

Практическое занятие
Моделирование работы цифрового функционального узла
«Триггер»

Цель работы: изучить принцип работы одного из основных логических функциональных узлов цифровых систем автоматизированного управления (ЦСАУ) и цифровых вычислительных систем – «Триггер».

Учебное оборудование: Моделирование работы функционального узла проводится в программной среде электронной лаборатории Electronics Workbench/Multisim.

Используются следующие виртуальные элементы и тестовые измерительные инструменты Electronics Workbench/Multisim:

	Название элемента	Элемент EWB	Библиотека EWB
	Источник постоянного напряжения	Battery	Sources
	Источник питания цифровых схем +5В	Vcc Voltage Source	Sources
	Заземление схемы	GND	Sources
	Переключатель	Switch	Basic
	Переменный резистор	Potentiometer	Basic
	Цифровой семисегментный индикатор со встроенным дешифратором	Decoded Seven-Segment Display	Indicators
	Генератор слова	Word Generator	Instruments
	Микросхема 74173 – 4- разрядный регистр с 3-мя состояниями выхода	4-bit D-type Reg w/3-state OUT	Digital ICs
	Вольтметр	Voltmeter	Indicators
	Осциллограф	Oscilloscope	Instruments

Краткие теоретические сведения

Триггер – это цифровое электронное логическое устройство, обладающее свойством неограниченно долго находиться в одном из двух устойчивых состояний («1» или «0») и скачкообразно переключаться из одного состояния в другое под воздействием внешнего сигнала.

Триггеры являются основным базовым элементом для электронного хранения информации в компьютерной вычислительной технике. Триггеры также широко используются во многих узлах электронной аппаратуры, в том числе цифровых системах автоматизированного управления и устройствах автоматики.

Свое название триггеры получили в соответствие со своим свойством фиксировать состояние сигнала на входе (англ. *trigger* – «защелка»). Кроме того, благодаря своему управляющему свойству – реакции на воздействие сигнала на входе – триггеры получили также свое второе название “flip-flop” («щелчок-хлопок»).

Триггеры бывают различных типов (см. классификацию триггеров, приведенную ниже). Все триггеры имеет два выхода – один прямой Q , другой – инверсный \bar{Q} . Количество входов при этом зависит от типа триггера.

Классификация триггеров

Триггеры делятся на различные типы в соответствие с различными критериями: по способу управления (функционального назначения), способу записи информации и по некоторым другим критериям. Рассмотрим вкратце различные типы триггеров.

Типы триггеров в зависимости от *способа управления* (функционального назначения):

- *RS-триггер*. Это базовый тип триггера. Имеет два управляющих входа (в варианте синхронного типа исполнения *RS-триггер* имеет соответственно еще один вход – для тактирующих импульсов).
- *JK-триггер*. Универсальный тип триггера. Строится на базе *RS-триггера*. Соответственно выполняет функции *RS-триггера*, но в отличие от него не имеет запрещенных состояний.
- *D-триггер*. Триггер задержки на один такт (от англ. *delay*). Имеет один управляющий вход и один тактирующий вход.
- *T-триггер*. Счетный триггер. Данный триггер имеет только один тактирующий вход, который выполняет функцию счетного входа. Это единственный вид триггера, текущее состояние которого определяется не информацией на входах, а состоянием его в предыдущем такте.

По *способу записи информации* триггеры подразделяются на асинхронные и синхронные, с прямыми и инверсными входами, а также с

дополнительными входами для внешней предварительной или принудительной установки/сброса триггера.

Асинхронный (не тактируемый) триггер имеет только управляющие входы. Триггер переключается сразу после поступления сигнала на один из входов.

Синхронный (тактируемый) триггер имеет управляющие (информационные) и синхронизирующие (тактирующие входы). Триггер переключается (при предварительном поступлении сигналов на информационные входы) после поступления синхронизирующего сигнала (синхросигнала). При отсутствии синхросигнала (синхроимпульса) триггер не переключается (остается в прежнем состоянии).

Также различают триггеры: 1) со *статическим* управлением – управление в этом случае производится по уровню (потенциалу) входного сигнала и 2) с *динамическим* управлением – управление производится по перепаду уровней (потенциалов) входного сигнала (в этом случае дополнительно различают триггеры, управляемые по переднему фронту или по срезу импульсов).

Кроме того, выделяют класс *двухступенчатых* триггеров, которые по сравнению с обычными одноступенчатыми триггерами благодаря второй ступени обладают дополнительными функциональными возможностями управления.

Базовый тип триггеров – RS-триггер

Рассмотрим подробнее базовый тип триггеров – RS-триггер.

Триггер имеет два отдельных информационных входа и два выхода:

S – от англ. *set* – устанавливать;

R – *reset* – возвращать в исходное состояние («сброс»);

Q – прямой выход (от англ. *quit* – выйти, покидать, выход);

\bar{Q} – инверсный выход (иногда также используется обозначение Q').

Таким образом, в триггере два выхода всегда находятся в противоположных состояниях (более точно, в комплементарных состояниях – см. Примечание ниже):

Примечание. «Комплементарные» (англ. *complement, complete*) – значит взаимно дополняющие друг друга (или вместе дополняющие до целого, до завершённого). Этот научный термин не следует путать с разговорным словом «комплимент» (англ. *compliment*), т.е. похвала, лесть.

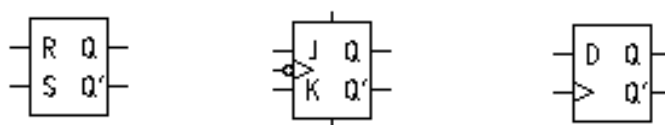


Рис. 1. Схематическое обозначение триггеров:
RS-триггер, JK-триггер и D-триггер

Как известно, все цифровые логические узлы и блоки (и, в конечном счете, вся компьютерная и информационно-вычислительная техника) строятся на базе всего трех основных логических элементов – И, ИЛИ, НЕ (AND, OR, NOT). Соответственно, триггеры также представляют собой комбинацию данных базовых элементов (см. рис. 2).

