен синхронный двоичный 4-разрядный счётчик **74NC161**, к входу **CLK** которого подключен источник тактовых импульсов **E1**, а к выходам **QA**, **QB**, **QC** и **QD** – шестнадцатеричный 7-сегментный индикатор

DCD_HEX и дешифратор **DC 4x10**. Выход дешифратора соединён с входами логического анализатора **XLA1**.

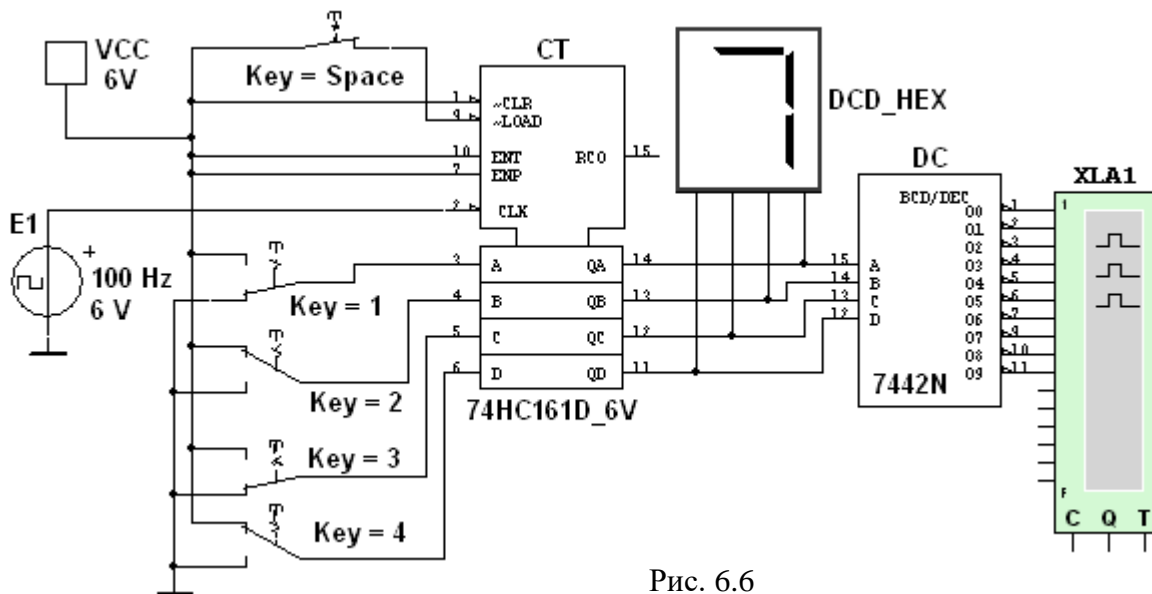


Рис. 6.6

К входам **A**, **B**, **C** и **D** счётчика **СТ** подключен источник постоянного напряжения **VCC**, переключатели **1**, ..., **4** для формирования входных двоичных кодов и ключ **Space** для изменения режима работы счётчика. В синхронном счётчике заданные с помощью ключей уровни сигналов подаются на входы всех триггеров, как и тактовые импульсы, которые подаются на счётные входы **CLK** всех разрядов счётчика.

При замкнутом ключе **Space** число поданных от генератора **E1** на вход счётчика импульсов высвечивается на индикаторе **DCD_HEX** в десятичном коде, от 0 до 15, после чего счётчик обнуляется и вновь начинается счёт. При этом на одном из выходов дешифратора **DC** формируется сигнал низкого уровня (логический 0), номер которого соответствует коду входного числа: от 0000 до 1001 (9_{10}).

При разомкнутом ключе **Space** сформированное с помощью переключателей на входе счётчика 4-разрядное двоичное число высвечивается на индикаторе в десятичном коде, а на экране анализатора на одном из выходов, соответствующем входному коду счётчика, формируется логический 0.

Задание 2. Замкнуть ключ **Space**, запустить программу моделирования суммирующего счётчика и наблюдать за показаниями индикатора. Убедиться, что на экране анализатора **XLA1** логические нули перестают формироваться после прихода 11-го тактового импульса и появляются вновь только с приходом 17-го импульса (рис. 6.7, а).

Разомкнуть ключ **Space**. Установить в диалоговом окне анализатора **XLA1** напряжение $V = 5$ В, частоту таймера $f_a = 2$ кГц, число импульсов, приходящихся на одно деление, **Clocks/div** = 60. (При таком режиме лучи медленно перемещаются на экране анализатора). С помощью активных клавиш 1, 2, 3 и 4 клавиатуры сформировать произвольные (или по указанию преподавателя) двоичные входные числа (коды), например 1001, 0011, 0000,

1110 и **подавать** их на входы **D, C, B** и **A** счётчика. **Убедиться**, что при подаче числа 1110_2 (14_{10}) ни на одном выходе дешифратора 4x10 не сформировался низкий уровень сигнала (рис. 6.7, б).

Скопировать на страницу отчёта результаты моделирования синхронного суммирующего счётчика (см. рис. 6.7).

Задание 3. Открыть файл **6.8.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для

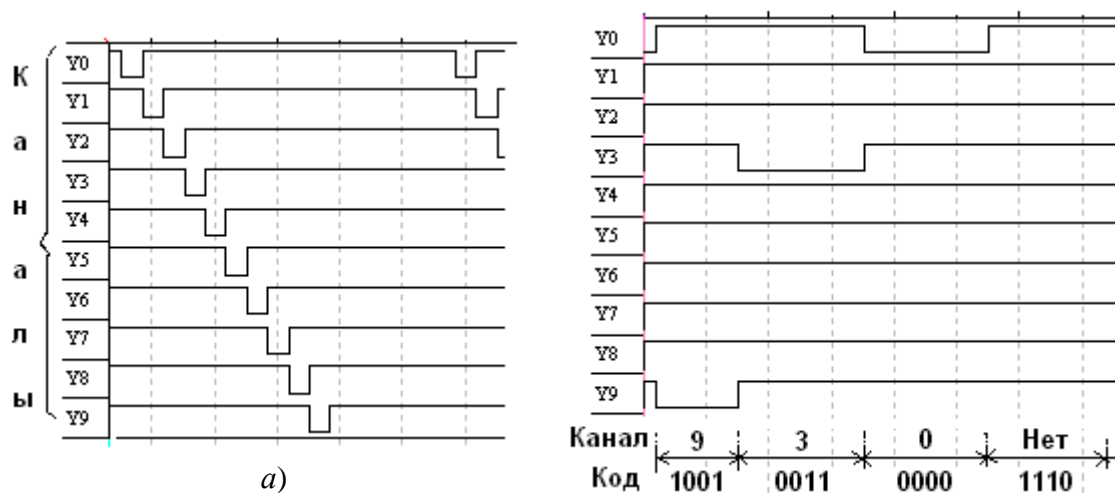


Рис. 6.7

б)

испытания *реверсивного двоичного счётчика* (рис. 6.8) и **установить** в диалоговых окнах компонентов их параметры или режимы работы.

Скопировать схему (рис. 6.8) на страницу отчёта.

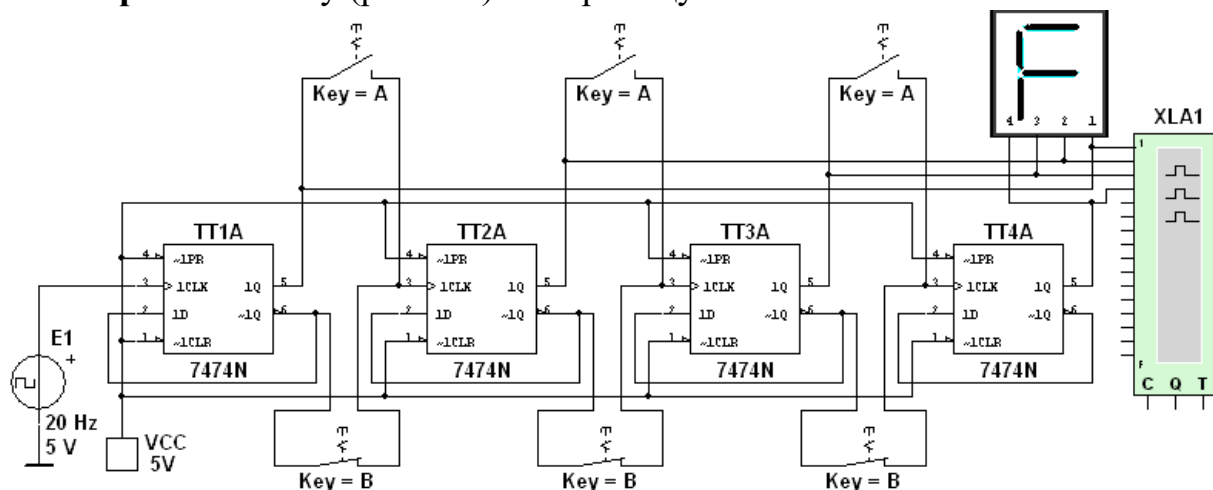


Рис. 6.8

В схеме реверсивного двоичного счетчика (рис. 6.6) с помощью групп ключей **A** и **B** осуществляется замыкание или размыкание выходов высокого или низкого уровня предыдущего триггера с входами $\overline{1PR}$ или $\overline{1CLR}$ триггера следующего разряда, причём при замкнутых ключах **A** и разомкнутых **B** (режим суммирования) с каждым тактовым импульсом увеличивается результат счёта, а при замкнутых ключах **B** и разомкнутых **A** (режим вычитания) – результат счёта уменьшается.

Установить в диалоговом окне анализатора **XLA1** напряжение $V = 5 \text{ В}$, частоту таймера $f_a = 2 \text{ кГц}$, число импульсов, приходящихся на одно деление, **Clocks/div** = 60. **Разомкнуть** ключи **В** и **замкнуть** ключи **А**. **Запустить** программу моделирования счётчика. При высвечивании числа 15 на 7-сегментном индикаторе **щёлкнуть мышью** на кнопке **Stop** (остановки моделирования) и **скопировать** окно анализатора с результатами моделирования на страницу отчёта (см. верхнюю часть рис. 6.9).

