

Министерство образования и науки Российской Федерации  
Балтийский государственный технический университет «Военмех»

*Э.А. БЕСПЕРСТОВ*

# МОДЕЛИРОВАНИЕ ЦИФРОВЫХ УСТРОЙСТВ В СРЕДЕ MULTISIM 7

Лабораторный практикум

Санкт-Петербург  
2018

УДК 004.94(076)  
Б53

**Бесперстов, Э.А.**

**Б53** Моделирование цифровых устройств в среде Multisim 7: лабораторный практикум / Э.А. Бесперстов; Балт. гос. техн. ун-т; – СПб., 2018. – 79 с.

ISBN 978-5-907054-10-3

Практикум включает описание программного комплекса Multisim 7 и семь лабораторных работ, каждая из которых содержит краткие теоретические сведения, порядок проведения и требования к содержанию объекта.

Предназначено для студентов, обучающихся по специальностям “Микропроцессорная техника”, “Автоматизированные системы обработки информации и управления”, “Информационные системы и технологии” и другим инженерным специальностям.

**УДК 004.94(076)**

**Р е ц е н з е н т** канд. техн. наук, доц. БГТУ *Ю.В. Петров*

*Утверждено  
редакционно-издательским  
советом университета*

**ISBN 978-5-907054-10-3**

© Э.А. Бесперстов, 2018  
© БГТУ, 2018

## КРАТКОЕ ОПИСАНИЕ ПАКЕТА MULTISIM 7

Разработка любого радиоэлектронного устройства сопровождается физическим или математическим моделированием. Физическое моделирование связано с большими материальными затратами, так как требуется изготовление макетов и их трудоемкое исследование. Иногда чисто физическое моделирование просто невозможно из-за сложности устройства. В этом случае прибегают к математическому моделированию с использованием средств и методов вычислительной техники.

Наиболее простой и легко осваиваемой программой, содержащей блок логического моделирования цифровых устройств, является программа Electronics Workbench (EWB) канадской компании Interactive Image Technologies. Особенностью программы является наличие в ней контрольно-измерительных приборов, по внешнему виду, органам управления и характеристикам максимально приближенных к их промышленным аналогам. Опыт использования программы в практических занятиях по ряду предметов показывает, что достаточно предварительного ознакомления с ней в течение двух часов для проведения лабораторных работ.

### Окно программы Multisim 7

**Окно программы** (рис. 1) содержит шкалу меню (вторая строка окна): *File, Edit, View, Place, Simulate, Transfer, Tools, Reports, Options, Window, Help*. Ниже описаны некоторые из них.

Меню *Simulate* содержит команды:

*Run* – запуск моделирования,

*Pause* – временный останов моделирования,

*Default Instrument Setting* – установка по умолчанию режима работы контрольно-измерительных приборов,

*Default Simulation Setting* – выбор идеального (при использовании компонентов типа Virtual) или реального режимов работы моделирования,

*Instruments* – список контрольно-измерительных приборов (линейка этих приборов расположена слева в окне),

*Analyses* – список команд моделирования.

Меню *Options* (настройка программы), содержит команды:

*Preferences* – пользовательские настройки (назначение команд описаны ниже),

*Modify Title Block* – внесение данных в разделы штампа,

*Global Restrictions* – установка общего пароля,  
*Circuit Restrictions* – установка атрибутов только Read-only.

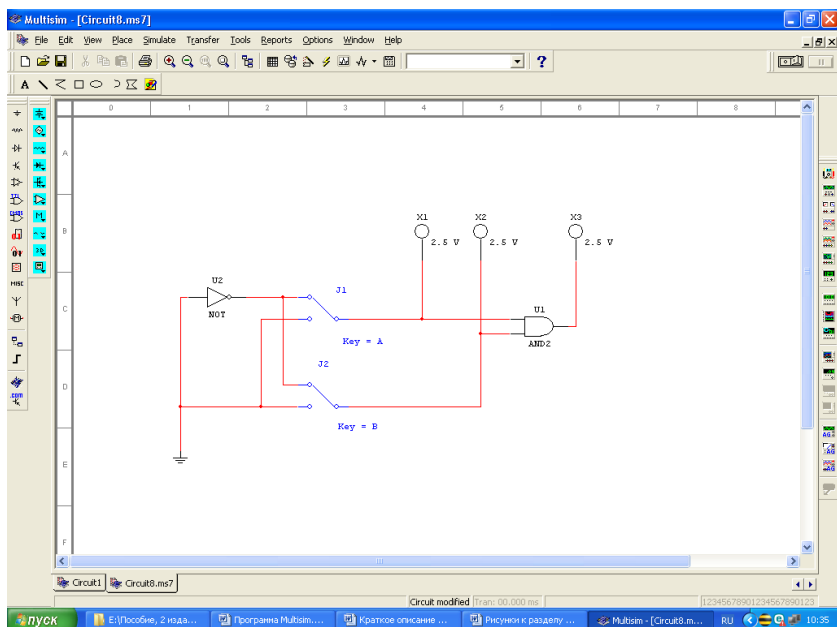




Рис. 1. Окно схемного редактора программы Multisim 7

Третья строка окна содержит:

- восемь мнемонических кнопок общесистемного характера (System);
- две кнопки, с помощью которых можно увеличить или уменьшить масштаб изображения (Zoom);
- девять специальных кнопок, дублирующих наиболее часто используемые команды (Design);
- выпадающий список использованных в текущей схеме компонентов (In Use List);
- кнопку пауза , временно приостанавливающую процесс моделирования;
- переключатель , запускающий и останавливающий процесс моделирования.

Слева в окне расположена вертикальная панель библиотек компонентов (базы данных – *Component Bars*), справа в окне –

вертикальная панель контрольно-измерительных приборов (*Instruments*).

Кнопки можно сделать видимыми или невидимыми, если вызвать команду *View > Toolbars* и во всплывающем меню установить или нет соответствующие флажки: *System, Zoom, Design, In Use List*. Здесь и далее в данном тексте на первом месте указывается меню из строки меню окна редактора схем, на втором месте – команда из выпадающего меню, появляющегося при выборе кнопки конкретного меню.

Панель библиотек компонентов можно сделать видимой или нет, если вызвать команду *View > Component Bars* и во всплывающем меню установить или нет флажок *Multisim Database*.

Панель контрольно-измерительных приборов можно сделать видимой или нет, если вызвать команду *View > Toolbars* и во всплывающем меню установить или нет флажок *Instruments*.

Панель библиотек компонентов (базы данных – *Component Bars*) содержит:

- источники (*Sources*);
- пассивные компоненты и коммутационные устройства (*Basic*);
- диоды (*Diodes*);
- транзисторы (*Transistors*);
- аналоговые микросхемы (*Analog*);
- цифровые микросхемы TTL серии (*TTL*);
- цифровые микросхемы КМОП серии (*CMOS*);
- одиночные цифровые схемы, АЛУ, регистры, счетчики, мультиплексоры, дешифраторы, ОЗУ и т.п. (*Misc Digital*);
- микросхемы смешанного типа (*Mixed*);
- индикаторные устройства (*Indicators*);
- компоненты смешанного типа (*Misc(ellaneous)*);
- аналоговые вычислительные устройства (*Controls*);
- радиочастотные компоненты (*RF*);
- электромеханические элементы (*Electro\_Mechanical*).

Панель контрольно-измерительных приборов (*Instruments*) содержит:

- цифровой мультиметр (*Multimeter*);
- функциональный генератор (*Function Generator*);
- измеритель активной мощности и коэффициента мощности (*Wattmeter*);
- осциллограф (*Oscilloscope*);
- измеритель АЧХ и ФЧХ (*Bode Plotter*);
- генератор слова (*Word Generator*);

- логический анализатор (*Logic Analyzer*);
  - логический преобразователь (*Logic Converter*);
  - измеритель нелинейных искажений в диапазоне частот от 20 до 200000 Гц (*Distortion Analyzer*);
  - спектральный анализатор (*Spectrum Analyzer*);
  - прибор для анализа электрических цепей в обобщенном виде – в виде четырехполюсников, имеющих два входа и два выхода (четыре полюса) (*Network Analyzer*).
- Последние три прибора недоступны.

## Редактор схем

Окно редактора схем предназначено для создания и редактирования принципиальных схем устройства. По умолчанию цвет фона окна черный. Изменить установки по умолчанию можно, выбрав команду *Options > Preferences*. После выбора данной команды открывается диалоговое окно *Preferences* (рис. 2).

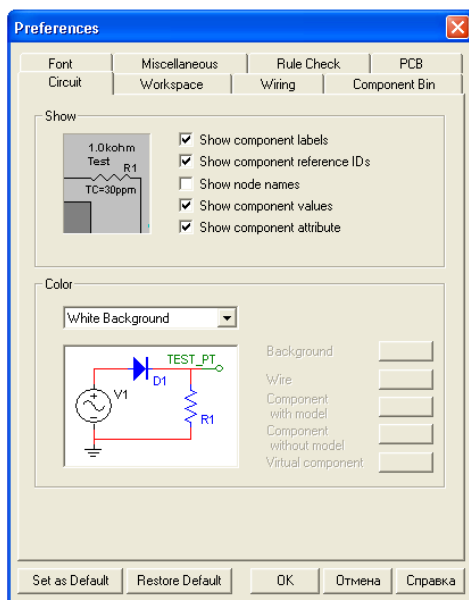


Рис. 2. Диалоговое окно *Preferences* с вкладкой *Circuit*

Диалоговое окно *Preferences* содержит шесть вкладок, на которых можно установить необходимые опции. При выводе этого окна активной является вкладка *Circuit*.

На вкладке *Circuit* имеются две панели: *Show* и *Color*.

Панель *Show* содержит окно просмотра, в котором можно увидеть установленные опции, и шесть флажков, с помощью которых устанавливаются необходимые опции.

Флажок *Show component labels* устанавливает видимость или невидимость позиционного обозначения компонента на схеме. Для того чтобы установить видимость, надо щелкнуть левой клавишей мыши по белому квадрату, расположенному слева от имени флажка. Если флажок выбран, то в белом квадрате появляется галочка.

Флажок *Show component reference IDs* устанавливает видимость на схеме порядкового номера компонента, флажок *Show node names* – видимость на схеме имени узла, флажок *Show component values* – видимость номинала компонента, флажок *Show component attribute* – видимость таблицы свойств компонента. Флажок *Adjust component identifiers* позволяет при упорядочивании схемы изменять идентификационные номера однотипных компонентов, присвоенных программой автоматически в порядке их установки.

Панель *Color* содержит выпадающий список, содержащий список команд, устанавливающих цвет фона, надписей и компонентов, окно просмотра, в котором можно просмотреть выбранный режим установки цвета, и пять кнопок, окрашенных в различные цвета, с помощью которых можно установить цвет для фона (*Background*), проводника (*Wire*), компонента, имеющего математическую модель (*Component with model*), компонента, не имеющего математической модели (*Component without model*) и виртуального компонента (*Virtual component*).

На вкладке *Workspace* (рис. 3) определяются настройки текущего окна.

На панели *Show* в окне, расположенном слева на этой панели, отображаются результаты установки опций с помощью флажков, расположенных справа от окна. Самый верхний флажок устанавливает, показывать или нет сетку (*Show grid*), второй сверху флажок – показывать или нет границы страницы (*Show page bounds*), третий флажок – показывать или нет штамп страницы (*Show title block*).