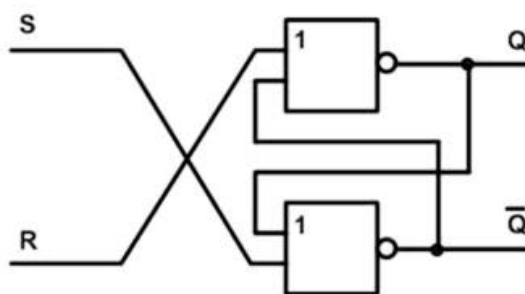


Рис. 2. Реализация RS-триггера на элементах ИЛИ-НЕ

В соответствие со свойством взаимосвязанности базовых логических функций (теорема Де Моргана, см. Примечание ниже) триггер можно также реализовать на базе элементов И-НЕ (в этом случае мы получим триггер с инверсными входами; для получения триггера с прямыми входами необходимо на входах данной схемы добавить соответственно два элемента НЕ).

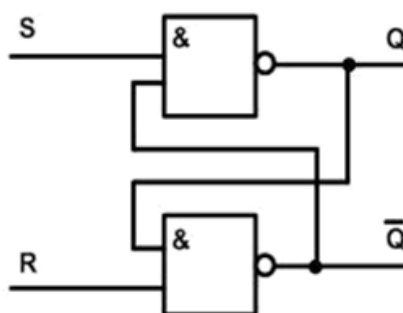


Рис. 3. Реализация RS-триггера на элементах И-НЕ

Примечание. Теорема Де Моргана (или законы Де Моргана) формулируется следующим образом:

«Отрицание конъюнкции (AND) есть дизъюнкция (OR) отрицаний».

«Отрицание дизъюнкции (OR) есть конъюнкция (AND) отрицаний».

В формульном выражении эти законы записываются следующим образом:

$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

Работа триггеров, как и любых других цифровых блоков, описывается с помощью *таблицы истинности*, а также *временной диаграммы*.

Примечание. Временная диаграмма – в дополнение к таблице истинности – показывает работу цифрового блока в динамике, т.е. как изменяются уровни напряжения для входных и выходных сигналов во времени и какие при этом требуются обеспечивать временные интервалы. Временная диаграмма соответствует той картине, которую бы мы наблюдали на экране осциллографа.

Таблица 1. Таблица истинности *RS*-триггера с прямыми входами

<i>S</i>	<i>R</i>	<i>Q</i> (<i>t</i>)	\bar{Q} (<i>t</i>)	Функция
Вход <i>Set</i>	Вход <i>Reset</i>	Выход прямой	Выход инверсный	
1	0	1	0	Установка «1»
0	0	$Q(t-1)$	$\bar{Q}(t-1)$	Режим хранения предыдущего состояния
0	1	0	1	Сброс
0	0	$Q(t-1)$	$\bar{Q}(t-1)$	Режим хранения предыдущего состояния
1	1			Запрещенное состояние

В соответствие с этой таблицей истинности, работа триггера происходит следующим образом.

При подаче активирующего сигнала (в случае прямых входов это логическая 1) на вход *S* () происходит установка триггера (*Set*), т.е. на выходе *Q* устанавливается логическая 1 (на инверсном выходе \bar{Q} , соответственно, логический 0).

После прекращения действия активирующего сигнала на входе *S*, (т.е. когда на вход *S* снова поступает логический 0) значение сигнала на выходе триггера не меняется, т.е. осуществляется режим хранения предыдущего состояния триггера. Можно сказать, что триггер фиксирует

поступивший ранее активирующий сигнал. Образно говоря, триггер как бы «защелкивает» этот сигнал (отсюда происходит его название «*trigger*», т.е. «защелка»).

То же самое происходит при подаче активирующего сигнала на вход R : триггер переходит в состояние «сброс» (*Reset*), т.е. на выходе Q устанавливается логический 0 (на инверсном выходе \bar{Q} , соответственно, логическая 1). При последующем прекращении действия этого сигнала значение сигнала на выходе триггера также не меняется.

Подача активного сигнала на оба входа R и S приводит триггер в неопределенное состояние. Поэтому эта комбинация входных сигналов считается запрещенной.

Подытоживая сказанное подчеркнем, что в результате своей работы триггер выполняет важную функцию **хранения данных**. Используя это свойство хранить данные, на базе триггеров строятся запоминающие устройства типа *SRAM* (см. Примечание ниже), которые используются во всех цифровых вычислительных устройствах (это статическая оперативная память цифровых устройств, регистры состояний, буферные регистры и т.п.).

Примечание. Как известно, память всех цифровых систем и электронных устройств делится на энергонезависимую и энергозависимую память.

Энергонезависимая память делится на:

1) *долговременную* память (так называемая память *RAM* – *Random Access Memory*, дословно с англ. – память с произвольным доступом) с возможностью перезаписи данных (например, жесткие диски, твердотельные накопители)

2) *постоянную* память (так называемая память *ROM* – *Read Only Memory*, дословно с англ. – память только для чтения), содержимое которой жестко и навсегда «прошивается» на этапе производства микросхемы.

Энергозависимая, или *оперативная*, память по принципу своей физической работы делится на *статическую* (*SRAM* – *Static RAM*) и *динамическую* (*DRAM* – *Dynamic RAM*) память. Основной тип оперативной памяти – это динамическая память (*DRAM*) ввиду его большой емкости (модификации *DRAM* – это память *DDR2*, *DDR3*, *DDR4* и новейшая память *DDR5*). Однако память типа *DRAM* требует дополнительных обслуживающих электронных схмотехнических узлов (прежде всего специальных контроллеров), что в итоге несколько усложняет аппаратуру. В случаях, когда к цифровым системам нет требований использования памяти с повышенной емкостью, обычно используется более простой тип *SRAM*. Именно на базе триггеров и строится тип памяти *SRAM*.

Задания

1. Триггеры, как устройства хранения состояний (потенциалов)

Как было сказано выше, одно из назначений триггеров – это устройство хранения одного из двух состояний (потенциалов напряжения), которые подаются на вход триггера.

Соберите схему, которая приведена на рис. 4. С помощью двух кнопочных переключателей проверьте работу триггера в соответствии с табл. 1.

