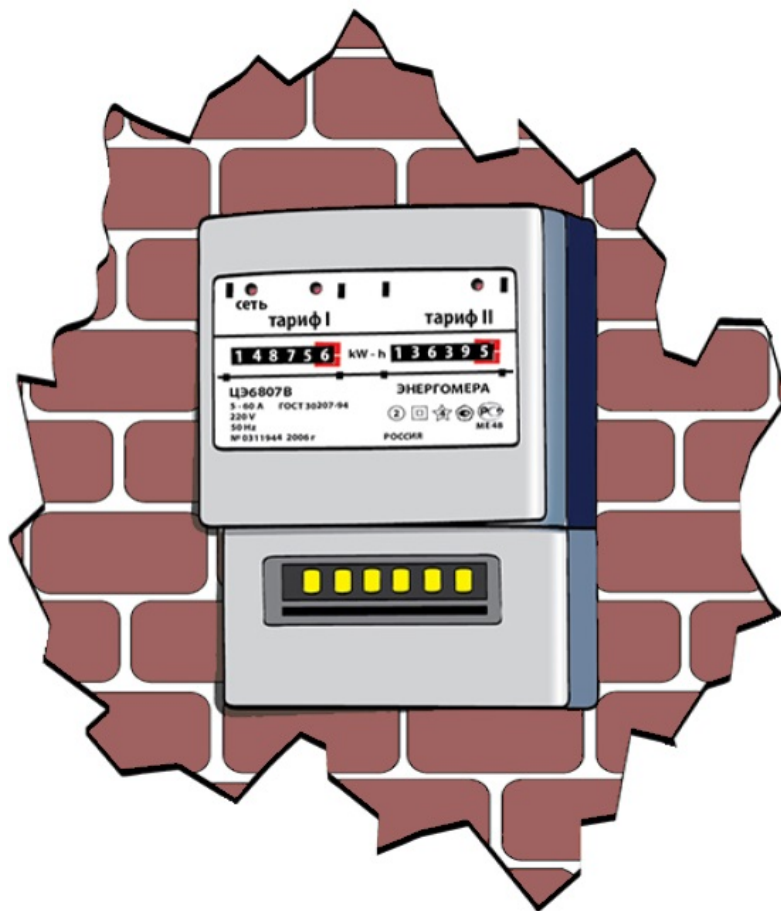


Китаев Ю.В.

ЛАБОРАТОРНЫЕ РАБОТЫ
“ЭЛЕМЕНТНАЯ БАЗА ЦИФРОВОЙ ЭЛЕКТРОТЕХНИКИ”

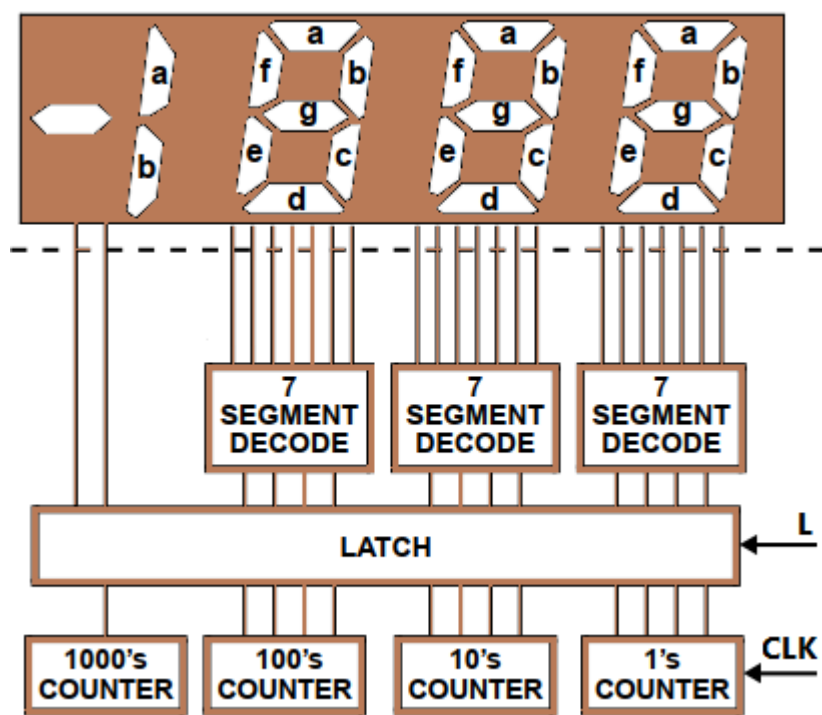


Санкт-Петербург

2022

ВВЕДЕНИЕ

Современные многофункциональные электроизмерительные приборы, например мультиметры выполняются с использованием специализированных микросхем основными компонентами, которых являются [аналого-цифровые преобразователи двойного интегрирования](#), двоично-десятичные счётчики импульсов, запоминающие регистры и преобразователи двоично-десятичного кода в семи-сегментный код. Ниже на рисунке приведена функциональная схема цифровой части микросхемы [ICL7107](#), входящей в состав мультиметров.



CLK - счётный вход двоично-десятичного счётчика (counter) для импульсов, число которых N пропорционально измеряемому напряжению,

L - импульс, переписывающий в регистр-защёлку (latch) результат текущего подсчёта, SEGMENT DECODE - преобразователи двоично-десятичного кода в семи-сегментный код.

Четыре представленных ниже лабораторных работы (20-ая вводная), позволяют не только познакомиться с принципом работы перечисленных устройств, но и произвести их расчёт.

ОГЛАВЛЕНИЕ

20. ГЕНЕРАТОР ИМПУЛЬСНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ ЗАДАННОЙ ФОРМЫ.....	3
21. СИНТЕЗ ПРЕОБРАЗОВАТЕЛЯ VIN (VCD) КОДА В СЕМИСЕГМЕНТНЫЙ.....	9
22. РАЗРАБОТКА СХЕМ НА РЕГИСТРАХ СДВИГА С ОБРАТНОЙ СВЯЗЬЮ.....	11
23. СИНТЕЗ СЧЕТЧИКА С ПРОИЗВОЛЬНЫМ МОДУЛЕМ СЧЕТА $M < 2^n$	15
НЕКОТОРЫЕ ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ.....	17
СПИСОК ЛИТЕРАТУРЫ.....	19

ЛАБОРАТОРНАЯ РАБОТА №20

РАЗРАБОТКА ГЕНЕРАТОРА ИМПУЛЬСНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ ЗАДАННОЙ ФОРМЫ

Цель работы: Изучение синтеза комбинационных схем.

Этапы проектирования.

1. Заданную последовательность разбиваем на ряд одинаковых интервалов N , на каждом из которых сигнал сохраняет постоянное значение.
2. Находим число независимых переменных: $n \geq \log_2 N$ (ближайшее большее целое число).
3. В качестве примера синтезируем импульсную последовательность Y , приведенную на рисунке 20.1 (высокий уровень Y соответствует логической "1", а низкий - "0").