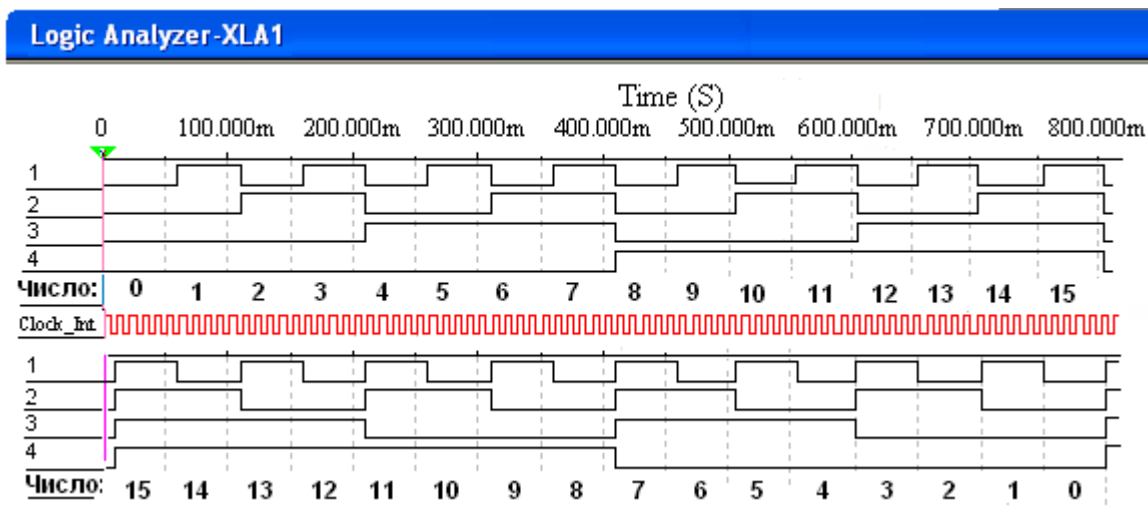


Рис. 6.9

Разомкнуть ключи **А** и **замкнуть** ключи **В**. **Щёлкнуть мышью** на кнопке **Stop** (продолжить моделирование), **остановить** моделирование при высвечивании числа 0 на индикаторе и **скопировать** окно анализатора с результатами моделирования в отчет (см. нижнюю часть рис. 6.9).

Задание 4. Открыть файл **6.10.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания *десятичного счётчика* (рис. 6.10) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 6.10) на страницу отчёта.

Результаты моделирования 4-разрядного двоичного счетчика (см. рис. 6.8 и рис. 6.9) показали, что с его помощью можно сосчитать до 15. Следовательно, для создания счётчика натуральных десятичных чисел в двоичном коде для одной декады нужно в схему двоичного 4-разрядного счётчика ввести обратные связи с логическими элементами, посредством которых сигнал с какого-то старшего разряда поступает в младшие и т. п., обеспечивая в целом выработку счётчиком двоичного эквивалента счётной декады.

В функциональной схеме десятичного счётчика (6.10), собранной на триггерах *JK*-типа, на каждый одиннадцатый тактовый импульс результат счёта сбрасывается в нуль и далее результат счёта увеличивается. Возврат счётчика при поступлении одиннадцатого тактового импульса в начальное состояние обеспечивается дополнительной комбинационной схемой с встроенными логическими элементами **И (AND)**, **ИЛИ (OR)** и **НЕ (NOT)**.

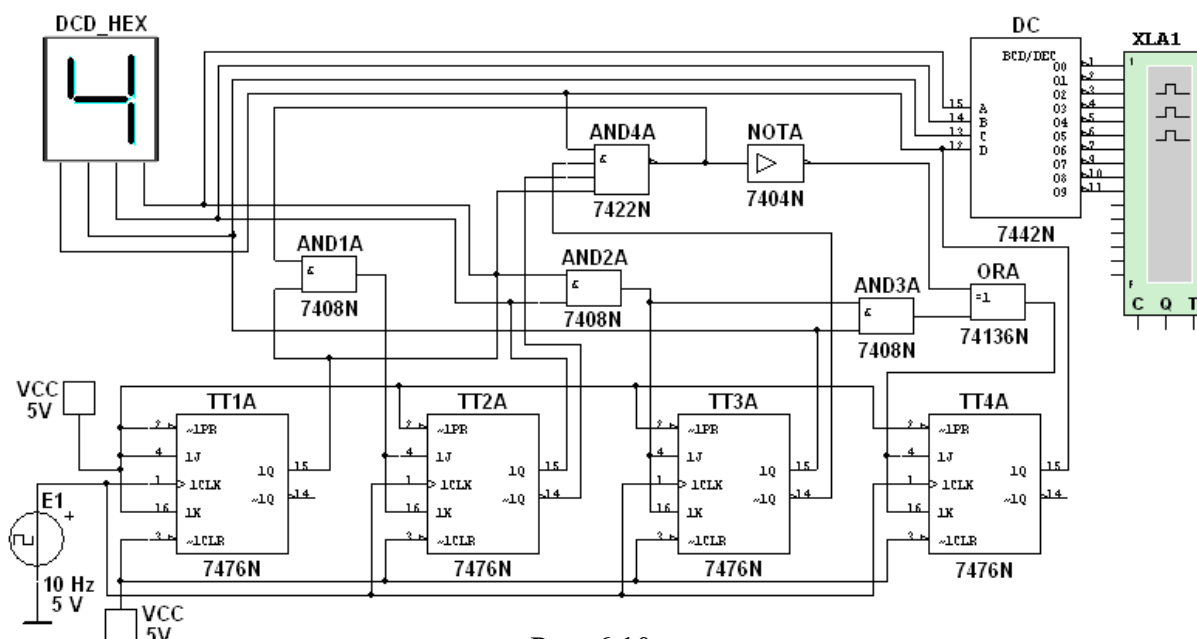


Рис. 6.10

Счётчик работает так же, как синхронный двоичный счётчик до поступления седьмого импульса, а далее, благодаря обратной связи, нарушается изменение естественной последовательности двоичных чисел на входах и выходах триггеров при подсчёте тактовых импульсов.

Запустить программу моделирования десятичного счётчика и **скопировать** окно анализатора с результатами моделирования на страницу отчёта.

СОДЕРЖАНИЕ ОТЧЁТА

1. Наименование и цель работы.
2. Перечень приборов, использованных в экспериментах, с их краткими характеристиками.
3. Изображения электрических схем для испытания суммирующего, реверсивного и десятичного счётчиков.
4. Копии временных диаграмм, отображающие работу исследуемых счётчиков.
5. Выводы по работе.

ТЕСТОВЫЕ ЗАДАНИЯ К РАБОТЕ 6

1. Укажите, **в каком виде** фиксируется в счётчике число поступивших на его вход импульсов?

- ☐ В виде двоичного кода, хранящегося в триггерах
- ☐ В виде потенциала (напряжения), хранящегося на зажимах выходного конденсатора счётчика
- ☐ В виде двоично-десятичного кода, хранящегося в выходном регистре
- ☐ В виде десятичного числа, высвечиваемого на индикаторе

2. Укажите необходимое **число выходов** двоичного счётчика для выдачи результатов счёта 28 импульсов.

- ☐ 3 ☐ 4 ☐ 5 ☐ 6 ☐ 8

3. Укажите, в **какой момент** 5-разрядный двоичный счетчик возвращается в начальное состояние?

- ☐ При поступлении на вход 16-го импульса
- ☐ При подаче на вход 32-го импульса
- ☐ При подаче на вход инверсного сигнала
- ☐ При переполнении, наступающем при числе импульсов $N = 2^5 - 1$

4. На 7-сегментном индикаторе десятичного счётчика высвечивается число 5. Укажите, какое **число** будет высвечиваться на индикаторе при подаче на вход ещё 6-ти импульсов?

- ☐ 0 ☐ 1 ☐ 2 ☐ 3

5. Укажите, **каким путём передаются сигналы** от разряда к разряду в синхронном счётчике?

☐ Естественным путём в различные интервалы времени в зависимости от сочетания входных сигналов

- ☐ Принудительным путём с помощью тактовых импульсов
- ☐ Посредством специальной переключающей схемы
- ☐ Путём подачи сигнала 0 на входы J всех JR -триггеров

6. Укажите, что понимают под **коэффициентом пересчёта** счётчика?

☐ Это минимально допустимый период следования входных импульсов, при котором обеспечивается надёжная работа счётчика

☐ Это интервал времени между моментами поступления входного импульса и окончания самого длинного переходного процесса в счётчике

☐ Это максимальное число единичных сигналов, которое может быть зафиксировано на счётчике

☐ Это модуль счёта, характеризуемый числом устойчивых состояний счётчика

7. Укажите, чему равен **модуль M пересчёта** двоичного n -разрядного счётчика?

- ☐ $M = 2^n$ ☐ $M = 2^n - 1$ ☐ $M = 2^n - 2$ ☐ $M = 2^{n-1}$

8. Укажите, сколько **триггеров** должен иметь двоично-кодированный счётчик с коэффициентом пересчёта $M = 8$?

- ☐ 2 ☐ 3 ☐ 4 ☐ 5 ☐ 6

9. Укажите **пути и средства**, с помощью которых изменяется направление счёта в реверсивном счётчике.

☐ Направление счёта определяется исключительно выбором инверсных выходов триггером для формирования сигнала переноса

☐ Направление счёта осуществляется с помощью разбиения разрядных схем счётчика на группы и применением внутри этих групп последовательного переключения триггеров

☐ Направление счёта изменяется путём изменения вида междуразрядных связей

☐ Изменение направления счёта осуществляется путём исключения лишних состояний разрядных схем

ЦИФРОАНАЛОГОВЫЙ ПРЕОБРАЗОВАТЕЛЬ

ЦЕЛЬ РАБОТЫ

Ознакомление с принципом работы и испытание интегрального цифроаналогового преобразователя.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И РАСЧЁТНЫЕ ФОРМУЛЫ

1. СТРУКТУРА РЕЗИСТИВНЫХ МАТРИЦ ЦАП

При построении устройств, связывающих цифровое устройство с объектами, использующими информацию в непрерывно изменяющейся форме, требуется преобразование информации из аналоговой формы в цифровую и из цифровой в аналоговую. Устройство, осуществляющее автоматическое преобразование непрерывно изменяющихся во времени аналоговых значений физической величины (напряжения, тока) в эквивалентные значения числовых кодов, называют *аналого-цифровым преобразователем (АЦП)*. Устройство, осуществляющее автоматическое преобразование входных значений, представленных числовыми кодами, в эквивалентные им значения какой-нибудь физической величины (напряжения, тока и др.), называют *цифроаналоговым преобразователем (ЦАП)*.