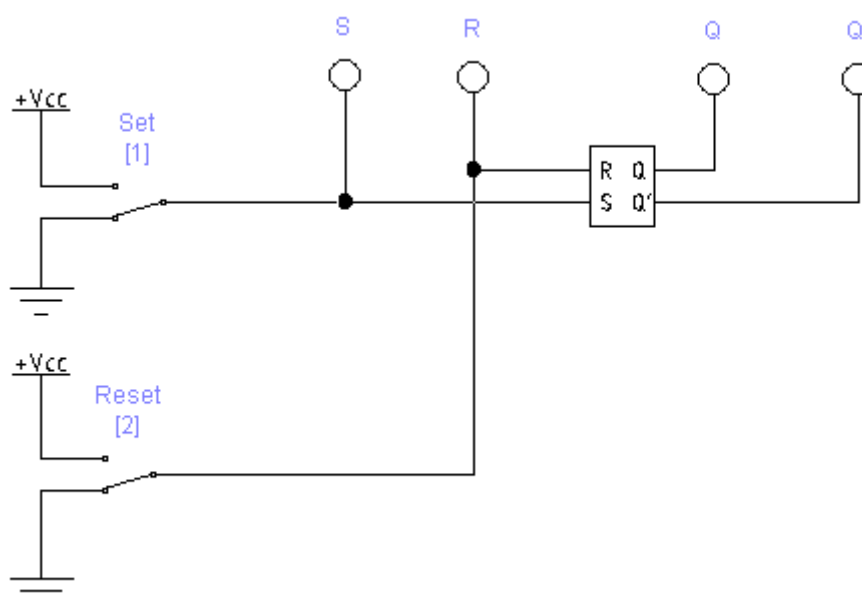


Рис. 4. Схема для изучения свойств триггера, как устройства хранения состояний (потенциалов)

2. Триггеры, как основа построения ячеек памяти

Рассмотренное выше свойство триггеров хранить состояния (потенциалы), которые подаются на вход, используется для построения ячеек памяти (см. краткие теоретические сведения, приведенные в начале этого методического пособия).

Соберите схему, которая приведена на рис. 5. Здесь в качестве триггера применен универсальный синхронный (тактируемый) *JK*-триггер. Назначение тактирующего входа – производить фиксацию состояния на входе триггера не произвольно, а только по определенной команде (сигналу). Обычно эту команду (сигнал) называют «Запись» (англ. “write”).

С помощью двух кнопочных переключателей проверьте работу триггера. Убедитесь, что фиксация состояния на входе триггера (считывание информации) производится не произвольно, а только по управляющему сигналу “write” («Запись»).

Приведенная на рис. 5 схема моделирует работу элементарной (однобитовой) ячейки памяти. Для формирования многобитовой ячейки памяти используются соответственно нужное количество параллельно подключенных триггеров. Так, на рис. 6 приведена схема моделирования работы 3-битовой ячейки памяти с помощью 3-х триггеров.

Соберите схему, которая приведена на рис. 6. Убедитесь, что данная схема выполняет функцию 3-битовой ячейки памяти.

Примечание. Обратите внимание, что управляющий сигнал подается по одной линии на все триггеры одновременно. Эта единая линия “write” в общем случае является частью общей шины управления (control bus).

Помимо сигнала “write” (запись) в цифровых устройствах обычно используется также сигнал “read” (чтение). В этом случае по команде “write” производится запись информации в ячейку памяти, а по команде “read” – чтение.