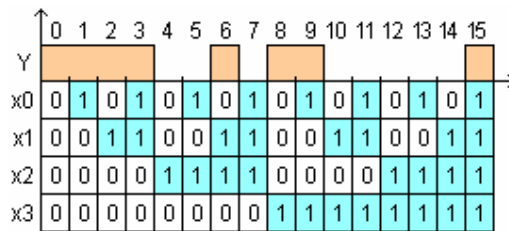


Рис. 20.1 Таблица истинности (временная диаграмма)

Приведенная последовательность (логическая функция Y), как видно из рисунка, разбивается на $N=16$ интервалов от 0 до 15-ти, причем число ее аргументов равно $n = \log_2 16 = 4$. Такое табличное задание логической функции (ЛФ) в зависимости от значений входных переменных называется таблицей истинности (ТИ).

4. Найдем алгебраическое выражение заданной ЛФ и заодно минимизируем ее с помощью таблицы Карно (ТК). Таблица Карно является разновидностью таблицы истинности (рис. 20.2).

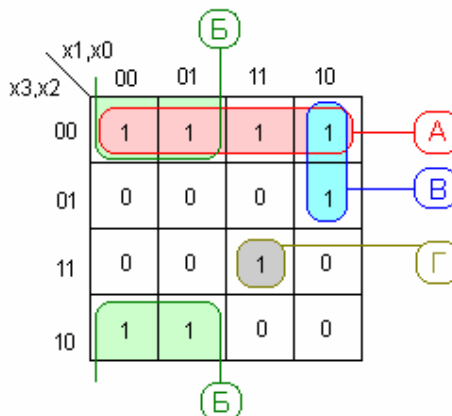


Рис. 20.2 Таблица Карно

4.1. Слева и сверху от ТК располагаются значения переменных x_3, x_2 и x_1, x_0 , причем соседние клетки ТК отличаются значением ТОЛЬКО одного аргумента. Отсюда следует, что противоположные клетки в ТК также являются соседними и таблицу

сначала можно "склеить" в цилиндр, а затем с помощью еще одной "склейки" превратить в непрерывную тороидальную поверхность ("сушку", как выразился один студент).

4.2. В клетки ТК переносим соответствующие значения ЛФ из таблицы истинности. Например $Y_0=1$ при $x_3x_2x_1x_0=0000$ из крайнего левого (нулевого) столбца таблицы истинности записываем в левую верхнюю клетку ТК, т.к. этой клетке соответствуют те же аргументы $x_3x_2x_1x_0=0000$. Значение $Y_4=0$ при $x_3x_2x_1x_0=0100$ из 4-го столбца переносим в клетку ТК с "координатами" $x_3x_2=01$ и $x_1x_0=00$ и так далее для всех остальных значений ЛФ.

ВНИМАНИЕ! В столбцах 0,1,2,3 таблицы истинности (рис.20.2) аргументы x_1, x_0 следуют в порядке 00,01,10,11, а в столбцах ТК в порядке 00,01,**11,10**. Аналогично обстоит дело и с остальными столбцами ТИ. Например $Y_4=0, Y_5=0, Y_6=1, Y_7=0$ записывается во вторую сверху строчку ТК в порядке 0001 ($Y_4=0, Y_5=0, Y_6=1, Y_7=0$). Можно сказать, что 2 правых столбца ТК переставлены местами. Точно также "переставлены местами" две нижних строчки ТК.

4.3. СОСЕДНИЕ клетки ТК с единичными значениями ЛФ объединяем в прямоугольники (импликанты). В прямоугольнике может быть **ТОЛЬКО** 2^k (т.е. 1, 2, 4, 8...) клеток. Чем больше клеток в прямоугольнике и чем меньше прямоугольников, тем проще результирующее алгебраическое выражение ЛФ. Прямоугольники могут перекрываться.

4.4. Для каждого прямоугольника записывается логическое произведение (логическое И) тех переменных, которые в соседних клетках не изменяют своего значения. Причем, если значение переменной в соседних клетках ТК равно единице, то переменная записывается в прямом виде. Если значение переменной равно нулю, то в - инверсном.

В нашем примере оптимальным решением является объединение 8-ми единичных клеток в 4-ре прямоугольника (А, Б, В и Г), как показано на рисунке. Например для прямоугольника "В" переменные x_1, x_0 входят в обе клетки без изменения, поэтому в логическом произведении они обе будут присутствовать - x_1 в прямом виде ($x_1=1$), а x_0 в - инверсном ($x_0=0$). В прямоугольнике "В" присутствуют также переменные x_3, x_2 , но x_2 в соседних клетках принимает различные значения, поэтому в произведение войдет только переменная x_3 в инверсном виде, т.к. $x_3=0$. Таким образом для прямоугольника "В" логическое произведение: $B = \sim x_3 * x_1 * \sim x_0$, где "~" обозначает инверсию.

Для прямоугольника "А" (фрагмент ТК приведен внизу) во всех парах соседних клеток обе переменные x_1 и x_0 изменяют значение, поэтому в произведение x_1 и x_0 не войдут. Наоборот, переменные x_3, x_2 во все 4-ре клетки входят без изменения и равны при этом нулю. Следовательно логическое произведение для прямоугольника "А" будет иметь вид: $A = \sim x_3 * \sim x_2$. На рисунке 20.2.1 соседние пары, меняющие значения выделены скобками.

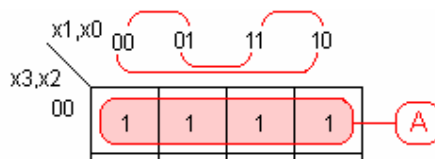


Рис. 20.2.1 Пример объединения клеток в ТК

В прямоугольнике "Б" в соседних колонках не меняет своего значения переменная x_1 , а в обоих строчках этого прямоугольника постоянно переменная x_2 . Учитывая, что значения обеих переменных равны нулю, в произведение они войдут в инверсном виде: $B = \sim x_2 * \sim x_1$.

Последний прямоугольник "Г" состоит из одной клетки, т.к. "единичных соседей" у него нет: $\Gamma = x_3 * x_2 * x_1 * x_0$ (все переменные в прямом виде, т.к. их значения равны 1).

4.5. Полученные логические произведения объединяются с помощью логического ИЛИ (или логического сложения) в искомую ЛФ в базисе "И-ИЛИ-НЕ":

$$Y = A + B + B + \Gamma = \overline{x_3} * \overline{x_2} + \overline{x_2} * \overline{x_1} + \overline{x_3} * x_1 * \overline{x_0} + x_3 * x_2 * x_1 * x_0 \quad (I)$$

Используя аксиому двойного отрицания и теорему двойственности получим выражение искомой ЛФ в базисе "И-НЕ":

$$Y = A + B + B + \Gamma = \overline{\overline{A + B + B + \Gamma}} = \overline{\overline{A} * \overline{B} * \overline{B} * \overline{\Gamma}} = \\ = \overline{\overline{x_3} * \overline{x_2} * \overline{x_2} * \overline{x_1} * \overline{x_3} * x_1 * \overline{x_0} * \overline{x_3} * x_2 * x_1 * x_0} \quad (II)$$

4.6. Схемная реализация полученных алгебраических выражений. На схемах логическое "И" обозначается знаком "&", логическое "ИЛИ" - знаком "1" и логическое отрицание (логическое "НЕ" или инверсия) обозначается кружком.