Итак, цифроаналоговый преобразователь предназначен для прямого преобразования входного двоичного кода, например, $A_i(a_2a_1a_0)$ в аналоговый эквивалент. Выходная аналоговая величина, обычно напряжение $u_{вых}$, иногда нормированное $u_{вых.н} = u_{вых}/u_{вых.мах}$, соответствует кодовой комбинации A_i , поступившей на вход, и воспроизводится для дискретных моментов времени (рис. 7.1, а). Сменяющиеся входные цифровые коды обуславливают сменяющееся ступенчатое напряжение на выходе (L – идеальная передаточная

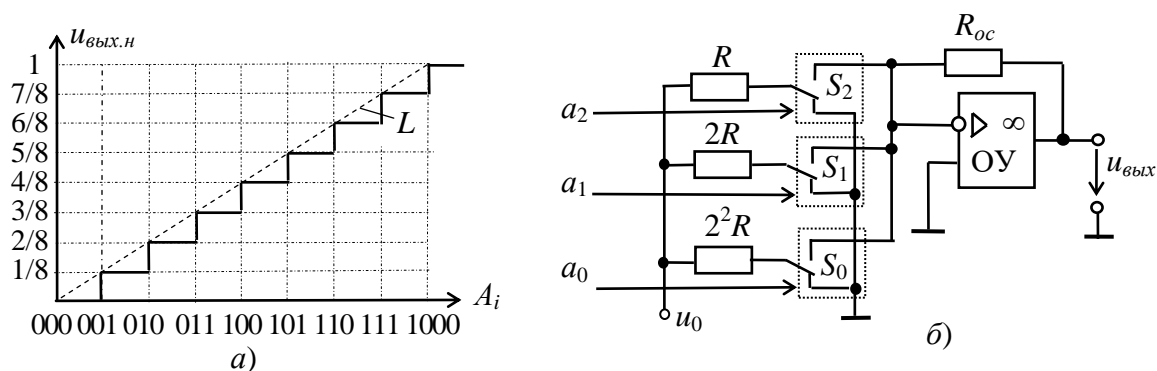


Рис. 7.1

характеристика ЦАП).

Существует два широко распространенных способа цифроаналогового преобразования с использованием:

- резистивной матрицы с весовыми двоично-взвешенными сопротивлениями;
- резистивной матрицы с двумя номиналами сопротивлений, которую обычно называют матрицей $R-2R$.

ЦАП с весовыми двоично-взвешенными сопротивлениями (рис. 7.1, б) состоит: из n переключателей S_i (по одному на каждый разряд), управляемых двоичным кодом A_i ; из матрицы двоично-взвешенных резисторов с сопротивлениями $2^{n-1}R$; источника опорного напряжения u_0 и выходного операционного усилителя ОУ, с помощью которого суммируются токи, протекающие через резисторы с двоично-взвешенными сопротивлениями, для получения аналогового выходного напряжения $u_{вых}$.

Каждый i -й разряд управляет переключателем S_i , который подключается к источнику опорного напряжения u_0 , когда $a_i = 1$, или к общей шине, когда $a_i = 0$. Сопротивления резисторов $2^{n-1}R$ (n – номер разряда входного кода), соединенных с ключами, таковы, что обеспечивают пропорциональность в них тока двоичному весу соответствующего разряда входного кода. Следовательно, ток на входе ОУ и выходное напряжение ЦАП:

$$i = \frac{a_{n-1}u_0}{R} + \frac{a_{n-2}u_0}{2R} + \dots + \frac{a_1u_0}{2^{n-1}R} + \frac{a_0u_0}{2^n R}; \quad u_{вых} = -R_{oc}i = -u_0 \frac{R_{oc}}{2^n R} \sum_{i=0}^{n-1} a_i 2^i.$$

Напряжение на выходе ЦАП пропорционально "весу" присутствующего на входах кода, а максимальное значение имеет место, когда все разряды примут значение 1, т. е.

$$u_{max} = \left| u_0 \frac{(2^n - 1)R_{oc}}{2^n R} \right|,$$

и оно всегда меньше опорного напряжения на шаг квантования $u_0 R_{oc} / (2^n R)$.

Номиналы сопротивлений резисторов в младшем и старшем разрядах отличаются в 2^{n-1} раз и должны быть выдержаны с высокой точностью. Например, для 12-разрядного ЦАП использование в старшем разряде резистора с сопротивлением 10 кОм потребует включения в младший разряд преобразователя резистора с сопротивлением порядка 20 МОм. Широкий набор

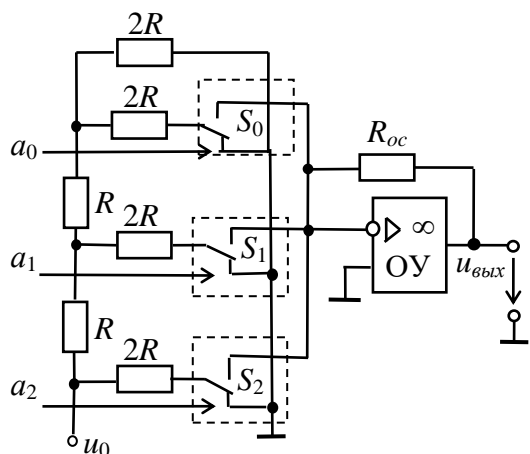


Рис. 7.2

номиналов резисторов и требования их высокой точности, в особенности при значительном числе разрядов n входного кода, создают трудности при реализации ЦАП посредством интегральной технологии.

Во второй схеме ЦАП с матрицей $R-2R$ используют резисторы с двумя номиналами сопротивлений, причём резисторы с сопротивлением R включены в каждый разряд (см. рис. 7.2 при $n = 3$). Однако в

этой схеме увеличиваются значения паразитных ёмкостей.

Принцип функционирования схемы основан на свойстве резистивного делителя $R-2R$ сохранять постоянное сопротивление нагрузки для источника опорного напряжения при замыкании ключей. Вследствие этого на выводах резистора R , начиная со старшего $n - 1$ разряда, опорное напряжение последовательно делится пополам, как и входящий в каждый узел матрицы ток. При этом напряжение на выходе преобразователя с матрицей $R-2R$:

$$u_{вых} = -u_0 \frac{R_{oc}}{R} (a_{n-1} 2^{-1} + a_{n-2} 2^{-2} + \dots + a_1 2^{-(n-1)} + a_0 2^{-n}) = -u_0 \frac{R_{oc}}{2^n R} \sum_{i=0}^{n-1} a_i 2^i.$$

Таким образом, выходное напряжение ЦАП пропорционально сумме напряжений со своими весами, обусловленными переключателями, подключенными к источнику опорного напряжения u_0 .

Недостатком ЦАП с матрицей $R-2R$ является сильное влияние на точность преобразования нестабильности сопротивлений переключателей в замкнутом состоянии, что снижает временную и температурную стабильность характеристик ЦАП. Этот недостаток в значительной степени удаётся устранить в схемах код-напряжение, выполненных на базе полупроводниковой технологии с использованием тонкоплёночных резисторов на кристалле и переключателей на КМДП-транзисторах, в которых нелинейность от $\pm 0,8\%$ до $\pm 0,003\%$ от опорного напряжения u_0 , время установления тока от 5 мкс до десятых долей микросекунд и менее, часто выходной диапазон напряжения ± 5 В. Опорное напряжение в схемах ЦАП может выбираться разной полярности или двуполярным.

2. ОСНОВНЫЕ ПАРАМЕТРЫ ЦАП

Основными параметрами ЦАП являются число разрядов $n = 8, \dots, 24$ и *абсолютная разрешающая способность* – среднее значение минимального изменения сигнала на выходе ЦАП, обусловленное увеличением или уменьшением его кода на единицу. Теоретически ЦАП, преобразующий n -разрядные двоичные коды, должен обеспечить 2^n различных значений выходного сигнала с разрешающей способностью $1/(2^n - 1)$. При числе разрядов $n = 8$ количество независимых квантов (ступеней) выходного напряжения ЦАП равно $2^8 - 1 = 255$, при $n = 12$, $2^{12} - 1 = 4095$ и т. д.

Абсолютное значение минимального кванта напряжения определяется как предельным принимаемым числом $2^n - 1$, так и максимальным выходным напряжением ЦАП, по-другому называемым напряжением шкалы или опорным напряжением u_0 . Значение абсолютной разрешающей способности ЦАП, часто обозначаемое ЗМР (значение младшего разряда), при $n = 8$ и опорном напряжении $u_0 = 5$ В

$$\text{ЗМР} = u_0 / (2^8 - 1) = 5 / 255 \approx 0,0196 \text{ В} = 19,6 \text{ мВ}.$$

Отличие реального значения разрешающей способности от теоретического обусловлено погрешностями и шумами входящих в ЦАП узлов. Точность ЦАП определяется значением абсолютной погрешности δ_a и нелинейностью

преобразователя δ_n . Абсолютная погрешность δ_a характеризуется отклонением максимального значения выходного напряжения u_{max} от расчётного, соответствующего конечной точке характеристики идеального преобразователя, и измеряется обычно в единицах ЗМР.

Нелинейность преобразователя δ_n характеризует отклонение действительной характеристики от линейной (от прямой линии L , см. рис. 7.1, а), проведенной через центры ступенек или через нуль и точку максимального значения выходного сигнала.

Из динамических параметров наиболее важным является максимальная частота преобразования f_{max} (десятки и сотни кГц) – наибольшая частота дискретизации, при которой параметры ЦАП соответствуют заданным значениям.

Работа ЦАП часто сопровождается специфическими переходными импульсами в выходном сигнале, возникающими из-за разности времени открывания и закрывания аналоговых переключателей в ЦАП. Особенно значительно выбросы проявляются, когда входной код 01...111 сменяется кодом 10...000, а переключатель старшего разряда ЦАП открывается позже, чем закрываются переключатели младших разрядов. Вследствие определённой идеализации при моделировании библиотечных ЦАП среды MS10 не всегда удаётся определить отмеченные выше параметры.

Библиотечные интегральные схемы ЦАП среды MS10 требуют для своей работы подключения только постоянного эталонного напряжения, заземления и входных сигналов.

УЧЕБНЫЕ ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ИХ ВЫПОЛНЕНИЮ

Задание 1. Запустить лабораторный комплекс Labworks и среду MS10 (щёлкнув мышью на команде **Эксперимент** меню комплекса Labworks). Открыть файл **7.3.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания интегрального *цифроаналогового преобразователя* (рис. 7.3, а) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 7.3, а) на страницу отчёта.

В схеме (рис. 7.3, а) использован библиотечный (**Mixed**) 8-разрядный цифроаналоговый преобразователь **DAC**, на входы которого подаются сформированные с помощью переключателей **0**, ..., **7** двоичные коды от 00000000 до 11111111₂ (FF₁₆ или 255₁₀). Выходное напряжение ЦАП можно измерить с помощью вольтметра **V1** или осциллографа **XSC1**, воспользовавшись визирными линиями, расположенными на его экране.