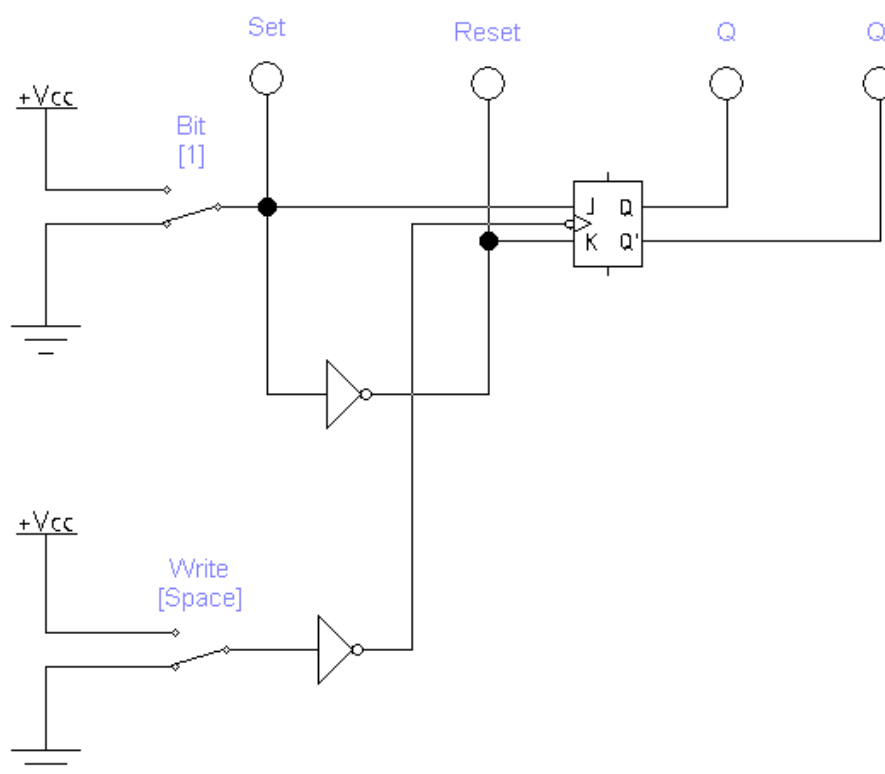


Рис. 5. Схема моделирования работы элементарной ячейки памяти

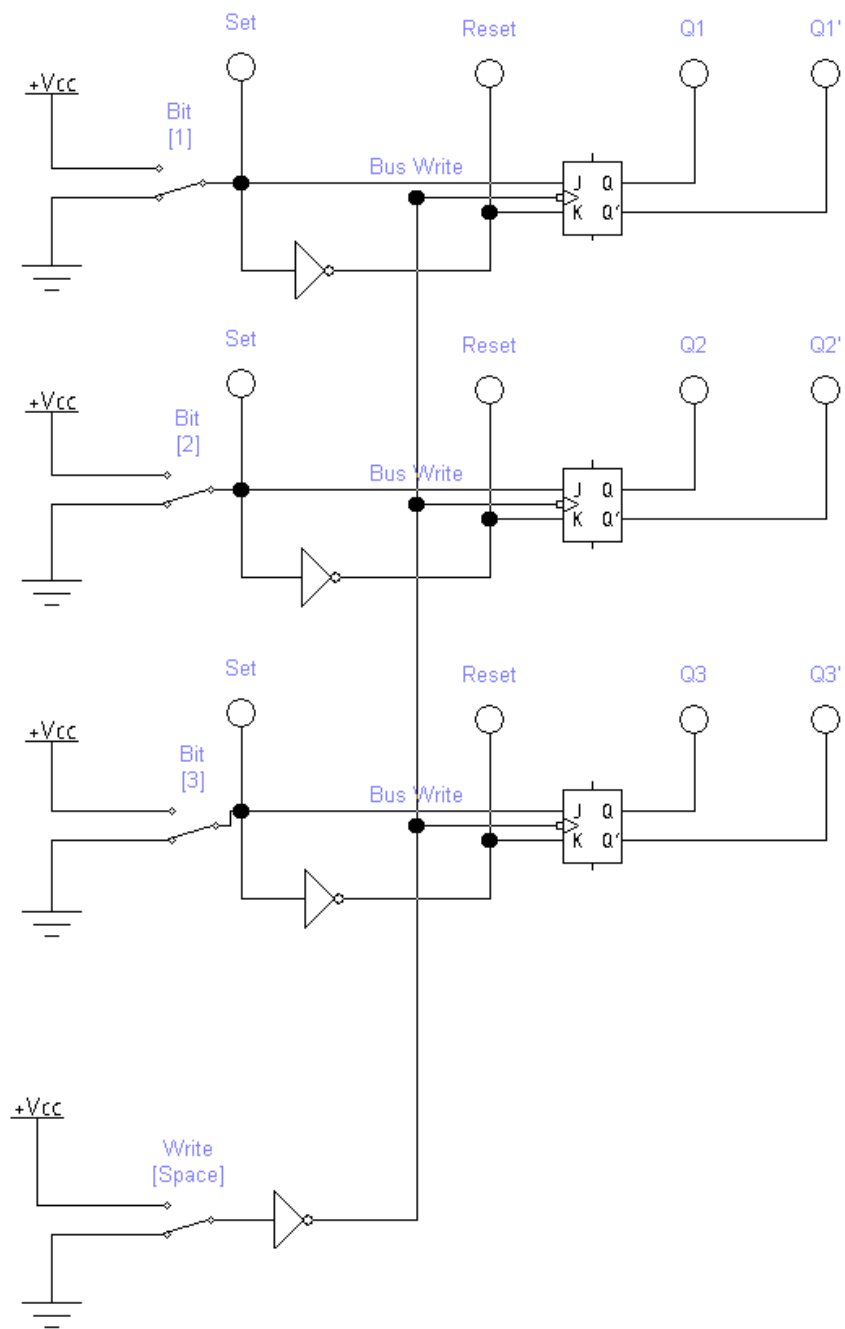


Рис. 6. Схема моделирования работы 3-битовой ячейки памяти

3. Триггеры как основа построения регистров

Краткое пояснение

С помощью триггеров формируются не только ячейки памяти общего назначения, а отдельные выделенные *регистры* (буферные регистры, регистры состояния внешних устройств и др.).

Регистры – это те же ячейки памяти на триггерах, но предназначенные для выполнения определенных целей. Назначение этих регистров можно определить по их названию. Так, *буферный регистр* служит для промежуточного хранения информации. Например, такая необходимость возникает, когда вычислительное устройство еще занято обработкой предыдущей информации, а новая информация уже поступила на вход.

Кроме того, буферные регистры выполняют роль *электрической развязки* (электрического согласования токов и напряжений) электронных цифровых узлов.

Второй тип регистров – *регистры состояния* внешних устройств. Эти регистры (обычно 8, 16 или 32-битовые) служат для хранения информации о состоянии какого-либо внешнего устройства (например, принтера, порта ввода/вывода и т.п.). Каждый отдельный бит этого регистра содержит некоторую информацию о состоянии устройства и режиме его работы (например, «устройство занято/свободно»), о режиме передачи данных («синхронная/асинхронная») и т.д. С помощью записи бита в соответствующий разряд регистра производится задание режима работы устройства, а посредством чтения бита из соответствующего разряда можно получить информацию о его состоянии.

На рис. 7 приведена схема моделирования работы 4-разрядного буферного регистра с помощью специализированной интегральной микросхемы 74173, которая входит в набор готовых цифровых микросхем библиотеки EWB Digital ICs.

Описание микросхемы

Микросхема 74173 (см. рис. 7) имеет два вывода для подключения питания, вывод сброса CLR (Clear), четыре информационных входа 1D – 4D и четыре информационных выхода 1Q – 4Q. Кроме этого имеются несколько выводов для подачи сигналов управления. Перечислим их.

Во-первых, это два вывода M и N (номера этих выводов 1 и 2), которые предназначены для задания режима разрешения записи. При подаче логического 0 (нулевого потенциала) устанавливается режим разрешения

записи. Сама запись производится подачей управляющего импульса (т.е. логической 1) на вход CLK (см. Задание 2 – работа JK-триггера). Соответственно, при подаче логического 1 (положительного потенциала) режим разрешения записи отменяется.

Во-вторых, в микросхеме имеются два дополнительных вывода G1 и G2 (номера выводов 9 и 10). Посредством этих выводов производится управление состоянием выходных линий 1Q – 4Q для исключения конфликтных ситуаций при подключении множества регистров и других электронных узлов к одной общей для всех шине данных.

Проверка работы регистра на рис. 7 производится с помощью Генератора слова (Word Generator) следующим образом. Вначале Генератор слова настраивается на задание в своей внутренней памяти тестового массива данных, например, от 0 до 7 (см. окно “Word Generator”, которое приведено в левом нижнем углу на рис. 7). Эти данные затем будут передаваться на входы регистра (точнее, на 4-разрядную шину данных – bit_0, bit_1, bit_2 и bit_3, т.е. набор из 4-х проводников, к которым подключены входы регистра). Буферный регистр при этом устанавливается в режим разрешения записи (на два вывода M и N подается логический 0, т.е. потенциал «Земля»).

При этом сама запись данных, подаваемых от Генератора слова на входы регистра, не производится до тех пор, пока на вход регистра CLK не будет дана команда “Write” (подачей положительного потенциала, т.е. логического 1). После подачи такой команды текущие заранее подготовленные данные на входах регистра будут зафиксированы. Эти данные будут храниться регистром, пока не будет подана следующая очередная команда на запись. На схеме на рис. 7 команда на запись производится с помощью переключателя (Switch) именем “Write”. При этом «переключение» Switch производится клавишей на клавиатуре <Пробел> (клавишу, на которую будет реагировать переключатель, можно переопределить в диалоговом меню EWB для элемента Switch, которое вызывается по двойному щелчку мышью).