На панели *Sheet size* (размер страницы схемы) в выпадающем списке выбирается формат страницы (A3, A4,...). На панели *Orientation* устанавливается ориентация страницы: портрет или альбомная (*Landscape*), на панели *Custom size* – ширина и высота страницы в дюймах или сантиметрах, на панели *Zoom level* – масштаб изображения.

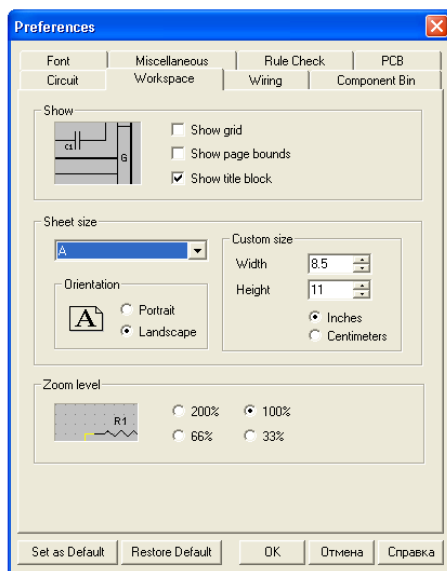


Рис. 3. Диалоговое окно *Preferences* с вкладкой *Workspace*

На вкладке *Wiring* (рис. 4) устанавливается ширина проводников и степень автоматизации их разводки, на панели *Wire width (drawing option)* – ширина линии. Результат отображается в окне на этой панели.

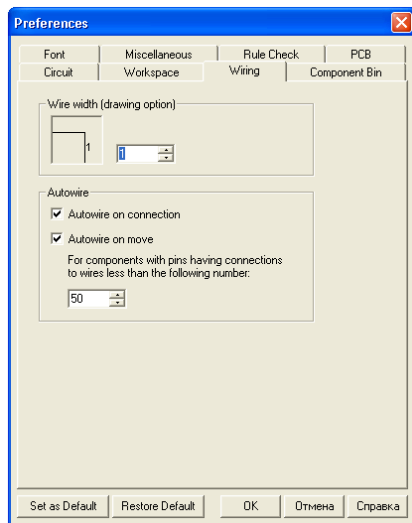


Рис. 4. Диалоговое окно *Preferences* с вкладкой *Wiring*



На панели *Autowire* устанавливается или нет автоматическое соединение проводников (флажок *Autowise on connection*) и автоматическое движение (флажок *Autowise on move*).

На вкладке *Component Bin* (рис. 5) на панели *Symbol standard* устанавливается стандарт изображения компонентов *ANCI* (США) или *DIN* (Европа).

На панели *Component toolbar functionality* устанавливается, изменять или нет цвет изображения компонента при его выборе.

На панели *Place component mode* выбирается режим размещения компонентов: разместить единственный компонент (переключатель *Place single component*), непрерывное размещение многосекционного компонента (переключатель *Continuous placement for multi-section part only*), непрерывное размещение (переключатель *Continuous*).

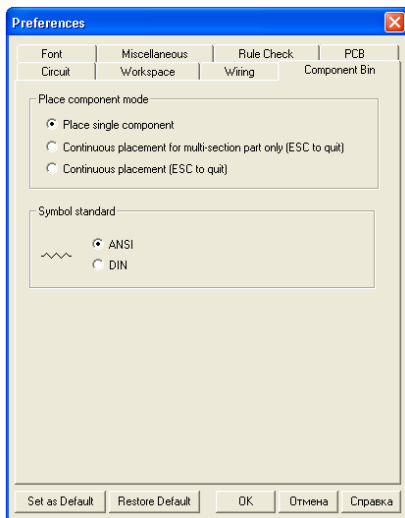


Рис. 5. Диалоговое окно *Preferences* с вкладкой *Component Bin*

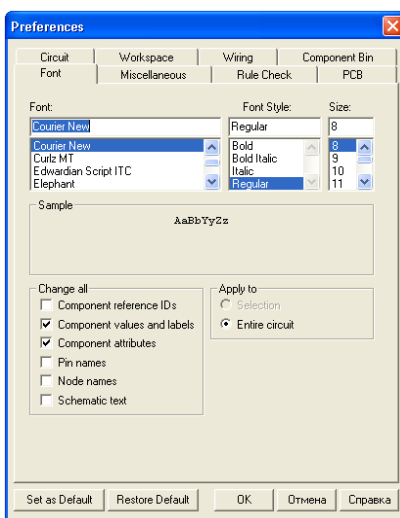


Рис. 6. Диалоговое окно *Preferences* с вкладкой *Font*

На вкладке *Font* (шрифт) (рис. 6) выбирается шрифт и его атрибуты для выбранных компонентов схемы.

На вкладке *Miscellaneous* (разное) рис. 7 устанавливается режим автосохранения данных, пути к рабочему каталогу, идеальный или реальный режим моделирования (переключатели *Ideal* и *Real* на панели *Digital Simulation Setting*) цифровых схем, соединить или рассоединить аналоговое и цифровое заземление (флажок *Connect ground to analog ground* на панели *PCB Ground Option*).

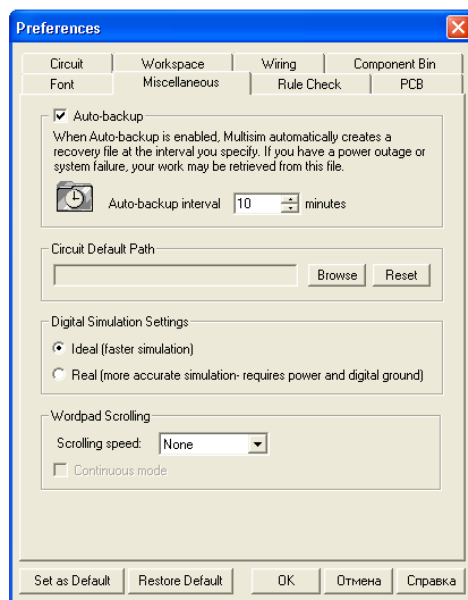


Рис. 7. Диалоговое окно *Preferences* с вкладкой *Miscellaneous*

Кроме того, окно *Preferences* содержит пять кнопок.

Для того чтобы записать выбранные опции для текущей схемы, надо щелкнуть левой клавишей мыши по кнопке *OK*.

Для того чтобы записать выбранные опции как устанавливаемые по умолчанию, надо щелкнуть левой клавишей мыши по кнопке *Set as Default* затем по кнопке *OK*.

Для того чтобы вернуться к предыдущим опциям, установленным по умолчанию, надо щелкнуть по кнопке *Restore Default*. Затем можно продолжить устанавливать новые опции.

Для того чтобы остановить изменения и закрыть диалоговое окно, надо щелкнуть по кнопке *Cancel*.

## Создание новой схемы

В этом разделе описывается процесс создания схемы, как поместить компоненты на схему и соединить их проводниками.

Создание новой схемы будет рассмотрено на примере схемы, исследующей логическую схему 2И (рис. 8).

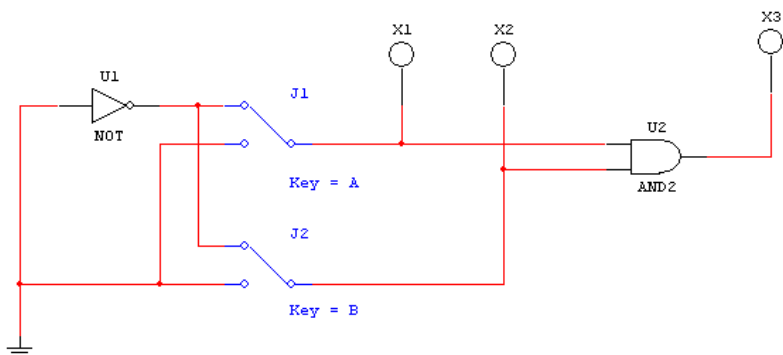


Рис. 8. Схема исследования логической схемы 2И

Прежде чем начать создание схемы, надо запустить программу *Multisim*. Для этого в окне *Windows* нажать на кнопку *Пуск*, в открывшемся меню выбрать команду *Программы*. В новом открывшемся меню выбрать строку *Multisim*. Автоматически откроется окно программы, в котором будет расположена страница новой схемы. На этой странице и будет создана схема исследования логической схемы 2И.

### ***Размещение компонентов на странице схемы***

Для размещения компонентов используют кнопки панели инструментов, расположенной вертикально на левой стороне окна. По умолчанию эта панель видима. Если она невидима, то надо нажать кнопку *Components* на строке инструментов *Design Bar*. Компоненты, необходимые для создания схемы, сгруппированы в логические разделы (*Parts Bin*). Каждому логическому разделу соответствует кнопка на шкале инструментов. При нажатии на одну из этих кнопок открывается соответствующее диалоговое окно *Select a component*. Разместить компоненты можно также с помощью команды *Place > Place Component*.

Разместите на странице схемы компонент источника постоянного напряжения +5v.

1. Нажать на кнопку *Sources*. Откроется панель, содержащая источники напряжения или тока (рис. 9). Удерживая курсор на кнопке, не щелкая по ней, можно увидеть название кнопки.

2. Это окно содержит пять белых окон: *Family*, *Component*, *Symbol[ANSI]*, *Model Manuf. VD:*, *Footprint Manuf.\Type*.

3. В окне *Family* выбрать строку *POWER\_SOURCE*.

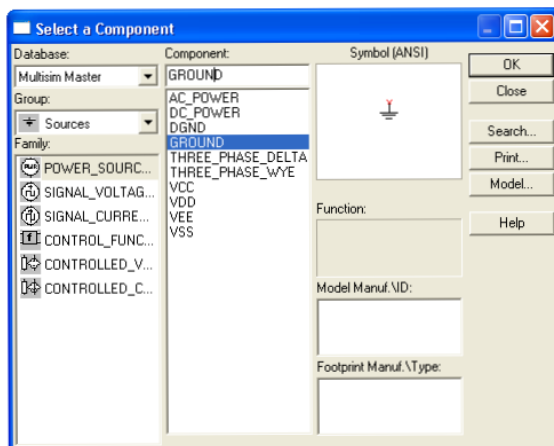


Рис. 9. Панель инструментов *Sources*

4. В окне *Component* выбрать строку *Vcc*.
5. В окне *Symbol[ANSI]* появится изображение этого элемента.
6. Щелкнуть по кнопке *OK*. Курсор примет форму выбранного компонента. Поместить выбранный компонент на схему и щелкнуть левой клавишей мыши по точке схемы, где должен быть расположен компонент, изображение компонента появится на схеме.

Если необходимо изменить свойства компонента: порядковый номер, имя и величину значения, то надо дважды щелкнуть по изображению компонента. Откроется диалоговое окно *Digital Power* (рис. 10).

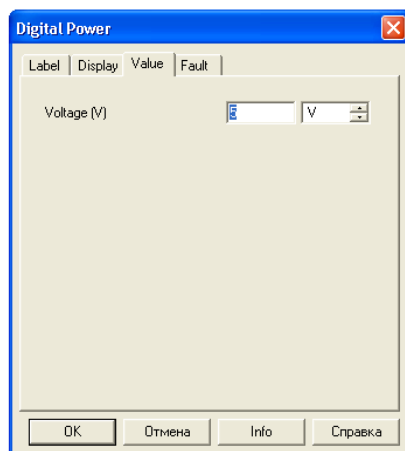


Рис. 10. Диалоговое окно *Digital Power* с вкладкой *Value*

Название окна зависит от назначения компонента. Это окно содержит четыре вкладки. По умолчанию открыта вкладка *Value*.