Задание 2. Получить на экране осциллографа **XSC1** ступенчатое выходное напряжение ЦАП (рис. 7.3, б). Для этого нужно вначале **замкнуть** переключатель **0**, т. е. **подать** напряжение 5 В на вход **D0** ЦАП, и **запустить** программу моделирования. На выходе ЦАП формируется напряжение, равное

ЗМР. Затем во время остановок моделирования замыкать поочерёдно

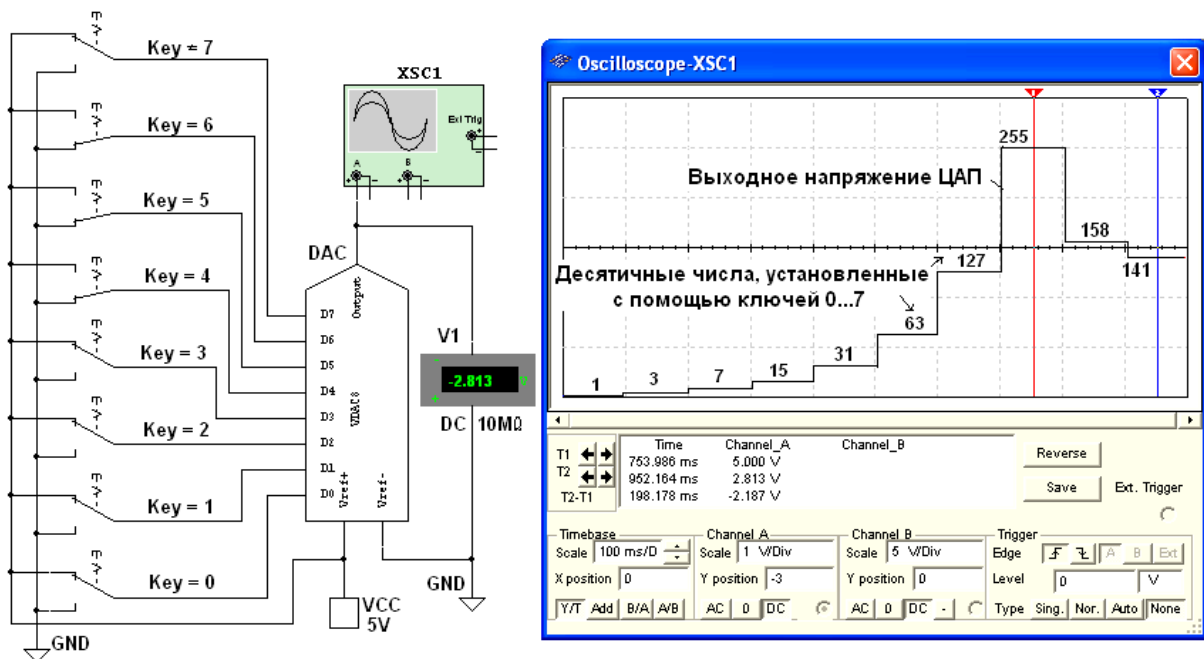


Рис. 7.3

переключатели **1, 2, ..., 7**, подавая входные десятичные комбинации 3, 7, 15, 31, 63, 127, 255 на входы **D0, ..., D7** ЦАП (рис. 7.3, б).

Повторить эксперимент, подавая на входы ЦАП сформированные с помощью переключателей шестнадцатеричные коды от 0 до FF (255_{10}) через шаг 10_{16} (16_{10}) и занося в табл. 7.1 показания вольтметра **V1** (значения выходного напряжения $u_{вых}$ ЦАП) при напряжении источника **VCC** $u_o = 5$ В. **Найти** частичные и усредненное значение ступени, частичные и усреднённое значение МЗР. **Построить** график $u_{вых}(N)$, выбрав соответствующие масштабы для напряжений и входных десятичных чисел N , откладываемых по осям координат.

Т а б л и ц а 7.1

| № п/п | Входной десятичный код N | Выходное напряжение, $u_{вых}$, В | Напряжение ступени $u_{вых2} - u_{вых1}$, В | Значение младшего разряда МЗР = $(u_{вых2} - u_{вых1})/16$, В |
|----------|----------------------------------|--|--|---|
| 1 | 0 | 0 | 0 | — |
| 2 | 15 | | | |
| 3 | 31 | | | |
| 4 | 47 | | | |
| 5 | 63 | | | |
| 6 | 79 | | | |
| 7 | 95 | | | |
| 8 | 111 | | | |
| 9 | 127 | | | |
| 10 | 143 | | | |
| 11 | 159 | | | |
| 12 | 175 | | | |
| 13 | 191 | | | |
| 14 | 207 | | | |
| 15 | 223 | | | |
| 16 | 239 | | | |
| 17 | 255 | | | |

Задание 3. Открыть файл **7.4.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания *цифроаналогового преобразователя* (рис. 7.4, а) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 7.4, а) на страницу отчёта.

Провести моделирование ЦАП, запрограммировав генератор **XWG1** (частота генерации сигналов $f_z = 1$ кГц) на возрастание и убывание шестнадцатеричных чисел от 0 до FF (255_{10}) при шаге 10_{16} (16_{10}).

Составить таблицу и **занести** в неё выходные напряжения ЦАП и величину ступеней, которые выводятся в нижнем окне осциллографа **XSC2**.

Измерение напряжений **проводить** с помощью визирных линий осциллографа, устанавливая их на двух соседних ступенях (см. рис. 7.4, б) при различных кодовых комбинациях на выходе генератора **XWG1** и напряжении $u_o = 5$ В источника **VCC**.

Так, при входных десятичных числах 175 и 191 и напряжении $u_o = 5$ В выходные напряжения ЦАП соответственно равны 3,437 В и 3,750 В, а напряжение ступени – 312,5 мВ. При этом $ЗМР = 312,5/16 = 19,53$ В. **Найти** и

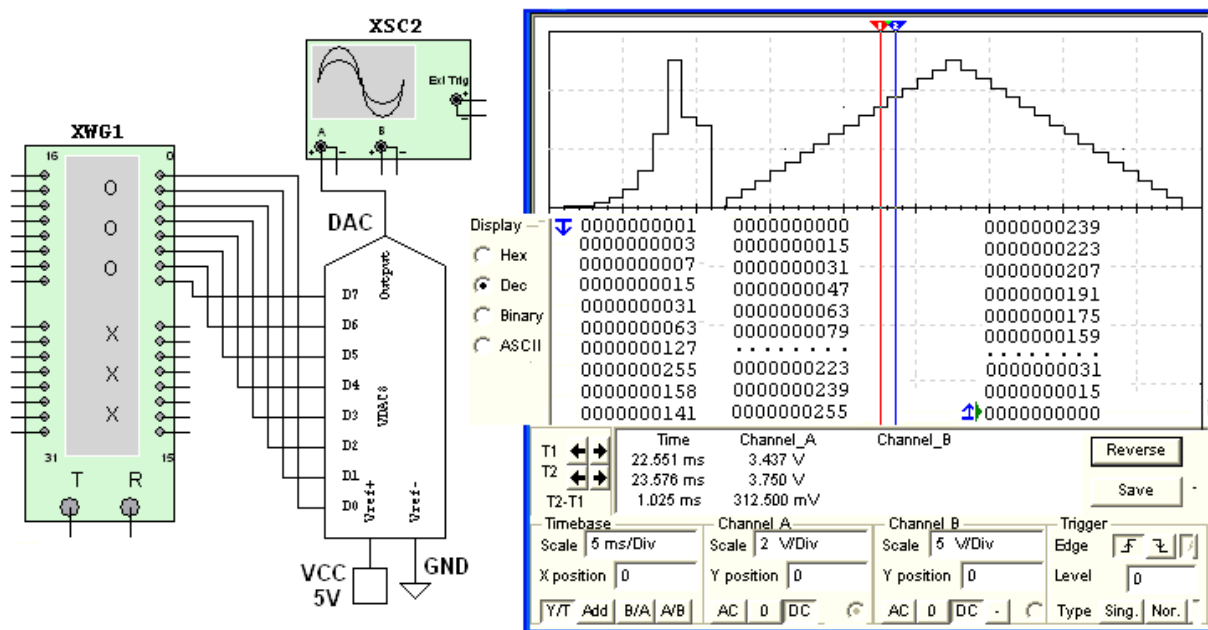


Рис. 7.4

сравнить усреднённое значение ЗМР с расчётным значением.

Установить напряжение $u_o = 10$ В источника VCC и **повторить** моделирование ЦАП при опорном напряжении 10 В. **Построить** графики $u_{вых}(N)$ при $u_o = 5$ В и $u_o = 10$ В на одном рисунке, выбрав соответствующие масштабы для напряжений и входных десятичных чисел N , откладываемых по осям координат.

СОДЕРЖАНИЕ ОТЧЁТА

1. Наименование и цель работы.
2. Перечень приборов, использованных в экспериментах, с их краткими характеристиками.
3. Изображения электрических схем для испытания цифроаналогового преобразователя.
4. Копии графиков выходного напряжения исследуемого ЦАП, отображающих его работу.
5. Графики $u_{вых}(N)$ при различных значениях опорного напряжения.
6. Выводы по работе.

ТЕСТОВЫЕ ЗАДАНИЯ К РАБОТЕ 7

1. Укажите **назначение** ЦАП.

- Для преобразования информации в аналоговой форме в цифровые коды
- Для преобразования цифрового кода N в пропорциональное аналоговое значение напряжения $u(N)$
- Для деления числа или частоты повторения импульсов на заданный коэффициент K
- Для преобразования информации из последовательной во времени формы представления в параллельную форму

2. Укажите, какая **структура резистивных матриц** ЦАП имеет преимущество при изготовлении преобразователя посредством интегральной технологии?

- ☐ Матрица с весовыми резисторами
- ☐ При изготовлении ЦАП с помощью интегральной технологии структура матриц не играет существенного значения, так как высокая точность и быстродействие систем код-напряжение зависят от типа переключателей (ключей) во входной разрядной цепи
- ☐ Матрица $R-2R$

3. Определите понятие "**абсолютная разрешающая способность**" ЦАП.