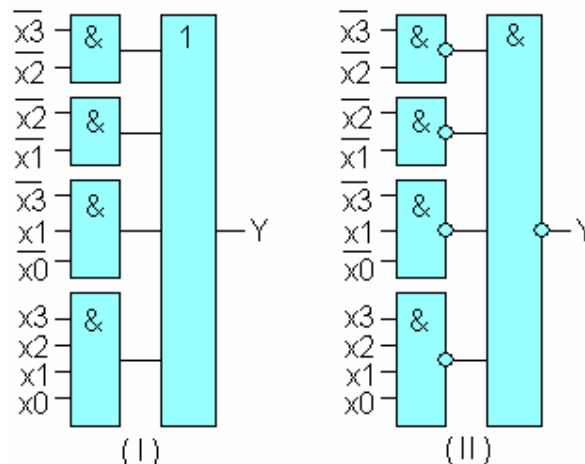



Рис. 20.2.2. Две реализации одной схемы

В схеме (II) на рисунке 20.2.2 используются только элементы "И-НЕ". На рисунке приведены обозначения логических элементов "И", "ИЛИ", "И-НЕ" в соответствии с ГОСТ 2.743-91.

Сборка и отладка схемы.

5. Запустите интегрированную среду разработки и отладки электронных схем Electronic Workbench (EW) . В открывшемся рабочем пространстве (рис. 20.2.3) переименуйте проект с обезличенного Untitled в XXXX_lab20, где XXXX номер вашей группы (например 8888_lab20). Для этого кликните в рабочем меню п. File | Save As ... и в открывшемся диалоговом окне введите имя рабочего файла 8888_lab20.ewb, папку и настройки оставьте по умолчанию.

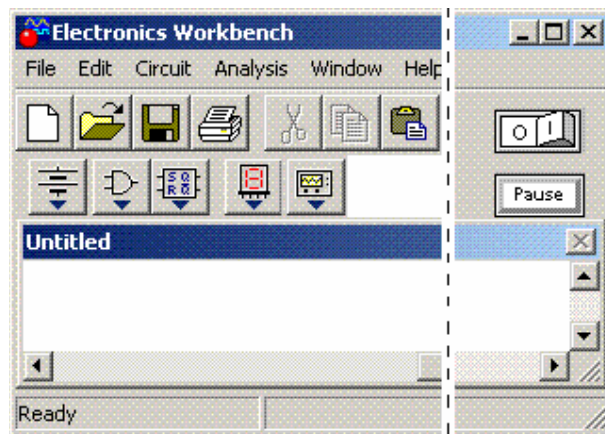


Рис. 20.2.3 Главное окно Electronic Workbench



5.1. Приступим к сборке схемы. Кликните по иконке логических элементов (ЛЭ)  на панели инструментов EW. В открывшемся "ящике" нам понадобятся следующие ЛЭ. Слева по порядку: "И &", "ИЛИ 1", "НЕ", "ИЛИ-НЕ", "И-НЕ" в зарубежном обозначении (рис. 20.2.4).



Рис. 20.2.4 Условные обозначения основных ЛЭ

Для реализации схемы (I) понадобятся ЛЭ "И", "ИЛИ" и "НЕ", а для схемы (II) - элементы "И_НЕ", "НЕ". Кроме того понадобятся некоторые инструменты, находящиеся под иконкой . Это осциллограф и генератор слов, на рисунке 20.2.5 отмечены стрелками.

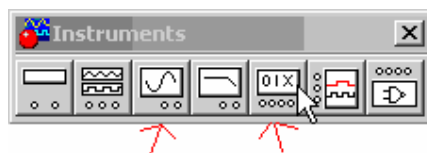


Рис. 20.2.5 Осциллограф и генератор слов

5.2. Для реализации схемы (I) понадобятся 4 логических элемента "И", один ЛЭ "ИЛИ" и 4 инвертора. Поместим их на рабочее поле нашего чертежа. Для этого подведите курсор к соответствующему ЛЭ, нажмите на левую кнопку и вытащите элемент на рабочее поле.

Из рисунка 20.2.6 видно, что все ЛЭ, кроме инверторов имеют два входа, а в схеме два двухвходовых ЛЭ "И", один трехвходовый и один четырехвходовый. Кроме того, требуется четырехвходовый элемент "ИЛИ". Изменить число входов можно сделав "правый клик" на нужном ЛЭ и далее в п. контекстного меню Component Properties... | Number of Inputs отметить нужное число. Развернуть ЛЭ на чертеже можно вызвав правым кликом контекстное меню и затем использовать п. меню Rotate или Flip...

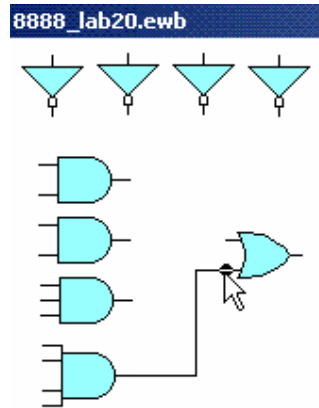


Рис. 20.2.6 Примерное расположение ЛЭ на чертеже

Необходимые соединения делаются следующим образом: подведите курсор к выводу, появится "электрический" курсор в виде жирной точки. Нажмите левую кнопку и тяните проводник к другому выводу в соответствии со схемой до появления следующей точки.

5.3. Подключите к схеме осциллограф и генератор слов. Окончательно схема (I) будет выглядеть примерно как на рис. 20.3.

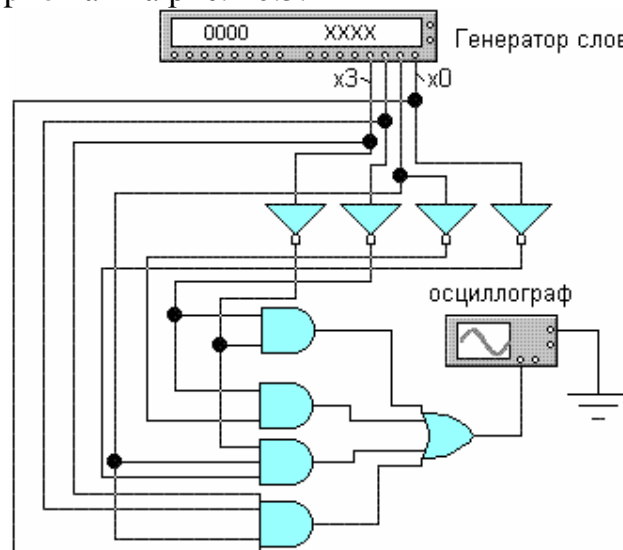


Рис. 20.3 Окончательная схема

Двойным кликом откройте рабочую панель генератора слов и измените следующие установки: конечное значение = F_{16} (15_{10}), затем кликнув по кнопке "Pattern" задайте режим работы генератора слов - "Up counter", что означает работу в режиме суммирующего счетчика (счетчики будут изучаться в курсе отдельно). Изменения отмечены на рисунке 20.3.1.