На вкладке *Value* расположены текстовая строка и выпадающий список.

По умолчанию значение напряжения источника равно 12v. Напряжение питания микросхем серии 74xxXX равно 5v. Поэтому надо в текстовой строке вкладки вместо цифры 12 ввести цифру 5, затем щелкнуть по кнопке *OK*.

В выпадающем списке указаны единицы измерения.

На вкладке *Lable* (рис.11) в текстовой строке *Reference ID* указывается название компонента.

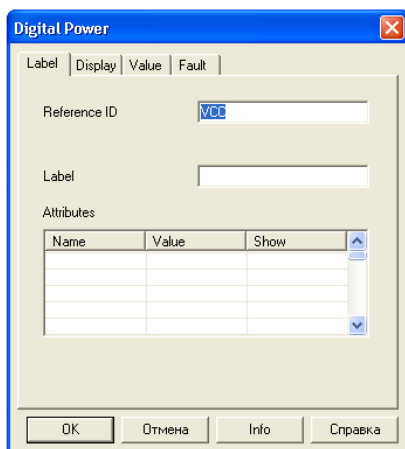


Рис. 11. Диалоговое окно *Digital Power* с вкладкой *Label*

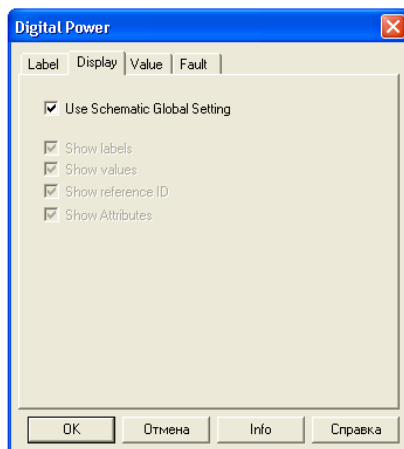


Рис. 12. Диалоговое окно *Digital Power* с вкладкой *Display*

На вкладке *Display* имеется пять флажков (рис. 12).

Флажок *Use Schematic Option global setting* устанавливает видимость всех надписей компонента, флажок *Show labels* – видимость ярлыка компонента, флажок *Show values* – видимость величины значения компонента, флажок *Show reference ID* – видимость порядкового номера компонента, флажок *Show Attributes* – видимость таблицы свойств.

Если установлен флажок *Use Schematic Global Setting*, то остальные флажки имеют серый цвет и недоступны.

На вкладке *Fault* (погрешность) (рис. 13) устанавливают границы погрешности.

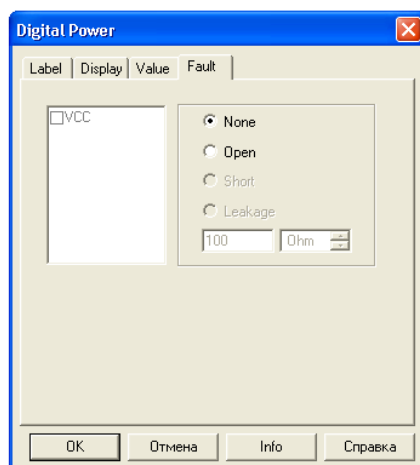



Рис. 13. Диалоговое окно *Digital Power* с вкладкой *Fault*

Разместить на схеме один компонент НЕ и один компонент 2И.

1. Нажать кнопку *Miscellaneous Digital* . Откроется окно выбора компонентов *Select a Component* с соответствующими компонентами (рис. 14).

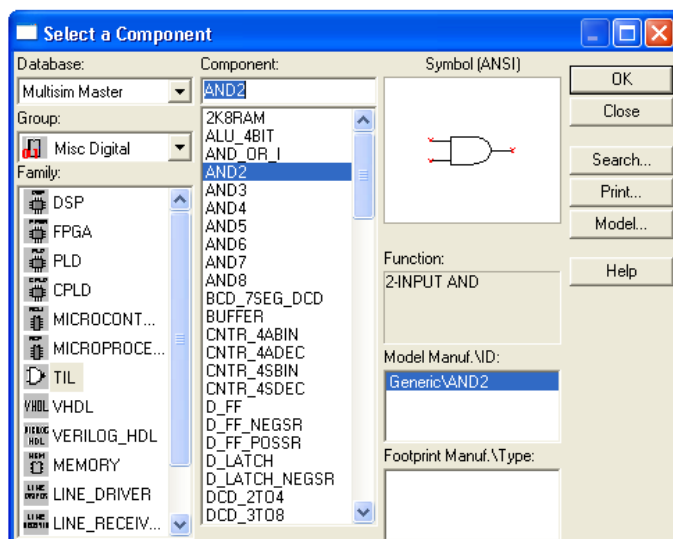


Рис.14. Окно выбора компонентов *Select a Component*

В этом окне в текстовой строке *Group* отображается изображение



кнопки и название группы библиотек компонентов *Misc Digital*.

Окно *Select a Component* содержит пять просмотровых окон:

*Family*,

*Component*,

*Symbol (ANCI)*,

*Model Manuf.ID:*,

*Footprint Manuf/Type:*.

В просмотрном окне *Family* расположен список библиотек, входящих в данную группу.

2. В просмотрном окне *Family* выбрать библиотеку *TIL*. В просмотрном окне *Component* отобразится список компонентов, входящих в выбранную библиотеку.

3. В просмотрном окне *Component* выбрать компонент *AND2* (2И). Изображение выбранного компонента появится в просмотрном окне *Symbol (ANCI)*. Изображение компонента может быть в формате *ANSI* (США) или *DIN* (Европа).


4. Поместить его на схему, щелкнув по кнопке *OK*. Ускорить поиск компонента можно, введя несколько первых символов имени компонента в строку *Component*.

5. В списке компонентов выбрать компонент *NOT* (НЕ) и поместить один образец этого компонента на схему.

Поместить на схему несколько копий одного и того же компонента можно двумя способами:

1) повторить несколько раз действия, описанные в п. 3,

2) воспользоваться списком компонентов, размещенных на схеме ранее. Для этого воспользоваться списком *In Use List* в строке *System toolbar*. Открыть список, нажав на черный треугольник, расположенный в окне списка справа. Выбрать в этом списке нужный компонент и поместить его на схему.

Разместить на схеме два переключателя на два положения. Для этого нажать кнопку *Basic* . Откроется окно выбора компонентов (*Select a Component*) с соответствующими компонентами (рис. 15).

1. В окне *Family* выбрать строку *SWITCH*.

2. В окне просмотра *Component* выбрать компонент *SPDT* и поместить два его образца на схему (как описано выше).

3. При необходимости повернуть эти образцы компонента на нужный угол. Размещенные на схеме компоненты можно вращать по

горизонтали или по вертикали или на 90° вправо или влево. Чтобы сделать это, необходимо:

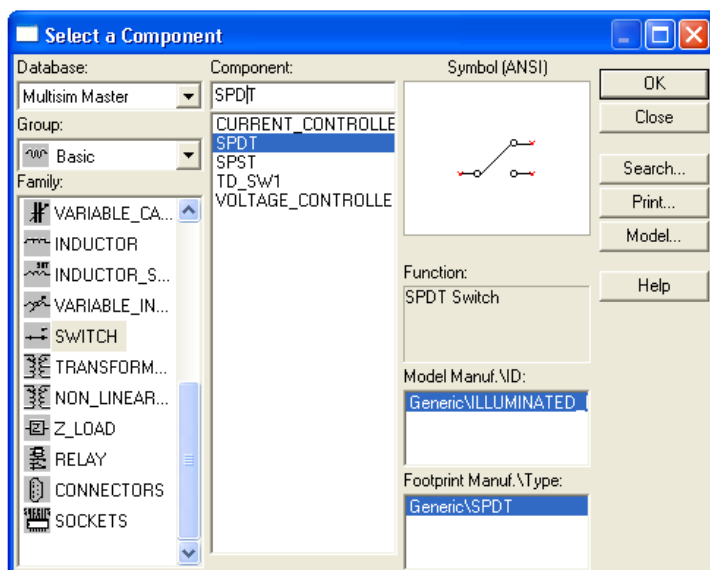


Рис. 15. Панель инструментов *Select a Component*

- 1) щелкнуть правой клавишей мыши по компоненту. Откроется всплывающее меню (рис. 16);
- 2) в этом меню выбрать нужную строчку и щелкнуть левой клавишей мыши по этой строке.

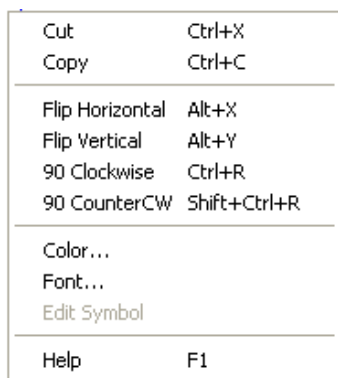


Рис. 16. Всплывающее меню



Компонент *SPDT* управляется клавишей, по умолчанию *KEY = SPACE*. Если необходимо заменить управляющую клавишу, то надо дважды щелкнуть по изображению компонента. Откроется диалоговое окно *SWITCH* (рис.17).

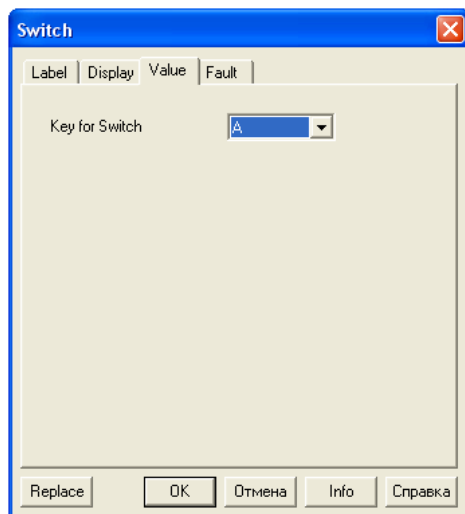


Рис. 17. Диалоговое окно *SWITCH* *Key for Switch*

В этом окне имеется строка *Key for Switch*, если щелкнуть по черному треугольнику, расположенному справа в этой строке, то появится всплывающий список, содержащий буквы английского алфавита (рис.18).

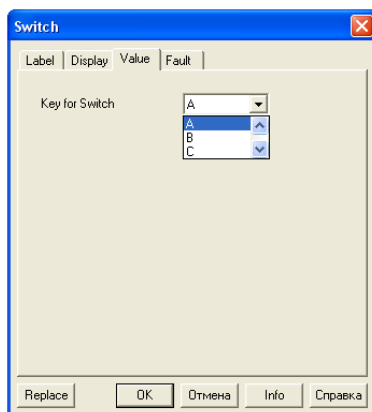



Рис. 18. Выпадающее меню

Выбрать в этом списке нужную букву и щелкнуть по кнопке ОК. Управляющая клавиша изменит название.

Разместить на схеме три светодиода. Для этого необходимо:

1. Нажать кнопку *Indicators* . Откроется панель *Select a Component*. (рис.19).