- ☐ Это возможное количество уровней аналогового сигнала, делённое на количество двоичных разрядов входного кода
- ☐ Это наибольшее значение отклонения аналогового сигнала от расчётного.
- ☐ Это максимальное отклонение ступенчато нарастающего выходного сигнала от прямой линии, соединяющей точки нуля и максимального выходного сигнала
- ☐ Это среднее значение минимального изменения сигнала на выходе ЦАП, обусловленное увеличением или уменьшением его кода на единицу

4. Укажите, для чего выбирают опорное напряжение **двуполярным**?

- ☐ Чтобы преобразовать двоичные коды в ток
- ☐ Для обеспечения работы ЦАП, содержащего резистивную матрицу с весовыми резисторами, диодные ключи и систему управления ключами
- ☐ Для увеличения диапазона $\pm u_{вых}$ выходного напряжения
- ☐ Чтобы получать на выходе двуполярное напряжение $\pm u_{вых}$ при различных входных кодах
- ☐ Чтобы максимальное выходное напряжение ЦАП не было меньше опорного напряжения u_o на величину ЗМР (ЗМР – значение младшего разряда)

5. Укажите **перспективы развития** ЦАП.

- ☐ Повышение быстродействия ключей и уменьшение времени установки ОУ
- ☐ Построение ЦАП без резистивной матрицы
- ☐ Применение стабилизированных источников опорного напряжения
- ☐ Уменьшение разрядности преобразователя код-напряжение (до 4...6)
- ☐ Улучшение качества резистивных матриц

Лабораторная работа 8 (Lr8)

АНАЛОГО-ЦИФРОВОЙ ПРЕОБРАЗОВАТЕЛЬ

ЦЕЛЬ РАБОТЫ

Ознакомление с принципом работы и испытание интегрального 8-разрядного аналого-цифрового преобразователя.

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ И РАСЧЁТНЫЕ ФОРМУЛЫ

1. СТРУКТУРНАЯ СХЕМА АЦП ПОСЛЕДОВАТЕЛЬНОГО ДЕЙСТВИЯ

Аналого-цифровой преобразователь (АЦП) – устройство, предназначенное для преобразования аналоговых величин в их цифровой эквивалент в различных системах исчисления. Входным сигналом АЦП в течение некоторого промежутка времени Δt является постоянное напряжение, равное отсчёту $u_{ex}(k\Delta t)$ входной аналоговой функции u_{ex} . За это время на выходе АЦП формируется цифровой (обычно двоичный) код

$$A_i(a_{n-1}a_{n-2}\dots a_1a_0),$$

соответствующий дискретному отсчёту напряжения $u_{ex}(k\Delta t)$. Количественная связь для любого момента времени определяется соотношением

$$A_i = u_{ex}(k\Delta t) / \Delta u \pm \delta_i,$$

где Δu – шаг квантования входного аналогового напряжения u_{ex} ; δ_i – погрешность преобразования напряжения $u_{ex}(k\Delta t)$ на данном шаге.

Физический процесс аналого-цифрового преобразования состоит из дискретизации по времени аналогового сигнала, квантования по уровню и кодирования [8]. Процесс *дискретизации* аналогового сигнала длительностью t_{ex} выполняется в соответствии с теоремой Котельникова, определяющей необходимый шаг дискретизации $\Delta t \leq 1/(2f_m)$, где f_m – максимальная частота спектра входного сигнала, и число шагов $M = t_{ex}/\Delta t$.

Процесс *квантования по уровню* дискретизированной функции $u_{ex}(k\Delta t)$ заключается в отображении бесконечного множества её значений на некоторое множество конечных значений $u_{\partial}(k)$, равное числу уровней квантования $N = u_{ex.max}/\Delta u$. Процесс квантования по уровню (округление каждого значения $u_{ex}(k\Delta t)$ до ближайшего уровня $u_{\partial}(k)$) приводит к возникновению ошибки (шума) квантования, максимальное значение которой $\pm 1/2\Delta u$ определяется разрядностью используемого выходного кода. При увеличении разрядности выходного кода ошибка квантования может быть уменьшена до сколь угодно малой величины, но не может быть сведена к нулю выбором параметров устройства, так как она присуща данному алгоритму.

Процесс *кодирования* заключается в замене найденных квантованных $N + 1$ значений входного сигнала $u_{\partial}(k)$ некоторыми цифровыми кодами.

На рис. 8.1, а приведена характеристика идеального АЦП в нормированных единицах входного напряжения $u_{вх.н} = u_{вх}/u_{вх.мах}$. Кроме ошибки квантования, при оценке точности АЦП учитывают дополнительные погрешности: *инструментальную* (погрешность смещения нуля, вызывающей смещение пунктирной прямой L влево или вправо от начала координат, см. рис. 8.1, а) и *апертурную*, возникающую из-за несоответствия значения входного сигнала $u_{вх}(k)$ преобразованному цифровому коду A_i . Несоответствие возникает, если входной сигнал в течение интервала дискретизации Δt изменяется более чем на значение шага квантования Δu .

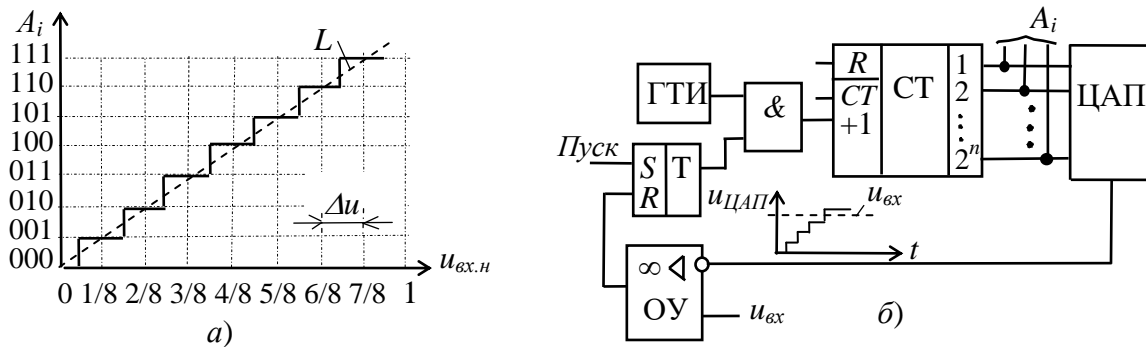


Рис. 8.1

2. ОСНОВНЫЕ ПАРАМЕТРЫ АЦП

К основным параметрам АЦП относят:

- число разрядов выходного кода $n = 8, \dots, 16$, отображающего исходную аналоговую величину, которое может формироваться на выходе АЦП. При использовании двоичного кода $n = \log_2(N + 1)$, где $N + 1$ – максимальное число кодовых комбинаций (уровней квантования) на выходе АЦП;
- диапазон изменения входного напряжения $u_{вх.мах}$. Отметим, что АЦП может обрабатывать входную информацию в виде однополярного аналогового напряжения с пределами $0 \dots u_{вх.мах}$ и двуполярного $\pm u_{вх.мах}/2$;
- абсолютная разрешающая способность $ЗМР = \Delta u$ (значение младшего разряда) – среднее значение минимального изменения входного сигнала $u_{вх}$, обуславливающего увеличение или уменьшение выходного кода на единицу. Значение ЗМР определяется разрядностью выходного кода и диапазоном входного напряжения;
- абсолютная погрешность δ_i преобразования в конечной точке шкалы есть отклонение реального максимального значения входного сигнала $u_{вх.мах}$ от максимального значения идеальной характеристики L АЦП (см. рис. 8.1, а). Обычно δ_i измеряется в ЗМР;
- максимальная частота преобразования (десятки и сотни кГц);
- время преобразования входного сигнала: $t_{пр.мах} \leq (1/2)\Delta t$.

Состав АЦП в отличие от ЦАП может изменяться в значительной степени в зависимости от выбранного метода преобразования и способа его реализации.

Наибольшее распространение получили три основных метода: последовательного счёта, поразрядного кодирования и считывания.

Метод последовательного счёта основан на уравнивании входной величины суммой одинаковых по величине эталонов (суммой шагов квантования). Момент уравнивания определяется с помощью одного компаратора, а количество эталонов, уравнивающих входную величину, подсчитывается с помощью счётчика.

Метод поразрядного кодирования (уравнивания) предусматривает наличие нескольких эталонов (часто реализованных в виде уравнивающего сдвигающего регистра), обычно пропорциональных по величине степеням числа 2, и сравнение этих эталонов с аналоговой величиной. Сравнение начинается с эталона старшего разряда. В зависимости от результата этого сравнения формируется значение старшего разряда выходного кода. Если эталон больше входной величины, то в старшем разряде ставится 0 и далее производится уравнивание входной величины следующим по значению эталоном. Если эталон равен или меньше входной величины, то в старшем разряде выходного кода ставится 1 и в дальнейшем производится уравнивание разности между входной величиной и первым эталоном.

Наибольшим быстродействием обладают преобразователи, построенные по методу считывания. *Метод считывания* подразумевает наличие $2^n - 1$ эталонов при n -разрядном двоичном коде. Входная аналоговая величина одновременно сравнивается со всеми эталонами. В результате преобразования получается параллельный код в виде логических сигналов на выходах $2^n - 1$ компараторов.

3. ВАРИАНТ РЕАЛИЗАЦИИ АЦП ПОСЛЕДОВАТЕЛЬНОГО СЧЁТА

В качестве примера рассмотрим структурную схему АЦП последовательного счёта с ЦАП в цепи обратной связи (рис. 8.1, б) и вариант её реализации (рис. 8.2). По сигналу "Пуск" на вход обнуленного счётчика СТ начинают подаваться импульсы генератора тактовой частоты ГТИ (см. рис. 8.1, б). По мере поступления этих импульсов растёт входной код ЦАП и ступенчато повышается напряжение $u_{ц\alpha\pi}$ на его выходе, причем уровень ступени соответствует шагу квантования Δu входного напряжения $u_{вх}$ АЦП.

Процесс преобразования заканчивается, когда напряжение $u_{ц\alpha\pi}$ станет чуть больше входного напряжения $u_{вх}$ АЦП, поданного на вход ОУ, на котором собран компаратор. При этом работа счётчика прекращается, а на его выходе устанавливается код A_i , являющийся цифровым эквивалентом напряжения $u_{вх}$.

Согласно рассмотренной структурной схеме АЦП на рис. 8.2 приведен вариант реализации модели 4-разрядного АЦП последовательного счёта с ЦАП, состоящего из операционного усилителя **OPAMP1** и резистивной матрицы **R1**, ..., **R4** со взвешенными сопротивлениями. Переключатели **Key1**, ..., **Key4** в схеме (при разомкнутом ключе **Space**) служат для проверки работы счётчика СТ, а осциллограф **XSC1** – для снятия осциллограмм напряжения с выхода ЦАП и входа компаратора.