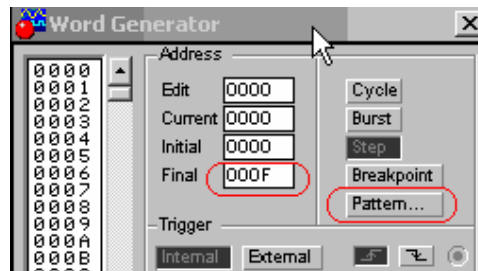



Рис. 20.3.1 Настройка генератора слов

Теперь откройте панель осциллографа, установите масштаб времени 2мсек/деление и включите установку переключателем . Зафиксировать временную диаграмму на экране осциллографа лучше всего в пошаговом режиме "Step" генератора слов. Как и должно быть экспериментально полученная временная диаграмма (рис. 20.4) совпала с заданной временной последовательностью (рис. 20.1). На рисунке выделен один период повторения.

На экране осциллографа (рис. 20.4) виден также незапланированный короткий всплеск (импульсная помеха). Появление таких злобредных "пичков" обусловлено конечным значением времени задержки распространения сигналов при прохождении через элементы схемы $t_{\text{зр}} > 0$ (или t_{pd} в зарубежном обозначении). В частности, триггеры генератора слов переключаются не одновременно, а последовательно начиная с младшего разряда. Таким образом, переход выходных значений генератора слов от комбинации $x_3x_2x_1x_0 = 1111$ к комбинации $x_3x_2x_1x_0 = 0000$ происходит не одновременно. Сначала переключается младший триггер x_0 и появляется код 1110 (14-й интервал), затем переключается триггер x_1 - появляется код 1100 (12-й интервал), далее x_2 - код равен 1000 (8-й интервал) и наконец-то схема переходит к коду 0000. Таким образом при переполнении счетчика возникают три незапланированные короткие кодовые комбинации.

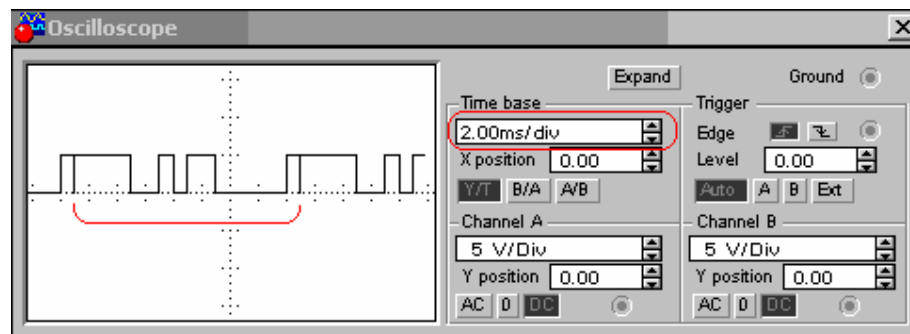


Рис. 20.4 Временная диаграмма

Проанализируем, как поведет себя разработанная нами схема при этих входных значениях. Для этого снова обратимся к рис. 20-1 с заданной импульсной последовательностью. При комбинации $x_3x_2x_1x_0=1000$ выходное значение $Y=1$, т.е. единичный уровень выходного сигнала прерываться не будет. А вот при значениях $x_3x_2x_1x_0=1100$ и 1110 выходное значение $Y=0$, т.е. единичный уровень выходного сигнала сменится в течение короткого времени равного $2 \cdot t_{\text{зр}}$, что и приводит к кратковременной помехе, которая видна на экране осциллографа. Естественно, такой же вывод можно получить подставляя значения $x_3x_2x_1x_0=1100$ и 1110 в формулу (I) или (II). Методы подавления таких помех изложены в соответствующих разделах учебного курса и далее в п.7.

5.4. Реализация схемы (II) и ее проверка проводятся аналогично.

Выполнение работы:

6. Получите задание, проведите соответствующий расчет, поочередно соберите схемы (I) и (II) и покажите временные диаграммы преподавателю.

7. Выполните модификацию схемы, устраняющую помехи на рис. 20.4 и также покажите временные диаграммы преподавателю.

ЛАБОРАТОРНАЯ РАБОТА №21

СИНТЕЗ ПРЕОБРАЗОВАТЕЛЯ ДВОИЧНОГО (ДВОИЧНО-ДЕСЯТИЧНОГО) КОДА В СЕМИСЕГМЕНТНЫЙ

Цель работы: Изучение преобразователей кодов.

На рисунке 21.1 показан фрагмент подключения одного светодиода (сегмента) к выходу преобразователя с открытыми коллекторами и приведены начертания цифр. В качестве примера рассмотрим разработку преобразователя двоично-десятичного кода в семисегментный (BCD/7seg).

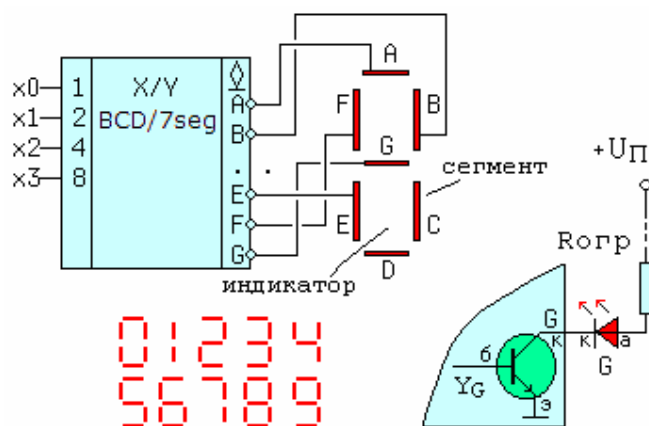


Рис.21.1 Блок схема преобразователя и фрагмент подключения отдельного светодиода (сегмента)

Этапы проектирования.

Такой преобразователь должен иметь четыре входа, т.к. для кодирования десятичных цифр от 0 до 9 достаточно $\log_2 10 = 4$ -х двоичных, и семь выходов, по одному на каждый сегмент.

Сформулируем условия свечения/гашения светодиода:

1) Светодиод включен, если напряжение на его аноде больше, чем на катоде (о конкретных значениях напряжения и тока пока речь не идет). Анод через ограничивающий ток резистор уже подключен к плюсу источника питания, поэтому на катоде должен быть потенциал близкий к нулю. Для этого p-n-p транзистор, работающий в ключевом режиме, должен быть открыт. Тогда потенциал его коллектора близок к нулю. Транзистор открыт, если потенциал на его базе больше нуля, т.е. должно быть $Y_G = 1$ (Y_G - логическая переменная, соответствующая сегменту G).

2) Светодиод выключен, если потенциалы его анода и катода равны. Это достигается, если ключевой транзистор закрыт и через него не протекает ток. Потенциал базы в этом случае должен быть равен нулю, т.е. $Y_G = 0$.