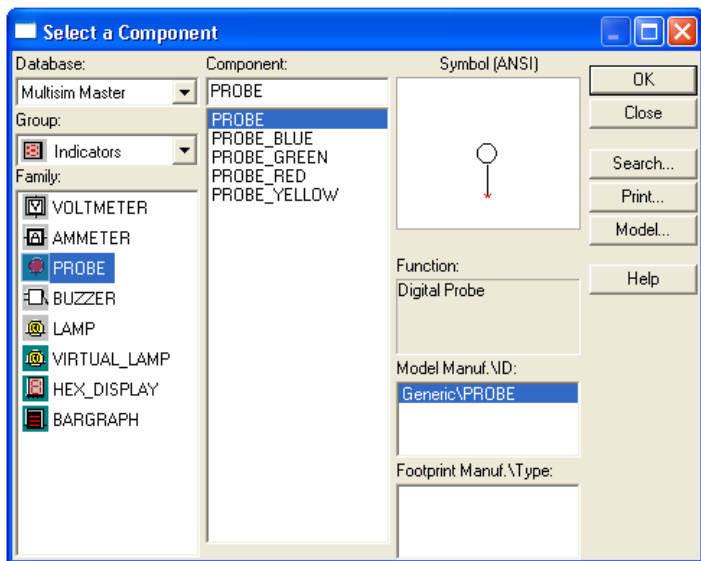


Рис. 19. Панель инструментов *Select a Component*


2. Из списка *Family* выбрать строку *Probe* и в окне *Component* выбрать строку *Probe*. Этот элемент может быть белого, голубого, зеленого, красного или желтого цвета.

3. В окне *Symbol[ANSI]* появится изображение этого компонента. Поместить три образца этого компонента на схему.

4. При необходимости повернуть эти образцы на нужный угол (как описано выше).

Поместить на схему несколько копий одного и того же компонента можно двумя способами: повторить несколько раз действия, описанные в п. 3, воспользоваться списком компонентов, размещенных на схеме ранее. Для этого открыть список *In Use List*, в строке *System toolbar*, нажав на черный треугольник, расположенный в окне списка справа. Выбрать в этом списке нужный компонент и поместить его на схему.

Разместить на схеме изображение компонента «земля».

1. Нажать на кнопку *Sources* . Откроется панель *Select a Component*, содержащая источники напряжения или тока (рис. 20) и символы «земля».

2. В окне *Family* выбрать строку *POWER\_SOURS*.

3. В окне *Component* выбрать строку *GROUND*.

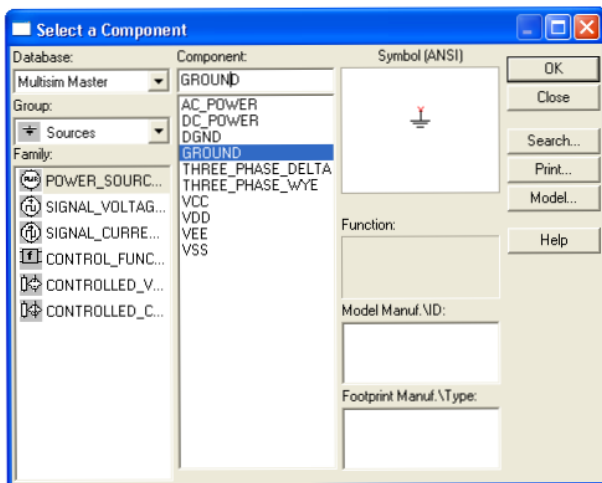


Рис. 20. Панель инструментов *Select a Component*

4. В окне *SYMBOL[ANSI]* появится изображение символа «земля».

5. Поместить на схему символ «земля», нажав кнопку *OK*.

В результате страница схемы будет выглядеть так, как показано на рис. 21.

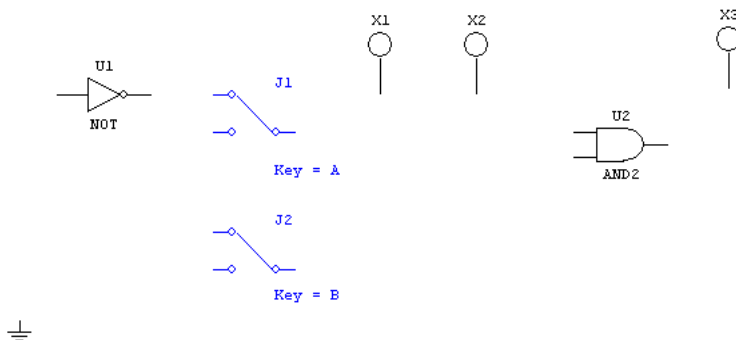


Рис. 21. Страница схемы с размещенными на ней компонентами

Используя команду *File > Save*, сохранить страницу схемы с размещенными на ней компонентами.

### ***Соединение компонентов проводниками***

После того как компоненты размещены на схеме, их необходимо соединить проводниками. Все компоненты имеют выводы, которые используются для этих целей. Соединить компоненты проводниками можно двумя способами: автоматически и вручную. При автоматическом соединении программа выбирает наилучший путь между двумя соединяемыми выводами компонентов. При автоматическом соединении программа избегает размещения проводников через другие компоненты или частичного перекрывания проводников. При ручном соединении компонентов пользователь должен сам контролировать путь проводника на схеме. При разводке одного проводника можно комбинировать эти два метода, например, начать разводку проводника вручную и затем позволить программе закончить ее.

Для того чтобы соединить два компонента автоматически, необходимо:

1. Щелкнуть по выводу первого компонента, например компонента «земля». Изображение курсора изменится на крест, показывая, что программа находится в режиме разводки.

2. Щелкнуть по первому выводу второго компонента логической схемы 2И ( $U_2$ ), к которому необходимо присоединить проводник. Соединение этих двух выводов первого и второго компонентов произойдет автоматически. После того как два вывода соединятся между собой, изображение курсора примет первоначальный вид. После этого можно выполнять очередные команды.

3. Повторять эти шаги до тех пор, пока все выводы всех компонентов не будут соединены нужным образом.

Чтобы удалить проводник, соединяющий два вывода, надо щелкнуть по проводнику правой клавишей мыши и во всплывающем меню выбрать команду *Delete*, или нажать клавишу *Delete*.

Для более точного определения пути проводника, соединяющего два вывода, используется ручная разводка. Программа предотвращает возможность присоединить два проводника к одному выводу. Это исключает ошибки соединения. Поэтому начинать разводку проводника надо не на выводах компонентов, а в какой-нибудь точке

проводника, соединяющих эти выводы. Для этого надо добавить точку соединения (*junction*) в данной точке.

Для того чтобы добавить точку соединения двух проводников, необходимо:

1. Выбрать команду *Place > Place Junction*. В точке проводника, на которой остановлен курсор, появится легкое изображение точки.

2. Передвинуть изображение точки в нужное место проводника и щелкнуть левой клавишей мыши. На проводнике появится точка.

Соединять выводы компонентов проводниками вручную удобней, если в редакторе схемы видна сетка. Для того чтобы сделать сетку видимой, если она невидима, надо щелкнуть правой клавишей мыши в какой-нибудь точке редактора схемы и из всплывающего меню выбрать команду *Show Grid*.

Для того чтобы выполнить соединение двух выводов вручную, необходимо:

1. Щелкнуть по точке соединения, только что размещенной на проводнике. Изображение курсора измениться на крест, показывая, что программа находится в режиме разводки проводников.

2. Перетащить курсор к выводу другого компонента и щелкнуть левой клавишей мыши. Это фиксирует проводник на его месте.

3. Если нужно изменить направление перемещения проводника, надо щелкнуть левой клавишей мыши в точке изменения направления движения. Маленькие черные квадраты показывают точки изменения направления движения.


4. Если необходимо переместить проводник на другое место, надо щелкнуть в любой точке проводника и передвинуть этот сегмент проводника.







## Режим моделирования схемы

После того как нарисована принципиальная схема исследуемого устройства, необходимо запустить процесс моделирования, чтобы исследовать поведение устройства.



Для того чтобы запустить процесс моделирования, надо

щелкнуть по кнопке *Simulate*  и из всплывающего меню

выбрать команду *Run* или на переключателе  выбрать состояние 1. Слева от команды *Run* появится галочка, означающая, что процесс моделирования начался.


Для того чтобы приостановить на время процесс моделирования, надо щелкнуть по кнопке Simulate  и из всплывающего меню выбрать команду Pause или на переключателе  выбрать кнопку . Чтобы возобновить процесс моделирования, надо щелкнуть по кнопке Simulate  и из всплывающего меню снова выбрать команду *Pause* или на переключателе  выбрать кнопку .

Процесс моделирования возобновится с того момента, когда он был остановлен.

Для того чтобы остановить процесс моделирования, надо щелкнуть по кнопке Simulate  и из всплывающего меню выбрать команду *Run* или на переключателе  выбрать состояние 0.

Галочка слева от команды *Run* исчезнет, показывая, что процесс моделирования остановлен. Если после остановки процесса моделирования запустить его снова, то в отличие от действия команды *Pause* процесс моделирования начнется из начальной точки.

Процесс моделирования можно запустить или приостановить с помощью команд *Simulation > Run* и *Simulation > Pause* соответственно.

Процесс моделирования можно запускать и останавливать с помощью переключателя , который можно сделать видимым или невидимым с помощью команды *View > Show Simulation Switch*.

## ЛАБОРАТОРНАЯ РАБОТА №1

### ИССЛЕДОВАНИЕ ЛОГИЧЕСКИХ СХЕМ

**Цель работы** – исследовать логические схемы И, И-НЕ, ИЛИ, ИЛИ-НЕ, исключающее ИЛИ.

## Краткие сведения из теории

### Аксиомы алгебры логики

Переменные, рассматриваемые в алгебре логики, могут принимать только два значения 0 или 1. В алгебре логики определены:

- отношение эквивалентности, обозначаемое знаком  $=$ ;
- операция сложения (дизъюнкция), обозначаемая знаком  $+$  или  $\vee$ ;
- операция умножения (конъюнкция), обозначаемая знаком  $\&$  или  $*$ ;
- операция отрицания (или инверсия), обозначаемая знаком надчеркивания или апострофом  $'$ .

Алгебра логики определяется следующей системой аксиом:

$$\begin{array}{ll} x = 0, \text{ если } x \neq 1, & \bar{0} = 1, \\ x = 1, \text{ если } x \neq 0; & \bar{1} = 0; \\ 1 + 1 = 1, & 0 * 0 = 0, \\ 0 + 0 = 0, & 1 * 1 = 1, \\ 0 + 1 = 1 + 0 = 1; & 1 * 0 = 0 * 1 = 0. \end{array}$$

### Логические выражения

Логические выражения связывают значение логической функции со значениями логических переменных.

Логические выражения могут записываться или в конъюнктивной, или дизъюнктивной нормальных формах. В дизъюнктивной форме логические выражения записываются как логическая сумма логических произведений, в конъюнктивной форме – как логическое произведение логических сумм. Порядок действий в логических выражениях такой же, как и в обычных алгебраических выражениях. Логические выражения связывают значение логической функции со значениями логических переменных.