При запуске моделирования АЦП сформированные генератором **E1** импульсы подаются на вход счётчика **СТ**, число которых последовательно высвечивается на 7-сегментном индикаторе. Выходные поразрядные сигналы со счётчика поступают также на входы логического анализатора **XLF1** и входы резистивной матрицы **R1**, ..., **R4**, а суммарное напряжение с матрицы – на вход ОУ. Ступенчатое напряжение $u_{ЦАП}$ с выхода **OPAMP1** (рис. 8.3) подаётся на вход компаратора, собранного на операционном усилителе **OPAMP2**. На этот же вход подано постоянное напряжение $u_{вх}$ с генератора **E7** через делитель **R6-R7**. В момент, когда указанные напряжения сравниваются, компаратор срабатывает, на элемент И (**AND**) подаётся логический 0 и прекращается работа счётчика, а на индикаторе высвечивается цифровой код (число шагов квантования),

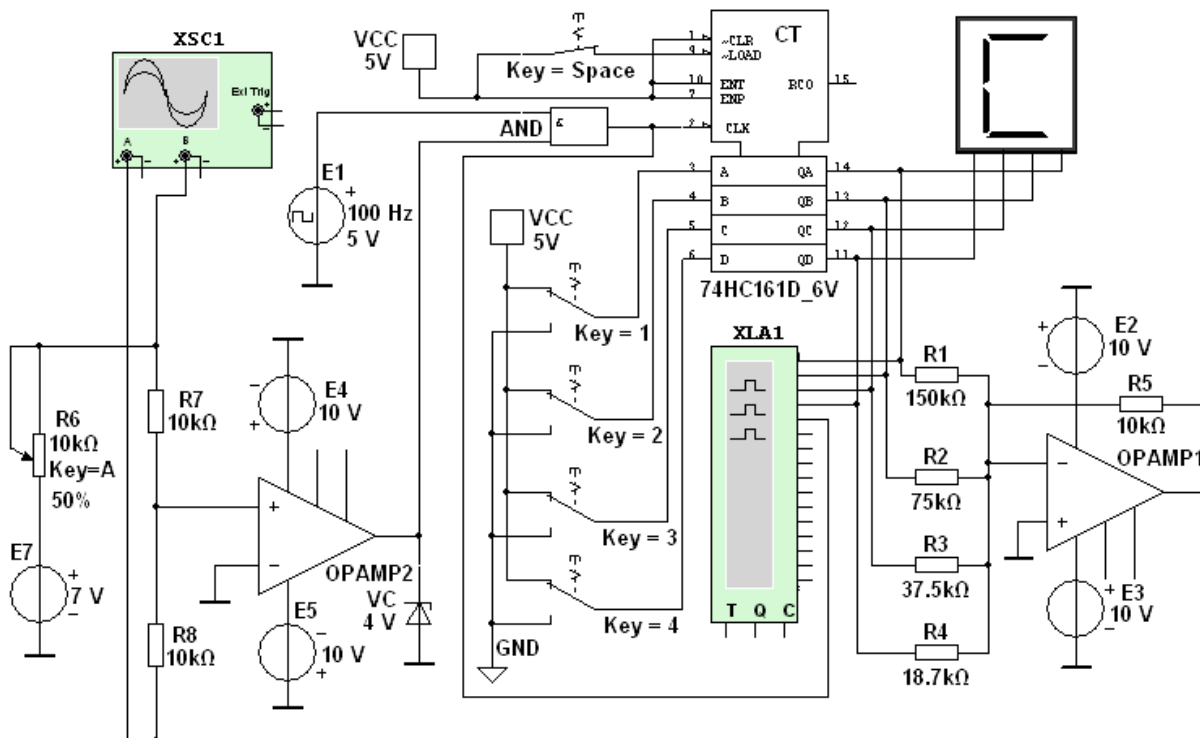


Рис. 8.2

соответствующий уровню $u_d(k)$.

Анализ временных диаграмм сигналов с выхода счётчика и осциллограмм напряжений с входов компаратора (см. рис. 8.3) показывает, что счётчик прервал счёт с приходом двенадцатого тактового импульса, поэтому на 7-сегментном индикаторе высветилось число C_{16} (12_2) (см. рис. 8.2).

Погрешность преобразования зависит от шага квантования (высоты ступени напряжения $u_{цан}$), погрешности в формировании ступенчатого напряжения $u_{цан}$ и ошибки компаратора в определении равенства $u_{вх}$ и $u_{цан}$. Время преобразования непостоянно и зависит от уровня напряжения $u_{вх}$. При заданном числе разрядов АЦП время преобразования определяется числом периодов счетных импульсов.

По структуре построения ИМС АЦП подразделяют на АЦП с применением ЦАП и без них. К БИС АЦП без ЦАП, например ИМС КР572ПВ2, К107ПВ2 и

др., относят АЦП последовательного счёта с двойным интегрированием (на первом такте – входного напряжения, на втором – эталонного напряжения с преобразованием результатов интегрирования во временной интервал и в эквивалентный цифровой код) для сглаживания импульсных помех, повышения точности и помехозащищённости данного типа АЦП [12].

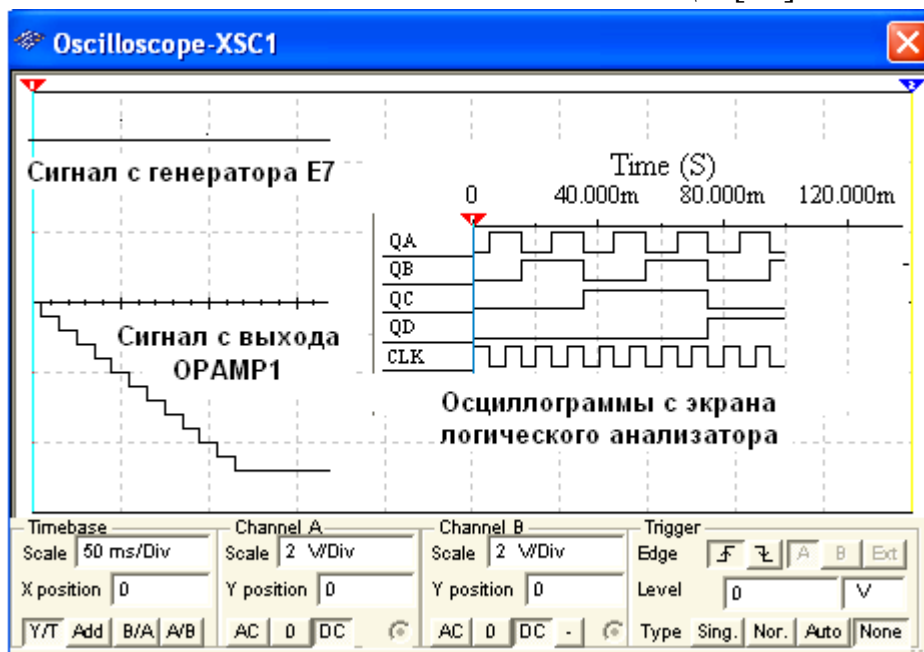


Рис. 8.3

Платы АЦП/ЦАП, например, модели LTC российской компании ЗАО "Л-КАРД", широко применяют в цифровых измерительных приборах, в системах и устройствах обработки и отображения информации, в автоматических системах контроля и управления, в устройствах ввода-вывода информации ЭВМ и т. д. Основные направления развития АЦП – повышение быстродействия основных узлов, в частности, компараторов до 5...10 нс, повышение их точности до 0,05...0,005%, увеличение разрядности преобразователей до 24, использование микропроцессоров в преобразователях.

УЧЕБНЫЕ ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ИХ ВЫПОЛНЕНИЮ

Задание 1. Запустить лабораторный комплекс Labworks и среду MS10 (щёлкнув мышью на команде **Эксперимент** меню комплекса Labworks). Открыть файл **8.4.ms10**, размещённый в папке **Circuit Design Suite 10.0** среды MS10, или **собрать** на рабочем поле среды MS10 схему для испытания *аналого-цифрового преобразователя* с ЦАП (рис. 8.4) и **установить** в диалоговых окнах компонентов их параметры или режимы работы. **Скопировать** схему (рис. 8.4) на страницу отчёта.

В схему (рис. 8.4) включены собственно библиотечный 8-разрядный АЦП (ADC); источники опорного напряжения **E1** и **E2** (подключены к входам **Vref+** и **Vref-** АЦП); генератор **E4** для синхронизации работы (подключен к входу **SOC**) и разрешения (вход **OE**) на выдачу двоичной информации на выходы **D0**,

..., **D7** АЦП, с которыми соединены входы логического анализатора **XLA1** и пробники **X0**, ..., **X7**; функциональный генератор **XFG1** в качестве источника входного сигнала $u_{вх}$ (подключен к входу **Vin**); ЦАП (**DAC**) и осциллограф **XSC1**. Выход **EOC** служит для передачи двоичной информации АЦП, например,

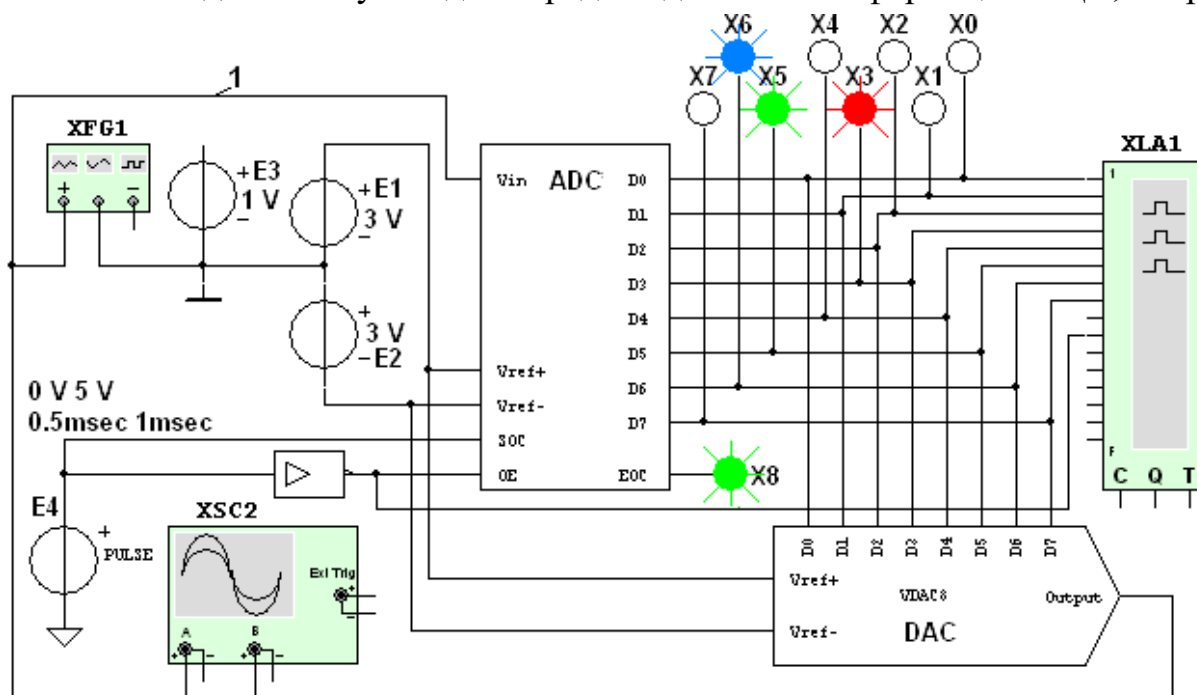


Рис. 8.4

на ЭВМ.