Теперь, в соответствии с полученными условиями, заполним таблицу истинности преобразователя. Например в цифре 0 должны светиться все сегменты за исключением сегмента G. В цифре 1 светятся только два сегмента B и C и т.д. Весовые коэффициенты b^i двоично-десятичных разрядов равны 2^i (8,4,2 и 1). На рис.21.2 приведена таблица истинности. В таблице заполнена только колонка для сегмента A. Нули в ней проставлены для тех цифр, в которых сегмент A не светится.

Десятичная цифра	8	4	2	1	сегменты						
	x3	x2	x1	x0	Y _A	Y _B	Y _C	Y _D	Y _E	Y _F	Y _G
0	0	0	0	0	1
1	0	0	0	1	0
2	0	0	1	0	1
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	1

Рис.21.2 Таблица истинности для сегмента А

В общем случае для синтеза этого ПК требуется составить семь уравнений. Найдем одно, для сегмента А, заполнив для него таблицу Карно. Ниже на рисунке приведена ТК прямого значения функции Y_A сегмента А. Когда "нулевых" клеток в таблице меньше, чем "единичных" и/или они компактно сгруппированы (рис.21.3), целесообразно искать алгебраическое выражение инверсной логической функции, т.е. ~Y_A. Логическая функция при этом может получиться значительно проще, т.е. содержать меньше переменных и слагаемых.

		Y _A			
		00	01	11	10
x3x2	00	1	0	1	1
	01	0	1	1	1
	11	Φ	Φ	Φ	Φ
	10	1	1	Φ	Φ

Рис. 21.3 Таблица Карно для сегмента А

Шесть значений ЛФ в таблице не определены (Φ) из-за отсутствия десятичных цифр больших девятки, поэтому для минимизации алгебраического выражения ЛФ доопределяем некоторые из них. Для нахождения инверсного значения ~Y_A необходимо нарисовать еще одну ТК с инвертированными значениями ЛФ. Можно проделать это мысленно и объединить в прямоугольники клетки с нулевыми значениями ЛФ. Из таблицы найдем: ~Y_A = x₂*~x₁*~x₀ + ~x₃*~x₂*~x₁*x₀. Проинвертировав полученное выражение получим окончательный результат:

$$Y_A = x_2 * \bar{x}_1 * \bar{x}_0 + \bar{x}_3 * \bar{x}_2 * \bar{x}_1 * x_0$$

В задании для работы будут варьироваться сегменты, количество входных двоичных наборов (10...16), а также начертания некоторых цифр.

Выполнение работы:

- получите задание,
- заполните таблицу истинности для заданного сегмента(ов) и соответствующую ему таблицу Карно,

- в) вычислите алгебраическое уравнение,
 г) запустите Electronic WorkBench (см. п.5 лабораторной работы №20) и соберите свою схему аналогично рис.20-3, только вместо осциллографа к выходам преобразователя подключите 7-ми сегментный индикатор,



- д) покажите преподавателю свечение сегмента в соответствующих кодах на входах преобразователя.

ЛАБОРАТОРНАЯ РАБОТА №22 РАЗРАБОТКА СХЕМ НА РЕГИСТРАХ СДВИГА С ОБРАТНОЙ СВЯЗЬЮ

Цель работы: Изучение работы регистра сдвига.

Такие схемы используются в технике связи, для кодирования (декодирования) данных, при нахождении циклического избыточного кода - CRC (при архивировании, передаче данных, ...) и т.д.

На рисунке 22.1 приведено схематичное обозначение трехразрядного регистра сдвига и диаграммы его состояний. С приходом очередного импульса сдвига входное значение F_{oc} записывается в младший разряд и одновременно содержимое регистра сдвигается на один разряд влево (регистр переходит в новое состояние). Составим диаграмму, начиная с нулевого состояния. Если в триггер младшего разряда Q_0 записывается 0, то на диаграмме новое состояние записывается слева, а если - 1, то - справа. Каждому переходу соответствует импульс сдвига.

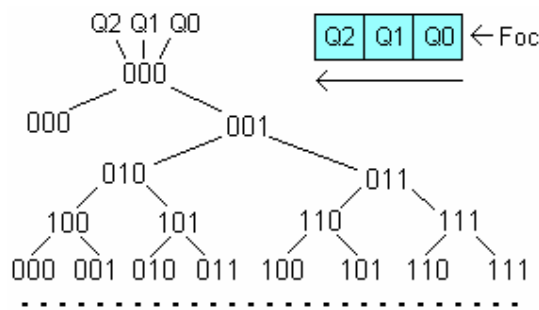


Рис. 22.1 Диаграмма состояний 3-х разрядного регистра сдвига

Объединяя одноименные узлы древовидной диаграммы, получим кольцевую диаграмму состояний трехразрядного регистра сдвига (рис. 22.2).

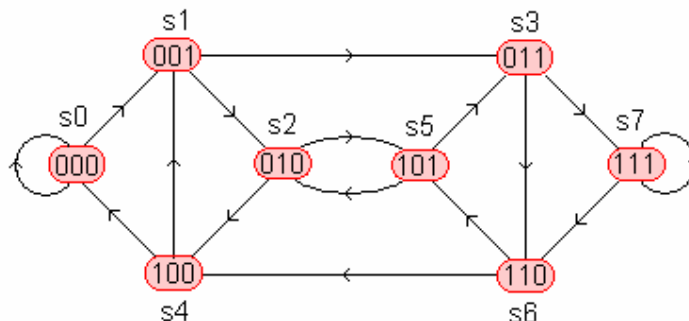


Рис. 22.2 Кольцевая диаграмма состояний 3-х разрядного регистра сдвига

Расчет схемы.

На рис. 22.3 приведен вариант типового устройства с комбинационной логической схемой (КЛС1) в цепи обратной связи. КЛС1 обеспечивает требуемые переходы. КЛС2 нужна только в том случае, если заданный алгоритм переходов не обеспечивается диаграммой состояний. Выходное значение F_{oc} КЛС1 записывается в младший триггер регистра с приходом очередного активного фронта импульса сдвига.

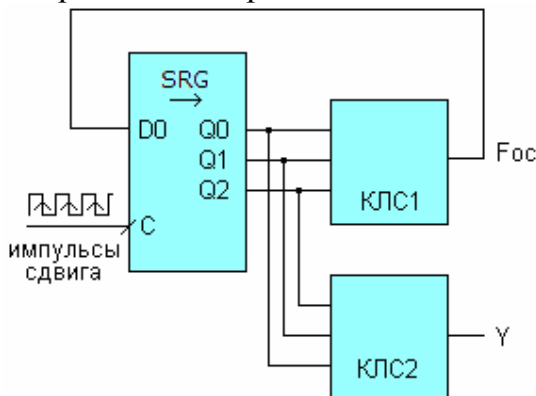
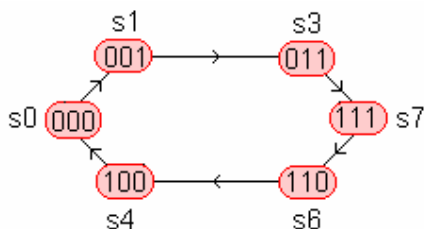


Рис. 22.3 Пример схемы

В качестве примера рассмотрим синтез схемы (техническое задание или ТЗ) имеющей 6 состояний: $s_0 \rightarrow s_1 \rightarrow s_3 \rightarrow s_7 \rightarrow s_6 \rightarrow s_4 \rightarrow s_0 \dots$



Составим таблицу состояний. КЛС1 должна генерировать при текущем состоянии s_i значение F_{oc} , которое с приходом тактового импульса записывается в младший разряд регистра Q0, как показано в таблице 22.1.