Здесь следует обратить внимание вот на что. Как было сказано выше, команда на запись подается подачей не просто положительного потенциала (+5 В), а именно *импульса*, точнее, перепада потенциала от нулевого к положительному. Такой импульс (смена потенциалов) формируется при смене логического 0 к 1. Однако если на линии остается положительный потенциал, то для подачи импульса необходимо сначала сбросить потенциал и только потом опять подать логическую 1. На схеме на рис. 7 это означает, что после подачи с помощью переключателя “Write” команды на запись

(после установки потенциала в логическую 1), необходимо еще раз нажать на переключатель (т.е. на клавишу <Пробел>), чтобы вернуть потенциал в исходное состояние. Это действие позволяет последующим новым нажатием на переключатель сформировать новый перепад потенциалов (и соответственно, подать новую команду на запись).

Примечание. Для упрощения выполнения данного действия можно было бы использовать ключ другого типа – без фиксации. Тот ключ, который применен на схеме на рис. 7 (Switch), это по сути ключ типа «тумблер», т.е. при каждом изменении рычажка переключателя его состояние сохраняется. При использовании ключа без фиксации (типа «кнопка») состояние ключа возвращается в исходное состояние, т.е. в состояние до нажатия на кнопку. К сожалению, такой элемент в EWB не представлен.

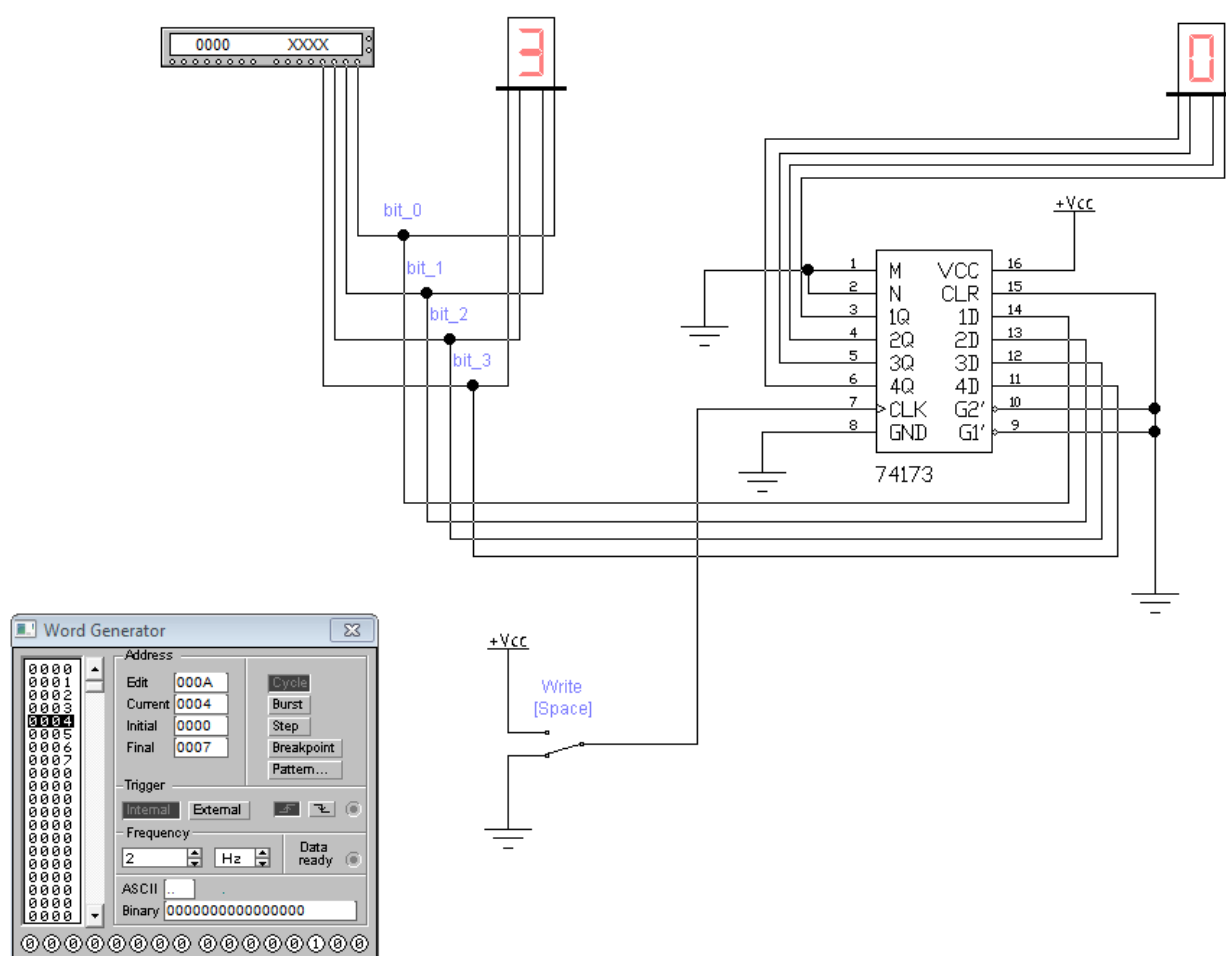


Рис. 7. Схема моделирования работы 4-разрядного буферного регистра с помощью специализированной интегральной микросхемы 74173.

На рис.7 прибор Word Generator перебирает данные (текущее значение последовательно перебираемых данных отображается на первом семисегментном индикаторе, в данном случае это значение равно 3).

Одновременно на выходе регистра (второй семисегментный индикатор) высвечивается значение 0. Это означает, что переключатель Switch еще не переводился в активное состояние, т.е. еще не была подана команда на запись.

Задания по пункту 3

3.1. Загрузите или соберите заново схему, которая приведена на рис. 7. Убедитесь, что микросхема 74173 выполняет на схеме функцию буферного регистра.

Скорость перебора прибором Word Generator своих заранее определенных данных, можно изменить (уменьшить или увеличить) заданием числового значения параметра “Frequency” (частота в Герцах для перебора данных).

Расширьте массив генерируемых прибором Word Generator данных до 16-ти значений (т.е. от 0 до F). С помощью первого семисегментного индикатора убедитесь, что Word Generator перебирает значения от 0 до F.