Задание 2. Исследовать точность преобразования АЦП уровней входного напряжения $u_{вх}$ в цифровой код с помощью пробников **X0**, ..., **X7**, логического анализатора **XLA1**, а также ЦАП и осциллографа **XSC1**.

С этой целью:

- временно **удалить** провод 1 (см. рис. 8.4) и подключить вход **Vin** АЦП к положительному полюсу источника постоянного напряжения **E3**;
- **составить** таблицу, аналогичную табл. 8.1, в первый столбец которой записать уровни напряжения

$u_{вх} = 0,1; 0,2; 0,5; 1,0; 1,5; 2,0; 2,4; -0,5; -1,0; -2,0$ В,

поочерёдно задаваемые в диалоговом окне генератора **E3**;

- **установить** в диалоговых окнах генераторов **E1** и **E2** ЭДС $E_1 = 2,5$ В, и ЭДС $E_2 = -2,5$ В;

– **запустить** программу моделирования АЦП и **вносить** в поля составленной таблицы значения напряжения $u_{вых(ЦАП)}$ с выхода ЦАП, измеряемые на экране осциллографа с помощью визирной линии; двоичный эквивалент $D_{(2)}$ преобразуемого напряжения, определяемый по свечению пробников **X7**, ..., **X0**; шестнадцатеричный код $D_{(16)}$, считываемый с дисплея анализатора **XLA1**;

- получаемые с выхода АЦП десятичные инверсные сигналы $D_{(10)инв}$ **пересчитать** на неинверсные $D_{(10)}$ по выражению

$$D_{(10)} = D_{(10)инв} - 128$$

и занести в соответствующие столбцы таблицы;

– расчётные десятичные эквиваленты $D_{(10)расч}$ двоичного кода $D_{(2)}$ на выходе АЦП при заданном значении входного напряжения $u_{вх}$ **определить** по формуле

$$D_{(10)расч} = 256u_{вх} / (E_1 + |-E_2|),$$

и занести во второй справа столбец таблицы;

– **рассчитать** погрешности измерения напряжения по выражению

$$\Delta U\% = 100(u_{вых(ЦАП)} - u_{вх})/u_{вх}$$

и занести в правый столбец таблицы.

В качестве примера в табл. 8.1 приведены данные измерений при моделирования АЦП при $E_1 = 3$ В и $E_2 = -3$ В, которые близки к расчётным значениям. Так, при $E_1 = |E_2| = 3$ В и $u_{вх} = E_3 = 1$ В расчётный десятичный эквивалент $D_{(10)расч} = 256 \cdot 1/6 \approx 42,67$ при измеренном $D_{(2)} = 10101010$ и $D_{(10)} = 42$. При этом погрешность измерения составила 3,56%.

Т а б л и ц а 8.1

| $u_{вх}$, В | $u_{вых(ЦАП)}$, В | $D_{(2)}$ | $D_{(16)}$ | $D_{(10)инв}$ | $D_{(10)}$ | $D_{(10)расч}$ | $\Delta U\%$ |
|-----------------|-----------------------|-----------|------------|---------------|------------|----------------|--------------|
| 0,1 | 0,09375 | 10000100 | 84 | 132 | 4 | 4,27 | 6,25 |
| 0,5 | 0,5156 | 10010101 | 95 | 149 | 21 | 21,33 | 3,12 |
| 1,0 | 0,9644 | 10101010 | AA | 170 | 42 | 42,67 | 3,56 |
| 2,0 | 2,017 | 11010101 | D5 | 213 | 85 | 85,34 | 0,85 |
| 2,5 | 2,484 | 11101010 | EA | 234 | 106 | 106,67 | 0,64 |
| 2,9 | 2,906 | 11111011 | FB | 251 | 123 | 123,74 | 0,21 |
| -1,0 | -0,9844 | 01010101 | 55 | 85 | -43 | -42,67 | 3,56 |

Задание 3. Исследовать процесс преобразования входного напряжения треугольной формы в цифровые коды, а затем с помощью ЦАП – в ступенчатое напряжение, аппроксимирующее напряжение $u_{вх}$.

Для этого:

- **удалить** провод, соединяющий выход генератора **E3** с входом **Vin** АЦП, и **восстановить** провод 1, соединяющий выход "+" функционального генератора **XFG1** с входом **Vin** АЦП (см. рис. 8.4);
- **установить** параметры генератора **XFG1** (рис. 8.5, а): напряжение треугольной формы со скважностью $N = 99$ и амплитудой 1 В (диапазон от –1 В до 0,98 В) и его частоту $f_2 = 50$ Гц;
- **запустить** программу моделирования АЦП;

– **получить и скопировать** на страницу отчета осциллограмму входного напряжения $u_{вх}$, осциллограмму ступенчатого напряжения $u_{вых(ЦАП)}$ с выхода ЦАП (см. рис. 8.5, б), и временные диаграммы сигналов с выходов **D0**, ..., **D7** АЦП, поступающих на входы логического анализатора **XLA1** и являющимися двоичными эквивалентами дискретных отсчётов $u_{вх}(k\Delta t)$ входного напряжения

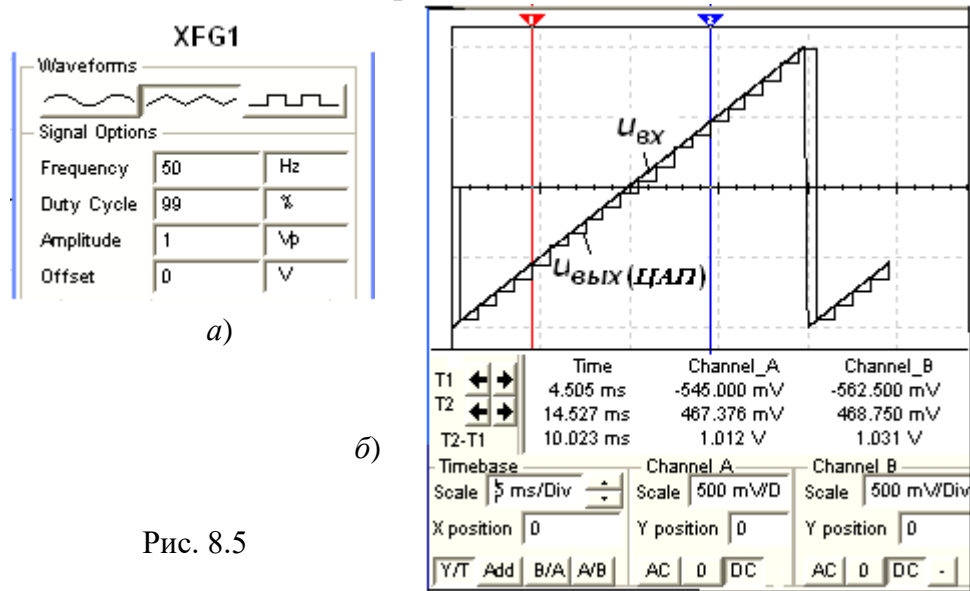


Рис. 8.5

(рис. 8.6);

– воспользовавшись визирными линиями, **провести анализ** формирования напряжения $u_{вых(ЦАП)}$, аппроксимирующего входное напряжение $u_{вх}$, в частности, **измерить** напряжение и высоту его ступеней в разные моменты преобразования (с интервалом в 1 мс в моменты положительного перепада тактового импульса синхронизации) и **сравнить** их с отсчётами $u_{вх}(k\Delta t)$ напряжения $u_{вх}$.

Так, при частоте синхронизации $f_c = 1$ кГц и частоте пилообразного напряжения $f_z = 50$ Гц образовалось на выходе ЦАП двадцать ступеней напряжения $u_{вых(ЦАП)}$, средняя высота которых равна $U_{см} \approx 93,7$ мВ при расчётном значении $\Delta u = u_{вх.max}/(N + 1) = 1,98/21 = 94$ мВ. Первая ступень высотой 66 мВ сформировалась по истечении 0,5 мс с момента включения моделирования при уровне входного напряжения $u_{вх} = -93,4$ мВ, вторая – при $u_{вх} = -0,849$ В высотой 93,75 мкВ и и т. д.

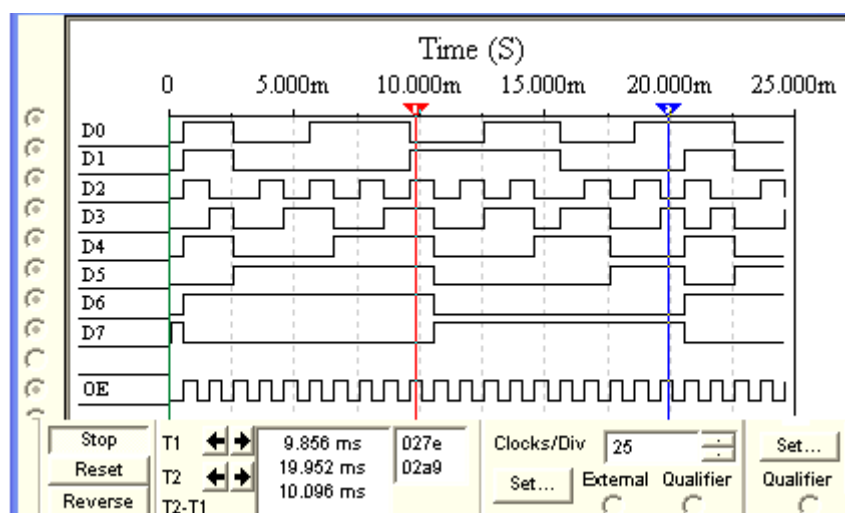


Рис. 8.6

Задание 4 (выполняется факультативно или по указанию преподавателя). Исследовать процесс преобразования АЦП входного синусоидального напряжения в цифровые коды, а затем с помощью ЦАП – в ступенчатое напряжение.