Таблица 22.1

состояние	Q2 Q1 Q0	F_{oc}
s_0	000	1
s_1	001	1
s_3	011	1
s_7	111	0
s_6	110	0
s_4	100	0
s_0	000	1
....

Переходим к расчету логической функции Fос КЛС1, для этого заполняем таблицу Карно. Символом Φ , как всегда обозначаем неопределенное на этапе технического задания значение логической функции (в ТЗ незадействованы и соответственно неопределены состояния 010 и 101).

		Q1Q0			
		00	01	11	10
Q2	0	1	1	1	Φ
	1	0	Φ	0	0

Объединяя отмеченные в ТК значения, получаем "очевидное" решение:

$$F_{ос} = \sim Q_2. \text{ (формула I)}$$

Проверим, как поведет себя логическая функция Fос при двух неиспользуемых состояниях: $Q_2Q_1Q_0 = 010$ и $Q_2Q_1Q_0 = 101$.

1). При $Q_2Q_1Q_0 = 010$ логическая функция $F_{ос}(010) = \sim Q_2 = 1$ и с приходом очередного импульса сдвига регистр перейдет в состояние $Q_2Q_1Q_0 = 101$.
 2). При $Q_2Q_1Q_0 = 101$ логическая функция $F_{ос}(101) = \sim Q_2 = 0$ и с приходом очередного импульса сдвига регистр перейдет в состояние $Q_2Q_1Q_0 = 010$ и далее процесс заикнется на этих двух состояниях.

Таким образом, проверка выявила побочный (незапланированный) бесконечный цикл, в который схема может перейти при включении питания, при воздействии помехи и т.д. Окончательный вид диаграммы состояний для нашего "очевидного" решения с учетом проведенного анализа будет выглядеть следующим образом (рис. 22.4):

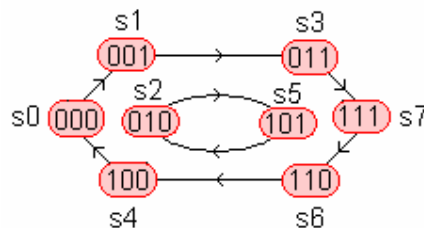


Рис. 22.4 Вариант диаграммы переходов с побочным циклом

Естественно, такое решение не может нас устроить. Выход из "порочного" цикла в нашем варианте можно осуществить тремя способами: переопределив одну или две связи в диаграмме состояний (на рис. 22.5 показаны схематично). В вариантах А и Б выход из побочного цикла гарантированно произойдет в течение двух тактовых интервалов импульсов сдвига, а в варианте В только за один, но в этом случае выражение для Fос будет сложнее (выбор "быстрого" или менее сложного решения, как всегда за разработчиком).

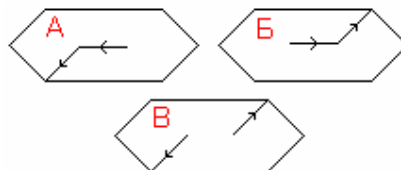


Рис. 22.5. Три варианта правильного решения задачи

Для корректировки технического решения выберем вариант Б. Как видно из рисунка, в этом случае необходимо переопределить переход из состояния 101 не к состоянию 010, а к 011. То есть, значение $F_{ос}(101)$ должно быть равно не нулю, а единице: $F_{ос}(101)=1$. Подкорректируем значение в соответствующей клетке таблицы Карно:

		Q1Q0			
		00	01	11	10
Q2	0	1	1	1	Φ
	1	0	1	0	0

Окончательно диаграмма состояний и алгебраическое решение будут выглядеть следующим образом (рис.22.6).

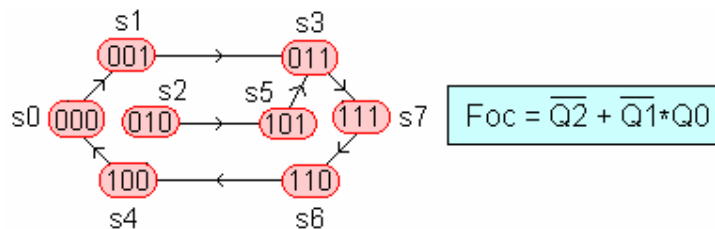


Рис. 22.6. Одно из правильных решений

Рисунок КЛС1, соответствующий формуле на рис. 22.6, не приводится.

Выполнение работы:

- получите задание,
- выявите возможные побочные циклы (если они есть) и выходы из них,
- нарисуйте окончательную диаграмму состояний аналогичную рисунку 22.6 но, естественно, для вашего варианта,
- заполните таблицу состояний и соответствующую ей таблицу Карно,
- вычислите алгебраическое уравнение,
- нарисуйте две схемы КЛС1: в базисе "И-ИЛИ-НЕ" и в базисе "И-НЕ" (см. лабораторную работу №20),
- запустите Electronic WorkBench (см. п.5 лабораторной работы №20) и соберите свою схему аналогично рис.22.3, причем схему регистра сдвига возьмите из конспекта, а D-триггер выберите следующим:



- тактовые импульсы сдвига можно получить в пошаговом режиме работы генератора слов на крайнем правом его выходе (см. п. 5.3 лаб. работы №20),

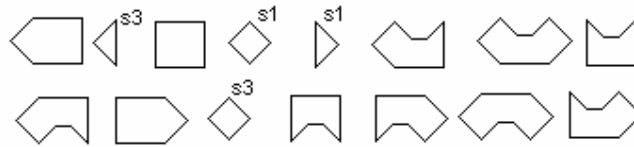


к) для контроля состояний используйте 7-ми сегментный индикатор со встроенным преобразователем BIN кода в семисегментный (см. лаб. работу №21).



л) смену состояний в соответствии с заданием предъявите преподавателю.

м) **некоторые варианты заданий:**



ЛАБОРАТОРНАЯ РАБОТА №23 СИНТЕЗ СЧЕТЧИКА С ПРОИЗВОЛЬНЫМ МОДУЛЕМ СЧЕТА $M < 2^n$

Цель работы: Изучение построения и функционирования счетчиков.

Для работы выбрана одна из множества схем (рис.23.1), например, на основе 4-х разрядного асинхронного счетчика (свойства и функционирование счетчика описаны в конспекте лекций).