3.2. Построение 8-разрядного буферного регистра.

Добавьте к схеме рис. 7 еще одну микросхему 74173 для получения 8-разрядного буферного регистра.

При этом добавьте к имеющимся двум еще дополнительно два семисегментных индикатора для отображения старшего разряда выходных данных в шестнадцатеричном виде – от 00 до 1F.

Одновременно расширьте массив генерируемых прибором Word Generator данных до 32-х значений (т.е. от 00 до 1F).

|| **Индивидуальное задание.** Зафиксируйте в буферном регистре входное число, которое соответствует **номеру студента по списку в журнале преподавателя** (от 0 до 31, т.е. до 1F).
|| Сохраните файл EWB со схемой на диск, а результаты моделирования в виде скриншота включите в отчет по выполненной работе.

3.3. Построение двух отдельных 4-разрядных буферных регистров.

Измените предыдущую схему так, чтобы добавленная вторая микросхема 74173 выполняла функцию второго отдельного 4-разрядного буферного регистра.

При этом добавьте еще один переключатель (Switch) для управления записью данных во второй регистр.

Индивидуальное задание. Зафиксируйте в первом буферном регистре входное число, которое соответствует *номеру студента по списку в журнале преподавателя*. В случае если номер по списку превышает число 15 (т.е. число F) следует вычесть число 15. Например, номер 16 заменяете на 1, 17 на 2, 18 на 3 и т.д.

Во втором регистре сохраните максимальное значение 4-битового числа, т.е. число F. Сохраните файл со схемой на диск, а результаты моделирования в виде скриншота включите в отчет по выполненной работе.

4. Триггеры, как устройства запуска электронных узлов

Краткое пояснение

Как было сказано выше, триггеры являются базовым элементом построения ячеек оперативной памяти информационно-вычислительной техники. На базе триггеров строятся также двоичные счетчики (см. Практическое занятие №3), сдвиговые регистры, делители частоты, формирователи уровней сигналов и т.д.

Кроме этого, свойство триггера *скачкообразно менять свое состояние* от нулевого уровня к положительному потенциалу при достижении (превышении) входного сигнала определенного порогового значения позволяет применять их в устройства автоматики для запуска (включения) электронных узлов.

На рис. 8 приведена схема использования триггера для формирования порогового сигнала включения.

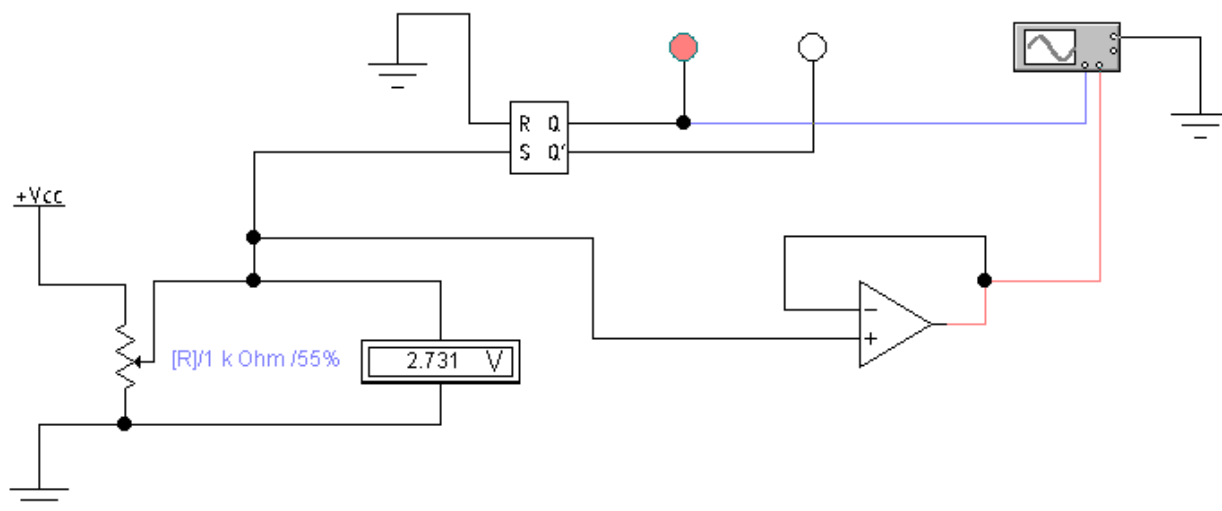


Рис. 8. Использование триггера для формирования порогового сигнала включения

Здесь на вход RS -триггера подается некоторое напряжение. В исходном состоянии триггер не активен (на выходе Q нулевой потенциал). Если напряжение на входе триггера не превышает определенного порогового уровня (а именно значения 2,5 В – т.е. половины от стандартного логического уровня +5 В), триггер остается в исходном (не активном) состоянии. По достижению (превышению) напряжения порогового уровня триггер переходит в другое состояние (происходит установка триггера). В результате на выходе Q появляется положительный потенциал, который можно использовать как сигнал включения различных электронных узлов устройств автоматики.

На схеме на рис. 8 для формирования входного напряжения используется переменный резистор – Potentiometer. Диалоговое окно “Properties” для просмотра и установки параметров переменного резистора Potentiometer приведено на рис. 9.

По умолчанию регулятор сопротивления (параметр Setting) установлен в среднее положение (50%). Это означает что в исходном состоянии переменный резистор имеет сопротивление, равное половине от его максимального значения, которое задается параметром Resistance (R). По умолчанию он равен 1кОм.