### Законы булевой алгебры

Законы булевой алгебры вытекают из аксиом и имеют две формы выражения: для конъюнкции и дизъюнкции. Эти законы используются при преобразованиях логических выражений:

переместительный закон:  $xy = yx$ ;  $x + y = y + x$ ;

сочетательный закон:  $x(yz) = (xy)z = xyz$ ;

$$x + (y + z) = (x + y) + x = x + y + x;$$

распределительный закон:  $x(y + z) = xy + xz$ ;

$$x + yz = (x + y)(x + z);$$

закон повторения:  $x + x = x$ ;  $x * x = x$ ;

закон обращения: если  $x = y$ , то  $\bar{x} = \bar{y}$ ;

закон двойной инверсии:  $\bar{\bar{x}} = x$ ;

закон универсального множества:  $x * 1 = x$ ;  $x + 1 = 1$ ;

закон дополнительности:  $x \bar{x} = 0$ ;  $x + \bar{x} = 1$

закон нулевого множества:  $x * 0 = 0$ ;  $x + 0 = x$ ;

закон поглощения:  $x + x * y = x$ ;  $xy + x \bar{y} = x$ ;

закон склеивания:  $(x + y)(x + \bar{y}) = x$   $xy + x \bar{y} = x$ ;

закон инверсии (закон Де-Моргана):  $\overline{xy} = \bar{x} + \bar{y}$ ;  $\bar{\bar{x} + \bar{y}} = xy$ .

## Логические функции

Любое логическое выражение, составленное из  $n$  переменных  $x_n, x_{n-1}, \dots, x_1$  с помощью конечного числа операций алгебры логики, можно рассматривать как некоторую функцию  $n$  переменных. Такую функцию называют логической. В соответствии с аксиомами алгебры логики функция может принимать в зависимости от значения переменных значение 0 или 1. Функция  $n$  логических переменных может быть определена для  $2^n$  значений переменных, соответствующих всем возможным значениям  $n$ -разрядных двоичных чисел.

Основной интерес представляют следующие функции двух переменных  $x$  и  $y$ :

$f_1(x, y) = x * y$  – логическое умножение,

$f_2(x, y) = x + y$  – логическое сложение,

$f_3(x, y) = \overline{x * y}$  – логическое умножение с инверсией,

$f_4(x, y) = \overline{x + y}$  – логическое сложение с инверсией,



$f_5(x,y) = x \oplus y = \overline{x}y + x\overline{y} = \overline{xy} + \overline{\overline{xy}} = \overline{xy + \overline{xy}}$  – суммирование по модулю два или исключающее ИЛИ,

$f_6(x,y) = \overline{x \oplus y} = xy + \overline{xy} = \overline{xy + \overline{xy}}$  – равнозначность.

## Логические схемы

Физическое устройство, реализующее одну из операций алгебры логики или простейшую логическую функцию, называется логическим элементом. Схема, составленная из конечного числа логических элементов по определенным правилам, называется логической схемой. Основным логическим функциям соответствуют выполняющие их схемные элементы. Например, функции  $f_1(x,y)$  соответствует логическая схема И, функции  $f_2(x,y)$  – логическая схема ИЛИ, функции  $f_3(x,y)$  – логическая схема И-НЕ, функции  $f_4(x,y)$  – логическая схема ИЛИ-НЕ.

## Таблица истинности

Так как область определения любой функции  $n$  переменных конечна (может принимать  $2^n$  значений), то такая функция может быть задана таблицей значений  $f(x)$ , которые она принимает в точках  $x_i$ , где  $i = 0, 1, \dots, 2^n - 1$ .

Такие таблицы называются таблицами истинности. В табл. 1 представлены таблицы функций  $f_1(x,y)$ , ...,  $f_6(x,y)$ .

Т а б л и ц а 1

$i$	Значения переменных		Функции					
	$x$	$y$	$f_1(x,y)$	$f_2(x,y)$	$f_3(x,y)$	$f_4(x,y)$	$f_5(x,y)$	$f_6(x,y)$
0	0	0	0	0	1	1	0	1
1	0	1	0	1	1	0	1	0
2	1	0	0	1	1	0	1	0
3	1	1	1	1	0	0	0	1

## Последовательность выполнения работы

### Исследование логической функции 2И

1. Нарисовать схему исследования функции 2И (рис. 20).

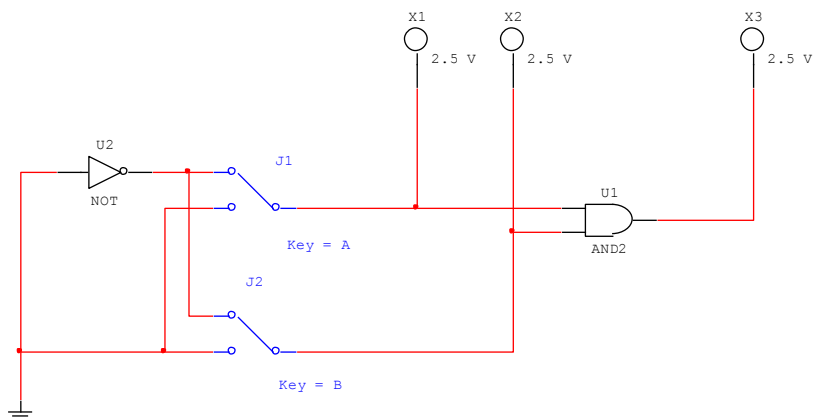



Рис. 22. Схема исследования функции 2И

Схема содержит исследуемую функцию И ( $U1$ ), два двух-позиционных переключателя ( $J1$ ,  $J2$ ), управляемые клавишами  $A$  и  $B$  (заглавные буквы латинского алфавита), источники сигналов логической единицы ( $U2$ ) и логического нуля, три светодиода ( $X1$ ,  $X2$ ,  $X3$ ).

2. Запустить процесс моделирования, нажав кнопку  на панели инструментов, и в появившемся меню выбрать команду *Run*.

3. Подать на входы схемы И все возможные комбинации уровней сигналов  $A$  и  $B$  с помощью переключателей  $J1$  и  $J2$ . Для каждой комбинации зафиксировать уровни входных сигналов  $A$  и  $B$  и уровень выходного сигнала  $Q$  (логическая единица – соответствующий светодиод  $X_i$  светится, логический ноль – соответствующий светодиод  $X_i$  не светится). Результаты измерений занести в таблицу истинности (табл. 2).

Т а б л и ц а 2

Входы		Выход
$A$	$B$	$Q$
0	0	
0	1	
1	0	
1	1	

## Исследование логической функции 2И-НЕ

1. Нарисовать схему исследования функции 2И-НЕ (рис. 23).

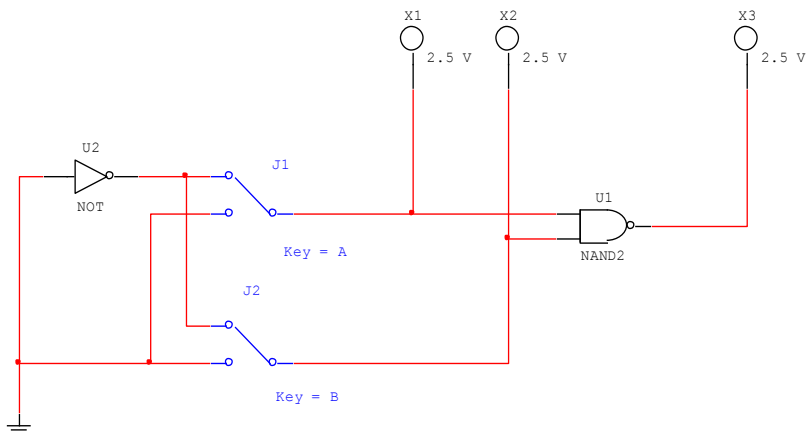



Рис. 23. Схема исследования функции 2И-НЕ

2. Запустить процесс моделирования, нажав кнопку  на панели инструментов, и в появившемся меню выбрать команду *Run*.

3. Подать на входы схемы И-НЕ все возможные комбинации уровней сигналов *A* и *B* с помощью переключателей *J1* и *J2*. Для каждой комбинации зафиксировать уровни входных сигналов *A* и *B* и уровень выходного сигнала *Q* (логическая единица – соответствующий светодиод  $X_i$  светится, логический ноль – соответствующий светодиод  $X_i$  не светится). Результаты измерений занести в таблицу истинности (табл. 3).

Т а б л и ц а 3

Входы		Выход
<i>A</i>	<i>B</i>	<i>Q</i>
0	0	
0	1	
1	0	
1	1	

## Исследование логической функции 2ИЛИ

1. Нарисовать схему исследования функции 2ИЛИ (рис. 24).

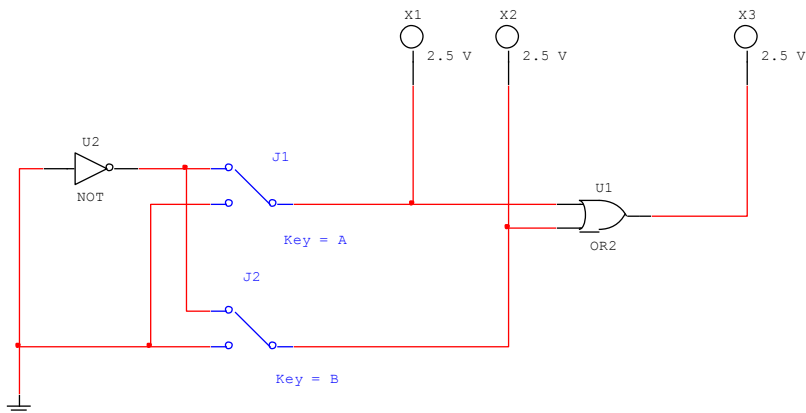



Рис. 24. Схема исследования функции 2ИЛИ

2. Запустить процесс моделирования, нажав кнопку  на панели инструментов, и в появившемся меню выбрать команду *Run*.

3. Подать на входы схемы ИЛИ все возможные комбинации уровней сигналов  $A$  и  $B$  с помощью переключателей  $J1$  и  $J2$ . Для каждой комбинации зафиксировать уровни входных сигналов  $A$  и  $B$  и уровень выходного сигнала  $Q$  (логическая единица – соответствующий светодиод  $X_i$  светится, логический ноль – соответствующий светодиод  $X_i$  не светится). Результаты измерений занести в таблицу истинности (табл. 4).

Т а б л и ц а 4

Входы		Выход
$A$	$B$	$Q$
0	0	
0	1	
1	0	
1	1	

## Исследование логической функции 2ИЛИ-НЕ

1. Нарисовать схему исследования функции 2ИЛИ-НЕ (рис. 25).

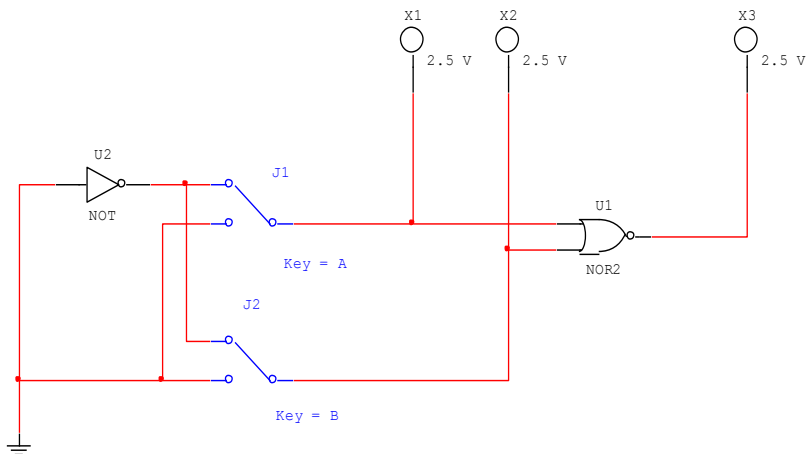



Рис. 25. Схема исследования функции 2ИЛИ-НЕ

2. Запустить процесс моделирования, нажав кнопку  на панели инструментов и в появившемся меню выбрать команду *Run*.