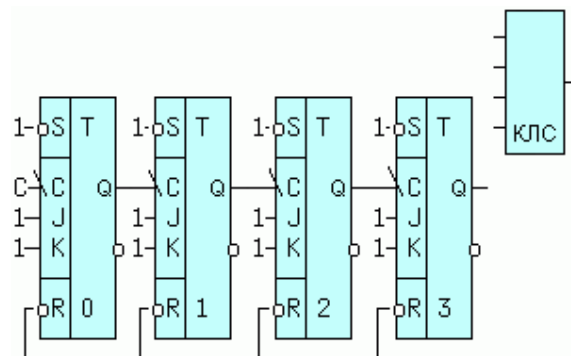


Рис.23.1 Четырехразрядный двоичный счетчик

Этапы проектирования.

В качестве примера проектирования выберем $M=10$ (двоично-десятичный счетчик, широко использующийся в схемотехнике). Такой счетчик должен иметь $M=10$ состояний от 0 до $M-1$ т.е. до 9-ти. С приходом каждого 10-го импульса счетчик должен возвращаться в исходное состояние, например в нулевое. Число триггеров находим из формулы: $n \geq \log_2 M$ (ближайшее большее целое число), т.е. $n=4$.

Перевод в нулевое состояние всех триггеров можно осуществить, сформировав с помощью комбинационной логической схемы (КЛС) сигнал $Y_{ос} = \sim R = 0$ ($\sim S = 1$), который "сбросит" все триггеры в момент поступления 10-го импульса. Во время счета от нуля до 9-ти сигнал $Y_{ос}$ должен принимать пассивное значение $Y_{ос} = 1$. В соответствии с этим описанием, заполним таблицу состояний двоично-десятичного счетчика.

Таблица 23.1

состояние	Q3Q2Q1Q0	Yoc
0	0000	1
1	0001	1
2	0010	1
3	0011	1
4	0100	1
5	0101	1
6	0110	1
7	0111	1
8	1000	1
9	1001	1
10	1010	0

Вычисление алгебраического выражения Yoc произведем с помощью таблицы Карно 23.2.

Таблица 23.2

		Q1Q0			
		00	01	11	10
Q3Q2	00	1	1	1	1
	01	1	1	1	1
	11	Φ	Φ	Φ	Φ
	10	1	1	Φ	0

Пять состояний счетчика в таблице состояний 23.2 не определены и могут быть использованы для получения более компактной формулы (см. п.4.3 лаб. работы №20). Найдем инверсное значение $\sim Y_{oc} = Q_3 * Q_1$ (см. лаб. работу №21). Окончательное выражение для Yoc будет иметь следующий вид:

$$Y_{oc} = \bar{Q}_3 * Q_1$$

Теперь схема (рис. 23.2) примет законченный вид:

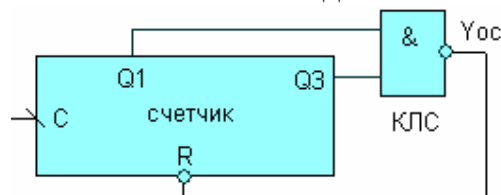


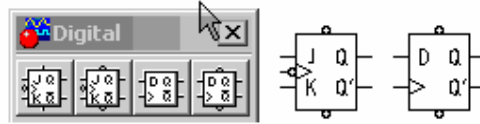
Рис.23.2 Схема счетчика по модулю 10

ВНИМАНИЕ: Варианты задания будут усложнены тем, что счетчик будет собираться с использованием двух различных типов триггеров (дополнительно к JK-триггеру с инверсным динамическим входом будет применен D-триггер с прямым динамическим входом). 1). Особенности проектирования счетчиков на триггерах с прямым и инверсным динамическим входом изложены в конспекте (см. первую лекцию

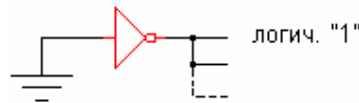
по счетчикам). 2). D-триггер необходимо преобразовать в Т-триггер, т.е. перевести его в счетный режим (также см. конспект).

Выполнение работы:

а) Получите задание: модуль счета М, а также набор триггеров (в задании последовательность триггеров в счетчике будет задана в виде, например "J-D-D-J"). Это означает, что крайние триггеры 0 и 3 - JK типа, а два средних 1 и 2 это D - триггеры,



не забудьте подключить, незадействованные входы $\sim S$ к логической "1", например таким образом:



б) заполните таблицу состояний и соответствующую ей таблицу Карно,
в) вычислите алгебраическое уравнение,

г) запустите Electronic WorkBench (см. п.5 лабораторной работы №20) и соберите свою ☐ схему аналогично рис.23-2. К счетному входу счетчика подключите младший разряд ☐ генератора слов (и переведите его в пошаговый режим суммирующего счетчика см. п. 5.3 лаб. работы №20).

д) для контроля состояний используйте 7-ми сегментный индикатор со встроенным ☐ преобразователем кода,



е) смену состояний в соответствии в заданием предьявите преподавателю.
ж) отключите обратную связь, удалив проводник Yoc из схемы, подключите освободившиеся входы $\sim R$ к логич. "1" и переделайте суммирующий счетчик в вычитающий (2 способа приведены в конспекте), результат предьявите преподавателю.

ЛАБОРАТОРНЫЕ РАБОТЫ №24-29 В ПРОЦЕССЕ ОФОРМЛЕНИЯ

НЕКОТОРЫЕ ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

Шина - группа проводников (линий связи), имеющих похожее функциональное назначение или выполняющих общую функцию.

Строб - короткий импульс, выделяющий установившееся, свободное от помех значение информационного сигнала.

Системная шина - совокупность трех шин: шины данных (ШД), шины адреса (ША) и шины управления (ШУ).

КОП - однобайтовый код операции (первый байт любой команды)

ALU Арифметико-логическое устройство (АЛУ)