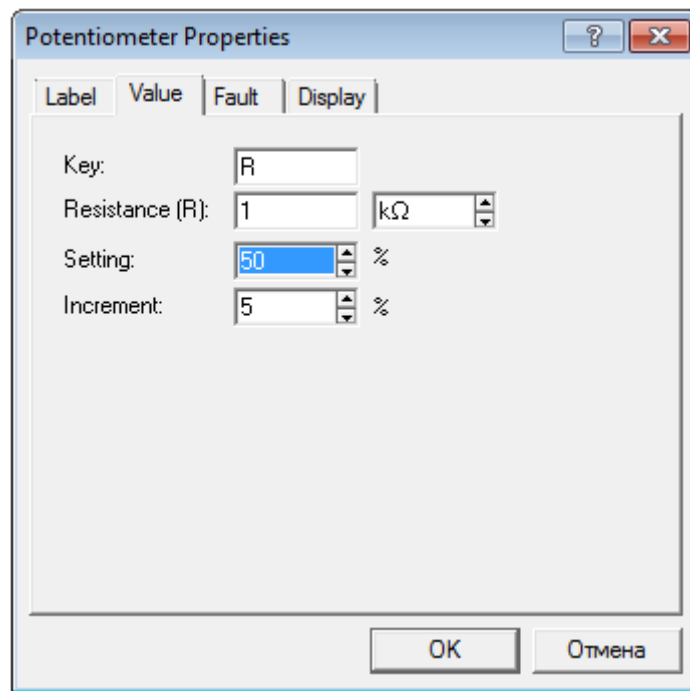


Рис. 9. Диалоговое окно для просмотра и установки параметров переменного резистора Potentiometer

Шаг регулировки сопротивления (параметр Increment) по умолчанию равен 5%. Параметр Key, который по умолчанию равен символу "R", определяет клавишу на клавиатуре, с помощью которой можно регулировать сопротивление переменного резистора в процессе моделирования работы схемы. Так, по нажатию клавиши <R> будет происходить уменьшение сопротивления (т.е. уменьшение параметра Setting), а по нажатию комбинации клавиш <Shift> + <R> будет происходить, наоборот, увеличение.

Входное напряжение формируется как падение напряжения на переменном резисторе. В случае, когда параметр Setting равен 50%, это напряжение будет равно +2,5 В (половина от максимального напряжения источника постоянного напряжения Vcc). Измерение уровня входного напряжения производится вольтметром (Voltmeter).

Для регистрации выходного сигнала используем два способа. Первый способ – обычный светодиод, или пробник (Red Probe). Он будет сигнализировать о факте перехода триггера в состояние «установка». Более детальное отслеживание процесса производится с помощью 2-канального осциллографа (Oscilloscope). На первый канал осциллографа подается входной сигнал, на второй канал – выходной.

Задания по пункту 4

4.1. Соберите схему, которая приведена на рис. 8. Задайте параметр Setting равным 0.

Запустите схему на моделирование. Двойным щелчком по осциллографу откройте переднюю панель этого прибора. Разверните переднюю панель на полное окно (кнопка Expand).

Не останавливая процесса моделирования, начинайте с помощью клавиатуры изменять входное напряжение (клавиатурная комбинация <Shift+R>). При этом следите за числовым отображением на схеме параметра регулировки переменного резистора и, одновременно, за показаниями вольтметра. По превышению параметра регулировки значения 50% должна произойти установка триггера – загореться светодиод на выходе Q. На осциллографе также будет видно, что произойдет смена потенциала выходного сигнала триггера (рис. 10).

4.2. Теперь начинайте уменьшать входное напряжение (клавиша <R>). Убедитесь, что уменьшение напряжения даже ниже порогового уровня не изменяет состояния триггера (рис. 11), поскольку в соответствии со своим свойством триггер сохраняет свое состояние.

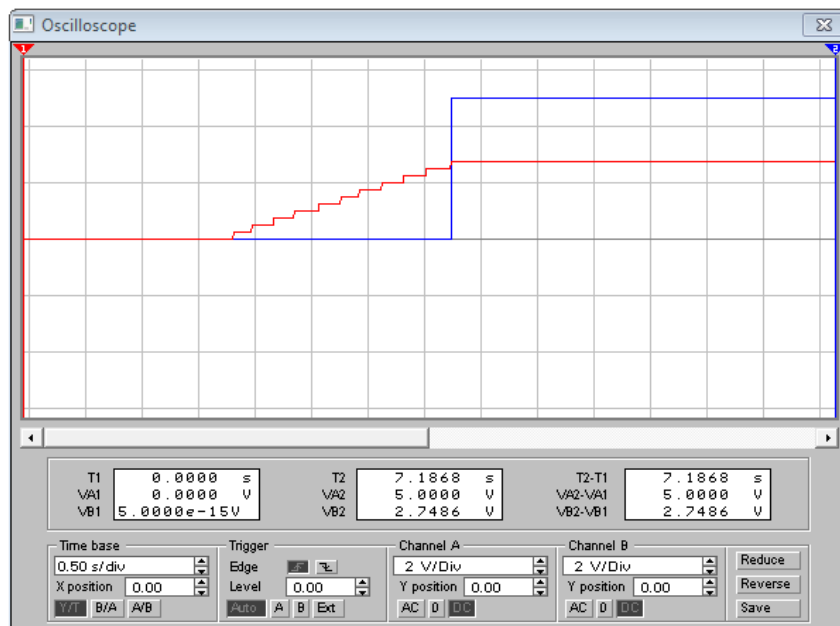


Рис. 10. Изменение входного и выходного сигнала в процессе увеличения входного напряжения (красная осциллограмма – входной сигнал, синяя осциллограмма – выходной)

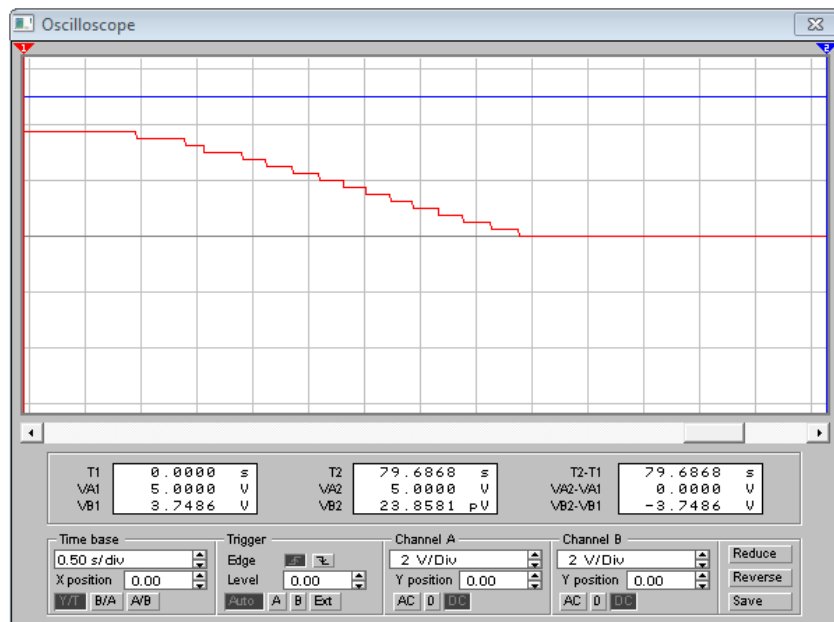


Рис. 11. Изменение входного и выходного сигнала в процессе уменьшения входного напряжения (красная осциллограмма – входной сигнал, синяя осциллограмма – выходной)

Для того чтобы сбросить триггер, необходимо подать сигнал на вход триггера R . Для этого немного изменим схему: добавим переключатель Switch для управления подачей сигнала для сброса на вход R (см. рис. 12).