3. Подать на входы схемы ИЛИ-НЕ все возможные комбинации уровней сигналов *A* и *B* с помощью переключателей *J1* и *J2*. Для каждой комбинации зафиксировать уровни входных сигналов *A* и *B* и уровень выходного сигнала *Q* (логическая единица – соответствующий светодиод  $X_i$  светится, логический ноль – соответствующий светодиод  $X_i$  не светится). Результаты измерений занести в таблицу истинности (табл. 5).

Т а б л и ц а 5

Входы		Выход
A	B	Q
0	0	
0	1	
1	0	
1	1	

## **Исследование логической функции Иключающее ИЛИ**

1. Нарисовать схему исследования функции ИЛИ-НЕ (рис. 26).

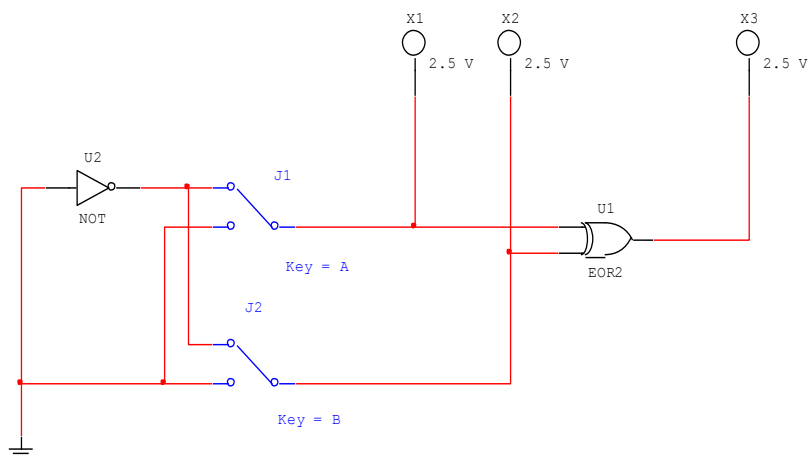



Рис. 26. Схема исследования функции Иключающее ИЛИ

2. Запустить процесс моделирования, нажав кнопку  на панели инструментов, и в появившемся меню выбрать команду *Run*.

3. Подать на входы схемы Иключающее ИЛИ все возможные комбинации уровней сигналов *A* и *B* с помощью переключателей *J1* и *J2*. Для каждой комбинации зафиксировать уровни входных сигналов *A* и *B* и уровень выходного сигнала *Q* (логическая единица – соответствующий светодиод  $X_i$  светится, логический ноль – соответствующий светодиод  $X_i$  не светится). Результаты измерений занести в таблицу истинности (табл. 6).

Т а б л и ц а 6

Входы		Выход
<i>A</i>	<i>B</i>	<i>Q</i>
0	0	
0	1	
1	0	
1	1	

## Содержание отчёта

1. Название пункта работы.
2. Исследуемая схема.
3. Результат моделирования.

### ЛАБОРАТОРНАЯ РАБОТА №2

## ИССЛЕДОВАНИЕ РЕГИСТРОВ, СЧЕТЧИКОВ И ДЕШИФРАТОРОВ

**Цель работы** – исследовать регистры, счетчики и дешифраторы.

### Краткие сведения из теории

#### Триггеры

Триггерами называются устройства, обладающие двумя устойчивыми состояниями ( $Q=1$  и  $Q=0$ ), способные находиться в одном из них сколь угодно долго и переходить из одного состояния в другое под воздействием внешних устройств. В каком из этих состояний окажется триггер, зависит от сигналов на входах триггера и от его предыдущего состояния, т.е. он имеет память. Таким образом, триггер является элементарной ячейкой памяти.

Тип триггера определяется алгоритмом его работы, в зависимости от которого триггер может иметь установочные, информационные и управляющие входы. Установочные входы устанавливают состояние триггера независимо от состояния других входов. Входы управления разрешают запись данных, подающихся на информационные входы. Наиболее распространенными являются триггеры *RS*-, *JK*-, *D*- и *T*-типов.

*RS*-триггер – простейший автомат с памятью, который может находиться в двух состояниях. Триггер имеет два установочных входа: установки *S* (*set* – установка) и сброса *R* (*reset* – сброс), на которые подаются входные сигналы от внешних источников. При подаче на установки активного логического уровня триггер устанавливается в 1 ( $Q = 1$ ,  $Q' = 0$ , здесь штрих означает инвертирование); при подаче активного уровня на вход сброса триггер устанавливается в 0 ( $Q = 0$ ,  $Q' = 1$ ). Если на оба установочных входа подать пассивный логический уровень, то триггер сохраняет предыдущее состояние выходов:  $Q = 1$  или  $Q = 0$ . Каждое состояние устойчиво и поддерживается за счет действия обратных связей.

Подача активного уровня одновременно на оба установочных входа запрещена, так как триггер не может быть установлен в ноль и единицу.

$RS$ -триггер может быть выполнен на элементах ИЛИ-НЕ или И-НЕ (рис. 27).

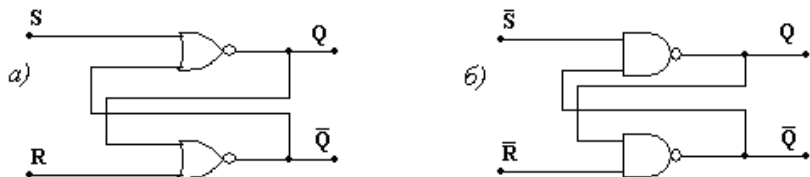


Рис. 27.  $RS$ -триггер: а – на элементах ИЛИ-НЕ, б – на элементах И-НЕ

Для  $RS$ -триггеров, выполненных на элементах ИЛИ-НЕ, активным уровнем на управляющих входах является уровень логической единицы, а для  $RS$ -триггеров, выполненных на элементах И-НЕ, активным уровнем на управляющих входах является уровень логического нуля.

$RS$ -триггер является основным узлом построения последовательных схем. Условия переходов триггеров из одного состояния в другое можно описать табличным, аналитическим или графическим способами. Табличное описание работы  $RS$ -триггера на элементах ИЛИ-НЕ и И-НЕ представлено в табл. 7 и 8 соответственно, где использованы следующие обозначения:  $Q_t$  – предшествующее состояние выхода;  $Q_{t+1}$  – новое состояние, устанавливающееся после перехода; – – неопределенное состояние.

Т а б л и ц а 7

$R$	$S$	$Q_{t+1}$
0	0	$Q_t$
1	0	0
0	1	1
1	1	–

Т а б л и ц а 8

$R$	$S$	$Q_{t+1}$
0	0	–
1	0	1
0	1	0
1	1	$Q_t$

Триггер  $JK$ -типа имеет более сложную структуру и более широкие возможности по сравнению с  $RS$ -триггером. Кроме информационных входов  $J$  и  $K$  и прямого и инверсного выходов  $Q$  и  $Q'$ ,  $JK$ -триггер имеет вход управления  $C$  (тактирующий или счетный) и два асинхронных установочных входа:  $R$  и  $S$ . Обычно активными



уровнями установочных сигналов являются нули. Установочные входы имеют приоритет над остальными входами. Активный уровень сигнала на входе  $S$  устанавливает триггер в состояние 1 ( $Q = 1$ ), а активный уровень на входе  $R$  устанавливает триггер в состояние 0 ( $Q = 0$ ), независимо от сигналов на остальных входах. Если на входы установки подать пассивный уровень сигнала, то состояние триггера будет изменяться по фронту импульса на счетном входе в зависимости от состояния входов  $J$  и  $K$ .

На рис. 28 изображен один из вариантов функциональной схемы  $JK$ -триггера и его условное графическое изображение, а на рис. 29 приведены временные диаграммы его работы при  $R = S = 1$ .

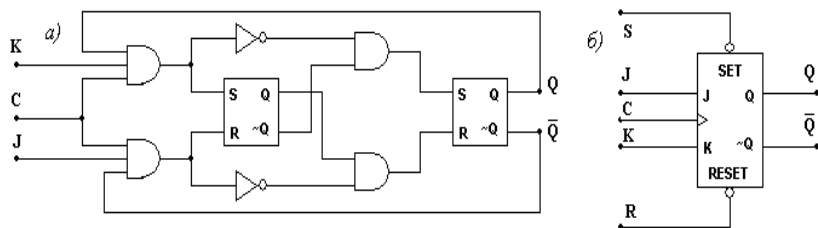


Рис. 28.  $JK$ -триггер: а – функциональная схема; б – условное графическое обозначение

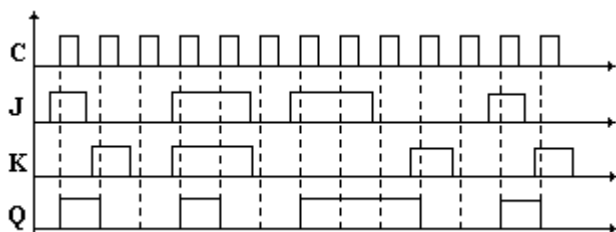


Рис. 29. Временная диаграмма работы  $JK$ -триггера

$D$ -триггер имеет один информационный вход  $D$  (*data* – данные) и один счетный вход  $C$ . Информация с входа  $D$  записывается в триггер по положительному перепаду импульса на счетном входе и сохраняется до следующего положительного перепада на счетном входе триггера. Кроме счетного  $C$  и информационного  $D$  входов триггер имеет два асинхронных установочных входа:  $R$  и  $S$ . Установочные входы имеют приоритет над остальными входами. Активный уровень сигнала на входе  $S$  устанавливает триггер в состояние 1 ( $Q = 1$ ), а активный уровень на входе  $R$  устанавливает

триггер в состояние 0 ( $Q = 0$ ), независимо от сигналов на остальных входах.

Условное обозначение  $D$ -триггера с диаграммами входных и выходных сигналов приведено на рис. 30.

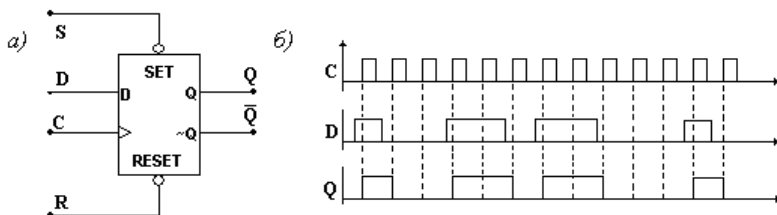


Рис. 30.  $D$ -триггер:  $a$  – условное обозначение;  $b$  – временные диаграммы

$T$ -триггер или счетный триггер – устройство, осуществляющее счетный режим. Такие схемы можно построить на основе  $JK$ - или  $D$ -триггеров. На рис. 31 представлена схема  $T$ -триггера и временная диаграмма, иллюстрирующая счетный режим.