С этой целью:

– щёлкнуть мышью на кнопке "Синусоидальное напряжение" генератора XFG1 (см. рис. 8.5, а) и установить частоту напряжения $f_2 = 25$ Гц, а затем, при остановке моделирования, $f_2 = 5$ Гц с изменением времени развёртки лучей осциллографа с 10 мс/дел на 50 мс/дел. Сместить вверх на 0,6 деления осциллограмму входного напряжения $u_{вх}$ (рис. 8.7);

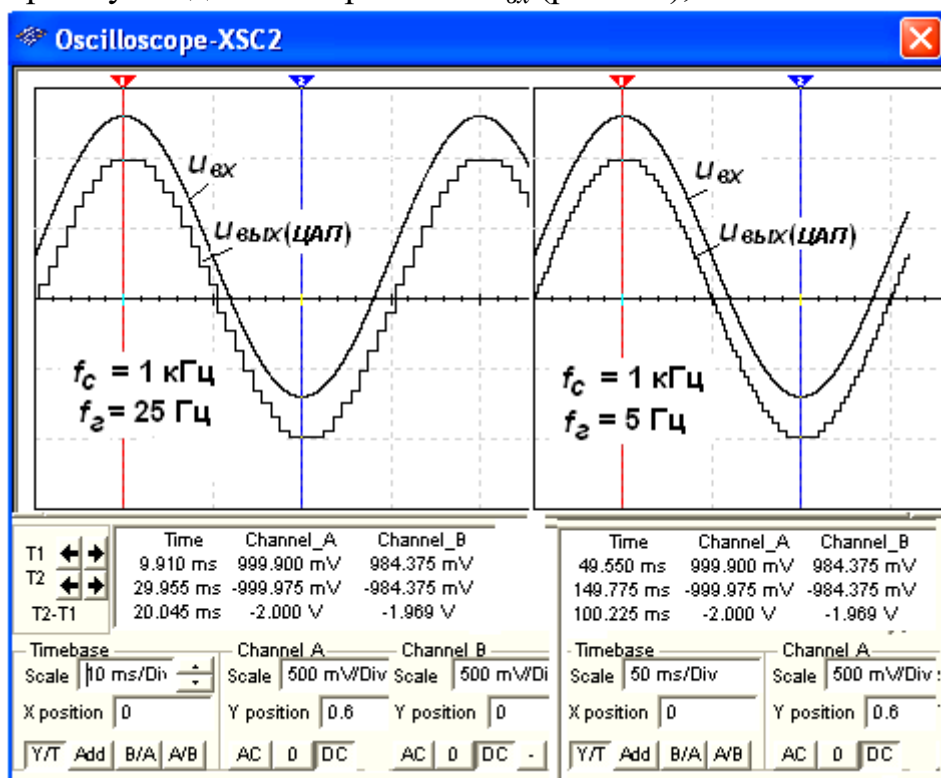


Рис. 8.7

– измерить напряжение $u_{вых(ЦАП)}$ и высоту его ступеней в разные моменты преобразования и сравнить их с отсчётами напряжения $u_{вх}(k\Delta t)$ входного напряжения $u_{вх}$ для моментов положительного перепада тактового импульса синхронизации.

Двоичные эквиваленты отсчетов напряжения $u_{вх}(k\Delta t)$ с выходов АЦП преобразуются с помощью ЦАП в аналоговый ступенчатый сигнал $u_{вых(ЦАП)}$ (см. рис. 8.7). При этом с уменьшением частоты сигнала увеличивается число ступеней и преобразованная кривая хорошо аппроксимирует входной сигнал. Высота ступеней переменная, от 46 мВ до 141 мВ, так как интервал дискретизации Δt при заданной частоте синхронизации постоянный. Особенно заметна верхняя и нижняя ступени с отклонением от амплитуды входного напряжения приближённо на 15,5 мВ, так как на интервалах дискретизации около амплитуд скорость изменения напряжения минимальная.

СОДЕРЖАНИЕ ОТЧЁТА

1. Наименование и цель работы.
2. Перечень приборов, использованных в экспериментах, с их краткими характеристиками.
3. Изображение электрической схемы для испытания аналого-цифрового преобразователя.
4. Копии осциллограмм и временных диаграмм сигналов с разных узлов схемы, отображающие работу исследуемого АЦП.
5. Таблица с результатами измерений и расчётов входных отсчетов входного напряжения и выходных кодов АЦП.
6. Выводы по работе.

ТЕСТОВЫЕ ЗАДАНИЯ К РАБОТЕ 8

1. Укажите **назначение** АЦП.

- ☐ Для преобразования кодов
- ☐ Для преобразования цифрового кода N в пропорциональное аналоговое значение напряжения $u(N)$
- ☐ Для преобразования постоянного напряжения, заданного на тактовом интервале, в двоичный код
- ☐ Для преобразования информации из последовательной во времени формы представления в параллельную форму

2. Укажите **формулу Котельникова**, с помощью которой определяют шаг дискретизации Δt аналогового сигнала.

- ☐ $\Delta t \leq 1/2f_m$
- ☐ $\Delta t \leq 1/f_m$
- ☐ $\Delta t \leq t_{ex}/2^{N+1}$
- ☐ $\Delta t \leq t_{ex}/2^{N-2}$

(f_m – максимальная частота спектра аналогового сигнала; t_{ex} – длительность аналогового сигнала; N – число уровней квантования)

3. Определите понятие "**абсолютная разрешающая способность**" АЦП.

- ☐ Это число уровней квантования, делённое на количество разрядов выходного кода
- ☐ Это наибольшее значение отклонения аналогового сигнала от расчётного
- ☐ Это среднее значение минимального изменения входного сигнала, обуславливающего увеличение или уменьшение выходного кода на единицу
- ☐ Это время преобразования отсчёта входного сигнала

4. Укажите, можно ли подавать на входы V_{ref+} и V_{ref-} АЦП **разные** (по модулю) **напряжения**?

- ☐ Да
- ☐ Нет

5. Укажите, можно ли **свести к нулю** погрешность квантования аналогового сигнала посредством выбора параметров устройства, например за счёт увеличения разрядности АЦП?

- ☐ Да
- ☐ Нет

6. Укажите, какую **погрешность** квантования имеет 8-разрядный АЦП при напряжениях на входах $V_{ref+} = 2$ В, $V_{ref-} = 0$ и отсчёте входного напряжения $u_{ex}(k\Delta t) = 1$ В?

- ☐ $\pm 4,15$ мВ
- ☐ $\pm 3,91$ мВ
- ☐ $\pm 3,15$ мВ
- ☐ $\pm 2,25$ мВ
- ☐ $\pm 1,95$ мВ

7. Укажите **десятичный эквивалент** двоичного кода на выходе 8-разрядного АЦП, если опорные напряжения $V_{\text{ref}+} = 2 \text{ В}$, $V_{\text{ref}-} = -2 \text{ В}$, а входное напряжение $u_{\text{вх}} = 0,5 \text{ В}$.

- ☐ 48 ☐ 32 ☐ 16 ☐ 8

8. Выберите из приведенных ниже значений минимально необходимые **значения опорных напряжений** $\pm V_{\text{ref}}$ для преобразования синусоидального напряжения $u_{\text{вх}}(t) = 1,41 \sin \omega t$.

- ☐ $\pm 1 \text{ В}$ ☐ $\pm 2 \text{ В}$ ☐ $\pm 3 \text{ В}$ ☐ $\pm 4 \text{ В}$ ☐ $\pm 5 \text{ В}$

9. Укажите значение расчётного **шестнадцатеричного кода** 16-разрядного АЦП, если на его вход подано напряжение $u_{\text{вх}}(k\Delta t) = 0,25 \text{ В}$ при $\pm V_{\text{ref}} = \pm 2 \text{ В}$.

- ☐ 1000 ☐ FFF ☐ 10000 ☐ FFFF ☐ FFA

10. Укажите **выражение**, с помощью которого определяют десятичный эквивалент двоичного кода на выходе 14-разрядного АЦП

- ☐ $D = 256u_{\text{вх}}/(V_{\text{ref}+} + |-V_{\text{ref}-}|)$ ☐ $D = 16384u_{\text{вх}}/(V_{\text{ref}+} + |-V_{\text{ref}-}|)$
☐ $D = 4096u_{\text{вх}}/(V_{\text{ref}+} + |-V_{\text{ref}-}|)$ ☐ $D = 65536u_{\text{вх}}/(V_{\text{ref}+} + |-V_{\text{ref}-}|)$

11. Укажите, как изменится **выходной код** АЦП при неизменном входном $u_{\text{вх}}$ и опорных напряжениях $V_{\text{ref}+} = 2 \text{ В}$ и $V_{\text{ref}-} = -2 \text{ В}$, если установить $V_{\text{ref}-} = 0$?

- ☐ Его значение уменьшится в 2 раза ☐ Не изменится
☐ Его значение увеличится в 2 раза ☐ Сменится на инверсный.

12. Укажите характер изменения **общей погрешности** преобразования входного сигнала при увеличении разрядности АЦП.

- ☐ Погрешность преобразования уменьшится ☐ Не изменится
☐ Погрешность преобразования увеличится ☐ Нет правильного ответа

13. Укажите перспективные **направления** развития АЦП.

- ☐ Повышение быстродействия основных узлов АЦП, в частности, компараторов
- ☐ Увеличение частоты генератора тактовых импульсов
- ☐ Применение стабилизированных источников опорного напряжения
- ☐ Уменьшение разрядности преобразователя напряжение-код (до 4...6)
- ☐ Использование микропроцессоров в преобразователях

14. Укажите, какие **операции** необходимо выполнить при аналого-цифровом преобразовании?

- ☐ Ограничение уровня и дискретизацию по времени аналогового сигнала
- ☐ Тактируемое интегрирование входного сигнала и сравнение полученного результата с эталонами
- ☐ Дискретизацию по времени аналогового сигнала, квантования по уровню его отсчётов и кодирование квантованных уровней
- ☐ Дискретизацию по времени аналогового сигнала, квантование по уровню для подачи на вход ЦАП

15. Укажите, обладает ли способ последовательного счёта аналого-цифрового преобразования наибольшим быстродействием?

- ☐ Да ☐ Нет