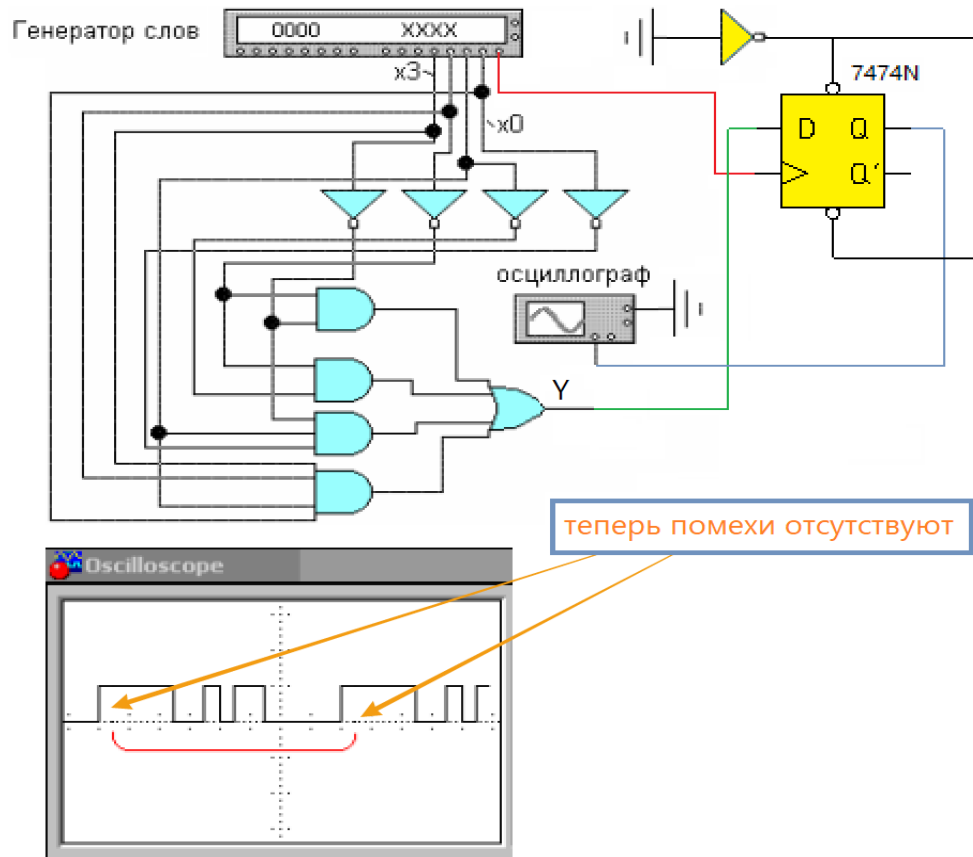
CE Chip Enable или Crystall Enable (тоже CS)
CMOS CoMpleMentary-syMMetry/Metal-oxide seMiconductor, комплементарная логика на транзисторах металл-оксид-полупроводник (КМОП)
CPU Центральный процессор
CS Chip Select или Crystall Select (тоже CE)
E2PROM Electrically Erasable PrograMMable ROM
EEPROM Electrically Erasable PrograMMable ROM
EPROM Erasable PrograMMable ROM
FIFO Структура данных, организованная по принципу "первым вошел - первым вышел" (конвейер или очередь)
FRAM Ferroelectric RAM
LED Светодиод
LIFO Структура данных, организованная по принципу "последним вошел - первым вышел" (стек)
MRAM Magnitoelectric RAM
OTP One TiMe PrograMMable (PROM)
OUM Ovonic Unified MeMory
PROM PrograMMable ROM (OTP)
RAM RandoM Access MeMory (ОЗУ, ЗУПВ)
ROM Read Only MeMory (ПЗУ)
АЛУ Арифметико-логическое устройство (ALU)
АЦП Аналого-цифровой преобразователь
БИС Большая интегральная схема
ВВ Ввод - вывод
ВУ Внешнее устройство
ЖКД Жидко-кристаллический дисплей (LCD)
ЖКИ Жидко-кристаллический индикатор
ЗУПВ Запоминающее устройство с произвольной выборкой (ОЗУ)
ИМС Интеральная микросхема (тоже ИС)
ИС Интегральная схема (тоже ИМС)
КЛС Комбинационная логическая схема (тоже КС)
КМОП К-МОП комплементарная логика на транзисторах металл-оксид-полупроводник (CMOS)
КОП Код операции
КС Это КЛС
ЛБ Логический базис
ЛФ Логическая функция
ЛЭ Логический элемент
МК Микроконтроллер
МП Микропроцессор
ОЗУ Оперативное запоминающее устройство (ЗУПВ)
ОК Открытый коллектор
ОЭВМ Однокристалльная ЭВМ (микроконтроллер)
ПДП Прямой доступ к памяти (DMA)
ПЗУ Постоянное запоминающее устройство (ROM)
ПИТ Программируемый нтервальный таймер
ППИ Программируемый периферийный интерфейс

ПС	Последовательностная схема
ПФ	Переключательная функция
РОН	Регистр общего назначения
САПР	Система автоматизированного проектирования
СБИС	Сверхбольшая интегральная схема
СДНФ	Совершенная дизъюнктивная нормальная форма
ТИ	Таблица истинности
ТК	Таблица Карно
ЦАП	Цифро-аналоговый преобразователь
ЦУ	Цифровое устройство
ШИМ	Широтно-импульсная модуляция (PWM)
ЭВМ	Ну, это и так понятно
ЭП	Элемент памяти
ЯП	Ячейка памяти

СПИСОК ЛИТЕРАТУРЫ

1. Китаев, Ю. В. Основы микропроцессорной техники : учебное пособие / Ю. В. Китаев. — Санкт-Петербург : НИУ ИТМО, [б. г.]. — Часть 1 — 2016. — 51 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/91388>
2. Китаев, Ю. В. Лабораторная установка на основе виртуальных приборов и USB интерфейса : учебное пособие / Ю. В. Китаев. — Санкт-Петербург : НИУ ИТМО, 2012. — 75 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/40870>
3. Китаев, Ю. В. Программирование МК на ассемблере ASM-51 и AVR Pascal : учебное пособие / Ю. В. Китаев. — Санкт-Петербург : НИУ ИТМО, 2011. — 90 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/71012>
4. Китаев, Ю. В. Программирование мк на ассемблере ASM-51 и AVR Pascal : учебное пособие / Ю. В. Китаев. — Санкт-Петербург : НИУ ИТМО, 2010. — 90 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/40773>
5. Китаев, Ю. В. Лабораторные и практические работы: Электроника и МП техника : учебное пособие / Ю. В. Китаев. — Санкт-Петербург : НИУ ИТМО, [б. г.]. — Часть 1 — 2008. — 90 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/43794>
6. Китаев, Ю. В. Основы цифровой техники : учебное пособие / Ю. В. Китаев. — Санкт-Петербург : НИУ ИТМО, 2007. — 87 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/43631>
7. Китаев, Ю. В. Основы программирования микроконтроллеров ATmega128 и 68hc908 / Ю. В. Китаев. — Санкт-Петербург : НИУ ИТМО, 2007. — 107 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/43634>

1. Перенесите соединительные проводники x3..x0 вашей схемы, подключённые к генератору слов на один разряд влево. К освободившейся правой клемме генератора подключите синхровход '>' динамического D тиггера и произведите остальную перекомпоновку в правой части схемы.
2. Установите в настройках генератора слов финальное 16-ричное значение 'Final' = 20 (hex) и запустите симуляцию кнопкой 'Burst' генератора.
3. Убедитесь, в отсутствии помех на экране осциллографа.



Объяснение очень простое - в начало цепочки триггеров генератора слов вклинивается дополнительный D-триггер, который срабатывает первым и запоминает предыдущее значение логической функции Y, которое уже естественно свободно от переходных процессов (помех). Вслед за этим начинают по цепочке переключаться триггеры x0-x1-x2-x3. Снова на короткое время появляются ложные состояния (помехи) и наконец на выходе Y появляется новое истинное значение, которое опять фиксируется D-триггером и отправляется на осциллограф. Таким образом происходит непрерывный процесс 'очищения'.