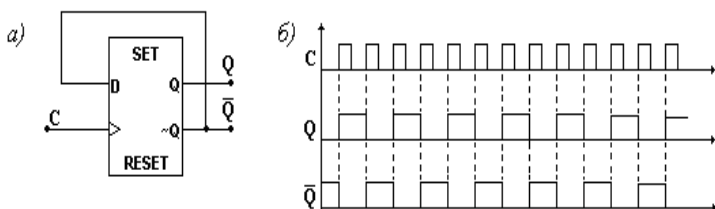


Рис. 31.  $T$ -триггер:  $a$  – условное обозначение;  $b$  – временная диаграмма

В  $D$ -триггере счетный режим реализуется при помощи обратной связи, когда на вход  $D$  подается сигнал с инверсного выхода триггера. Таким образом, всегда осуществляется неравенство сигналов на входе  $D$  и на выходе  $Q$  (если  $Q = 1$ ,  $D = 0$  и наоборот). Следовательно, при каждом положительном перепаде сигнала на счетном входе  $C$  состояние выхода будет изменяться на противоположное.

Таким образом, на каждые два входных тактовых импульса  $T$ -триггер формирует один период выходного сигнала  $Q$ , т.е. период выходного сигнала в два раза больше периода входного сигнала. Следовательно, триггер осуществляет деление частоты  $f_T$  на его входе на 2:  $f_Q = f_T/2$ , где  $f_Q$  – частота следования импульсов на выходе триггера.

## Регистры

Триггерным регистром называется совокупность триггеров с определенными связями между ними, при которых они действуют как единое устройство. Регистры выполняются на синхронных триггерах *JK*- или *D*-типа. В зависимости от выполняемых функций регистры делятся на два типа: накопительные (параллельные) и сдвигающие.

В последовательном регистре выход предыдущего триггера подается на вход следующего триггера, а тактовые импульсы подаются на входы *C* всех триггеров, составляющих регистр, одновременно (рис. 32). При этом содержимое каждого триггера записывается в последующий триггер. Такие регистры называются сдвиговыми регистрами или регистрами сдвига.

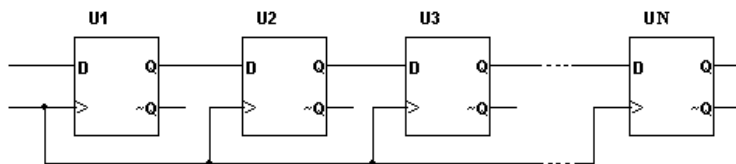


Рис. 32. Последовательный регистр (регистр сдвига)

Если на вход *D* регистра сдвига подать 1, а на вход *C* тактовую частоту, то 1 начнет продвигаться по регистру сдвига, т.е. под воздействием первого тактового импульса 1 запишется в первый триггер регистра. Под воздействием второго тактового импульса эта единица перепишется во второй триггер и т.д., пока под воздействием *N*-го тактового импульса единица не выйдет из регистра сдвига. На рис. 33 приведена временная диаграмма работы четырехразрядного регистра сдвига.

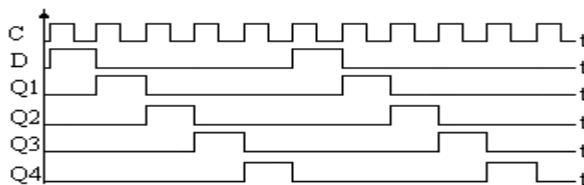


Рис. 33. Временная диаграмма четырехразрядного регистра сдвига

Регистры сдвига могут быть:

- с последовательным вводом и выводом;
- с последовательным вводом и параллельным выводом;
- с параллельным вводом и последовательным выводом;
- с переменным направлением сдвига (реверсивные регистры сдвига).

Кроме последовательных регистров сдвига имеются параллельные регистры, когда информация подается одновременно на все  $N$  триггеров и считывается одновременно с выходов всех триггеров регистра (рис. 34). Тактовая частота подается одновременно на все триггеры.

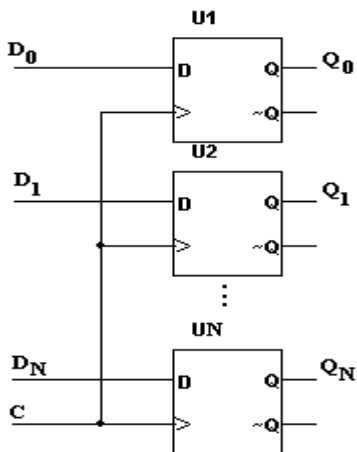


Рис. 34. Параллельный регистр

Параллельные регистры используются для хранения двоичной информации небольшого объема в течение короткого промежутка времени.

### Счетчики

Счетчиком называется устройство, подсчитывающее число входных импульсов. Состоянием счетчика называется число, равное количеству тактовых импульсов, поступивших на его вход. Счетчик состоит из  $n$  последовательно соединенных счетных триггеров, причем выход одного триггера соединен с тактовым входом следующего. Счетчики бывают суммирующими (прямой счет) и вычитающими (обратный счет). В суммирующих счетчиках каждый входной импульс увеличивает число на его выходах на единицу, в вычитающих счетчиках – уменьшает это число на единицу. Для того чтобы построить суммирующий счетчик, необходимо счетный вход очередного триггера подключить к инверсному выходу предыдущего (рис. 35). Для того чтобы изменить направление счета (реализовать вычитающий счетчик), необходимо счетный вход очередного триггера подключить к прямому выходу предыдущего, при этом изменяется последовательность переключения триггеров (рис. 36).

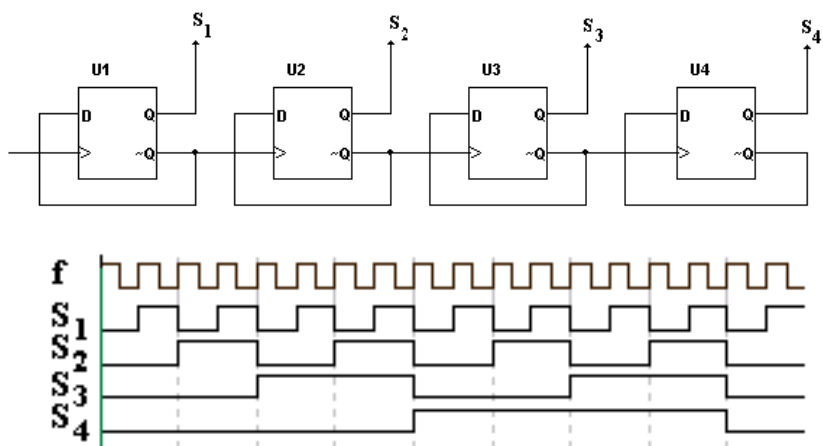


Рис. 35. Суммирующий счетчик и диаграмма его работы

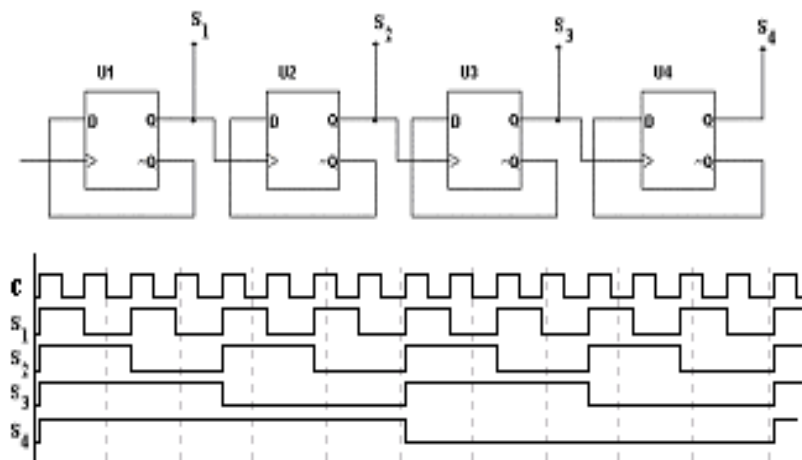


Рис. 36. Вычитающий счетчик и диаграмма его работы

Счетчик характеризуется числом состояний в течение одного периода (цикла). Для двоичных счетчиков полный цикл счета равен  $N = 2^n$  от состояния 0...000 до состояния 1...11. Число состояний называется коэффициентом пересчета  $K_{сч}$ . Коэффициент пересчета равен отношению числа импульсов  $N_c$  на входе к числу импульсов  $N_{Q_{ст}}$  на выходе старшего разряда за период:

$$K_{\text{сч}} = \frac{N_C}{N_{Q\text{ст}}}.$$

Если на вход счетчика подавать периодическую последовательность импульсов с частотой  $f_C$ , то частота  $f_Q$  на выходе старшего разряда счетчика будет меньше в  $K_{\text{сч}}$  раз:  $K_{\text{сч}} = f_C / f_Q$ . Поэтому счетчики также называют делителями частоты, а величину  $K_{\text{сч}}$  – коэффициентом деления. Для увеличения величины  $K_{\text{сч}}$  надо увеличить число триггеров в цепочке. Каждый дополнительный триггер удваивает число состояний счетчика и число  $K_{\text{сч}}$ .

## Дешифраторы

Комбинационной схемой называется логическая схема, реализующая однозначное соответствие между значениями входных и выходных сигналов. Дешифратор – логическая комбинационная схема, которая имеет  $n$  информационных входов и  $2^n$  выходов и преобразует один код в другой. Каждой комбинации логических уровней на входах будет соответствовать активный уровень на одном из  $2^n$  выходов. Как любая логическая схема, дешифратор может быть задан таблицей истинности. В табл. 9 представлена таблица истинности дешифратора  $3 \times 8$ . Эта таблица имеет три столбца, соответствующих входным сигналам:  $X_0, X_1, X_2$ , и восемь столбцов, соответствующих выходным сигналам:  $Y_0, Y_1, Y_2, Y_3, Y_4, Y_5, Y_6, Y_7$ . В первых слева трех столбцах расположены все возможные комбинации входных сигналов, а в последних восьми столбцах – соответствующие им комбинации выходных сигналов.

Т а б л и ц а 9

$X_2$	$X_1$	$X_0$	$Y_7$	$Y_6$	$Y_5$	$Y_4$	$Y_3$	$Y_2$	$Y_1$	$Y_0$
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

Из таблицы видно, что схема имеет восемь выходов, на одном из которых потенциал высокий, на остальных низкий. Номер единственного выхода, на котором формируется высокий потенциал, соответствует двоичному числу, формируемому состояниями входных сигналов. Этот принцип формирования выходного сигнала можно описать следующим образом:  $Y_i = 0$ , если  $i$  равно  $k$ ;  $Y_i = 1$ , если  $i$  не равно  $k$ , здесь  $i$  – номер разряда,  $k = 2^2 X_2 + 2^1 X_1 + 2^0 X_0$ .

Выражения для каждого выхода дешифратора:

$$Y_0 = \neg X_2 / X_1 / X_0,$$

$$Y_4 = X_2 / X_1 / X_0,$$

$$Y_1 = \neg X_2 / X_1 X_0,$$

$$Y_5 = X_2 / X_1 X_0,$$

$$Y_2 = \neg X_2 X_1 / X_0,$$

$$Y_6 = \neg X_2 X_1 X_0,$$

$$Y_3 = X_2 X_1 / X_0,$$

$$Y_7 = X_2 X_1 X_0,$$

здесь  $\neg$  означает инвертирование.