4.3. Запустите измененную схему на моделирование. Повторите все действия, проведенные в п. 4.2. Затем подайте сигнал на вход R (нажатием на клавишу <Пробел>). Убедитесь, что в результате триггер изменит свое состояние и произойдет скачкообразное выключение активного выходного сигнала (рис. 13).

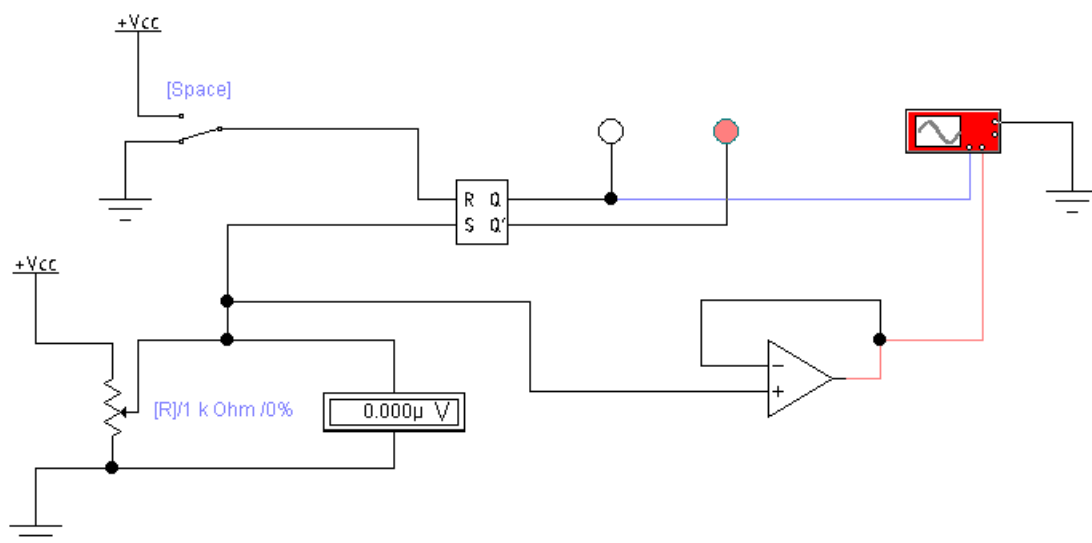


Рис. 12. Та же схема, что и на рис. 8, только с добавлением переключателя для управления подачей сигнала на триггерный вход R

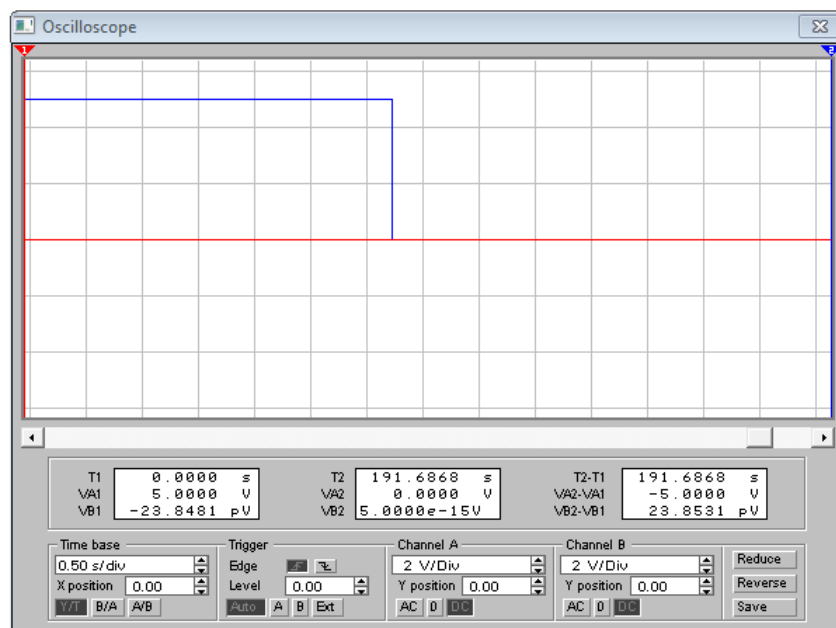


Рис. 13. Изменение входного и выходного сигнала при подаче положительного потенциала на триггерный вход R .

5. Отчет о проделанной работе

Подготовьте отчет о проделанной работе. Отчет должен содержать название заданий, скриншоты (копии экрана монитора) с результатами проведенного моделирования и сформулированные выводы.

Контрольные вопросы

1. Дайте определение триггера.
2. Назовите базовый тип триггеров. Какие входы и выходы имеет базовый тип триггера? Почему два информационных входа RS -триггера называют комплементарными?
3. Поясните работу RS -триггера и приведите его таблицу истинности.
4. Приведите схемную реализацию RS -триггера на элементах ИЛИ-НЕ.
5. Приведите схемную реализацию RS -триггера на элементах И-НЕ.
6. Поясните, о чем говорит теорема Де Моргана.
7. По каким критериям производится классификация триггеров на различные типы?
8. Перечислите основные типы триггеров в зависимости от способа управления (функционального назначения).
9. Поясните назначение и особенности работы JK -триггера.

10. Поясните назначение и особенности работы *D*-триггера.
11. Поясните назначение и особенности работы *T*-триггера.
12. На какие два типа подразделяются триггеры по способу записи информации? Поясните, в чем заключается различие асинхронного и синхронного типов триггера?
13. Какую функцию выполняет тактирующий вход синхронного триггера? Могут ли триггеры выполнять свою функцию, если они не имеют тактирующего входа?
14. Перечислите основные выводы микросхемы демultipлексора/дешифратора 74173. Укажите их назначение.
15. Поясните, как работает буферный регистр.
16. В качестве каких функциональных устройств можно использовать триггеры? Какие при этом свойства триггера используются?
17. Приведите классификацию типов памяти компьютерной информационно-вычислительной техники.