Таким образом, схема дешифратора должна содержать три схемы НЕ и восемь схем ЗИ (рис. 37).

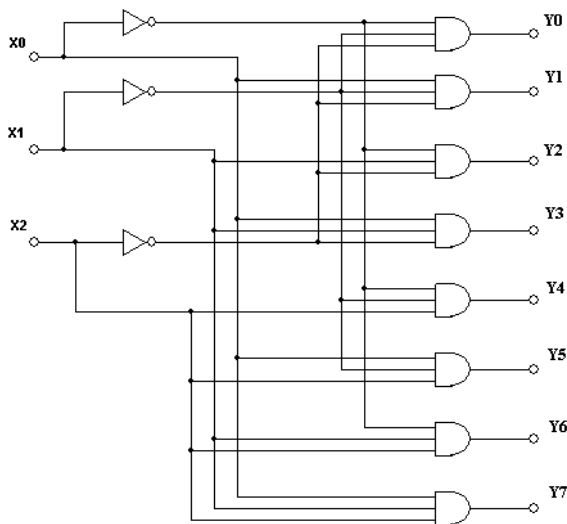


Рис. 37. Схема дешифратора 3 × 8

## Последовательность выполнения работы

### Исследование регистра

1. Нарисовать схему четырехразрядного регистра сдвига с автоматической записью единицы в первый разряд регистра (рис. 38).

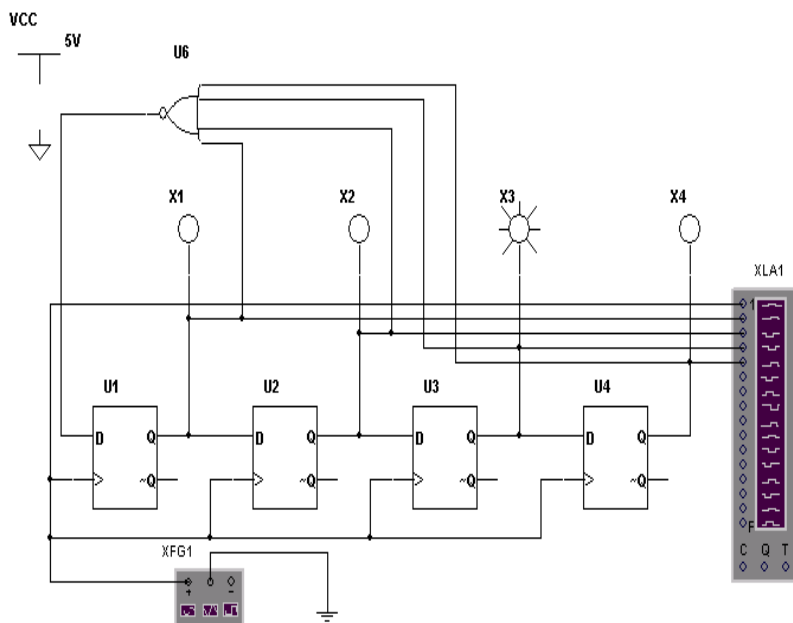



Рис. 38. Четырехразрядный регистр сдвига с автоматической записью единицы

Схема содержит четыре *D*-триггера, четыре светодиода, одну логическую схему 4ИЛИ-НЕ, функциональный генератор и логический анализатор. Логическая схема 4ИЛИ-НЕ служит для автоматической записи единицы в регистр. На выходе этой схемы единица будет только тогда, когда все разряды регистра будут находиться в нулевом состоянии.

2. Открыть окно функционального генератора и установить вид генерируемых сигналов – прямоугольные импульсы, генерируемую частоту 1000 Гц, амплитуду генерируемых импульсов 5 В.

3. Открыть окно логического анализатора, дважды щелкнув по иконке логического анализатора.

4. Запустить процесс моделирования, нажав кнопку  на панели инструментов.


5. Зарисовать диаграммы сигналов с экрана логического анализатора.

6. Зарисовать все состояния светодиодов и сравнить их с временными диаграммами.



## Описание используемых контрольно-измерительных приборов

### Функциональный генератор (Function Generator)

На лицевой панели функционального генератора (рис. 39) расположены три кнопки , с помощью которых выбирается форма генерируемого сигнала: синусоидальная (выбирается по умолчанию), треугольная и прямоугольная.

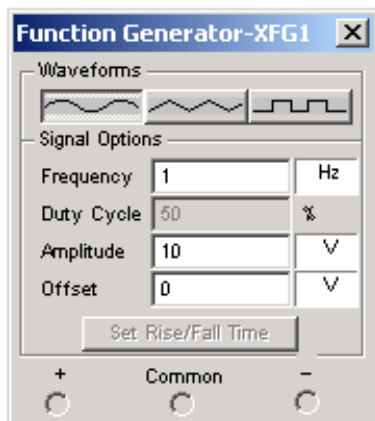

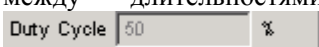
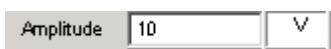




Рис. 39. Лицевая панель функционального генератора

В четырех белых окнах устанавливают параметры выходного сигнала:

- частоты – ,
- коэффициента заполнения в %: для импульсных сигналов – отношение длительности импульса к периоду повторения (величина обратная скважности), для треугольных сигналов – соотношению между длительностями переднего и заднего фронтов ,
- амплитуды выходного сигнала ,
- смещения (постоянной составляющей) .

Внизу лицевой панели расположены выходные зажимы ; при заземлении клеммы COM (общий) на клеммах «+» и «-» генератор выдает парафазный сигнал.

### Логический анализатор (Logic Analyzer)

Логический анализатор предназначен для отображения на экране монитора 16-разрядных кодовых последовательностей одновременно в 16 точках схемы, а также в виде двоичных чисел на входных клеммах-индикаторах (рис. 40).

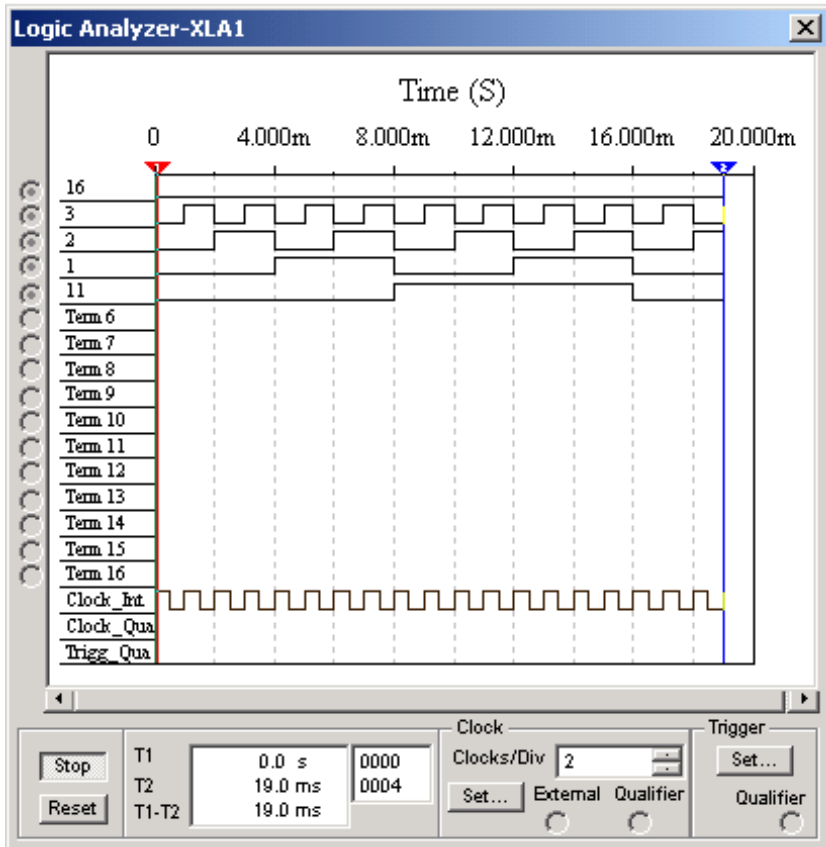


Рис. 40. Лицевая панель логического анализатора

Анализатор снабжен двумя визирными линейками, что позволяет получать точные отсчеты временных интервалов  $T1$ ,  $T2$  и  $T2 - T1$  (окно в поле  $T1$ ,  $T2$ ,  $T2-T1$ ), а также линейкой прокрутки по горизонтали для просмотра осциллограмм на большом временном интервале.

На панели *Clock* в строке *Clocks/Div* задается длительность развертки. В этом блоке имеются клеммы для подключения как обычного (*Extend*), так и избирательного (*Qualifier*) источника запускающих сигналов, параметры которых устанавливаются с помощью меню, вызываемого кнопкой *Set*.

В окне *Trigger* можно задать логические сигналы 1, 0 и X, при наличии которых производится запуск анализатора.

Кнопки *Stop* и *Reset* предназначены для остановки и перезапуска режима моделирования.

### Исследование суммирующего счетчика

1. Нарисовать схему четырехразрядного счетчика, считающего в прямом направлении (рис. 41).

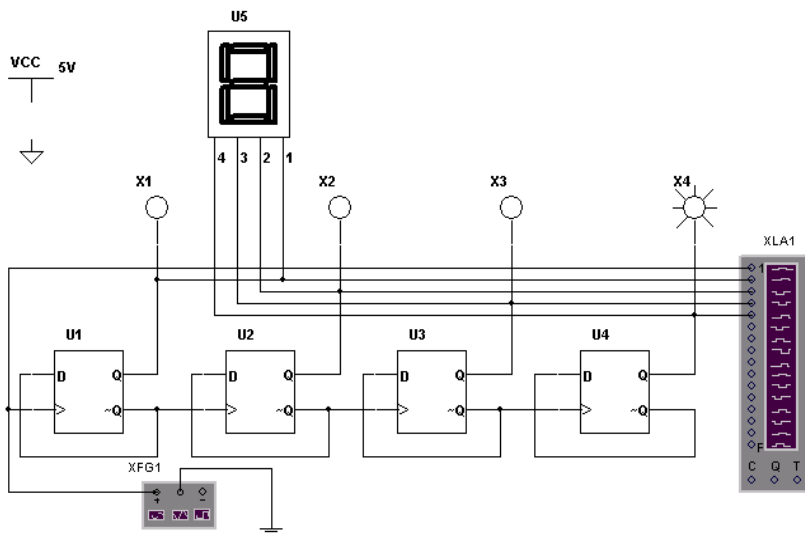



Рис. 41. Четырехразрядный суммирующий счетчик

Поместить на схему четыре *D*-триггера, четыре светодиода, функциональный генератор, логический анализатор и 16-ричный индикатор.

2. Открыть окно логического анализатора, щелкнув по иконке логического анализатора.

3. Запустить процесс моделирования, нажав кнопку  на панели инструментов.

4. Зарисовать диаграммы сигналов с экрана логического анализатора.

5. Зарисовать все состояния светодиодов и сравнить их с временными диаграммами.

6. Наблюдать за показаниями 16-ричного индикатора и сравнить его показания с соответствующими состояниями светодиодов.

### *Исследование вычитающего счетчика*

1. Нарисовать схему четырехразрядного счетчика, считающего в обратном направлении (рис. 42).

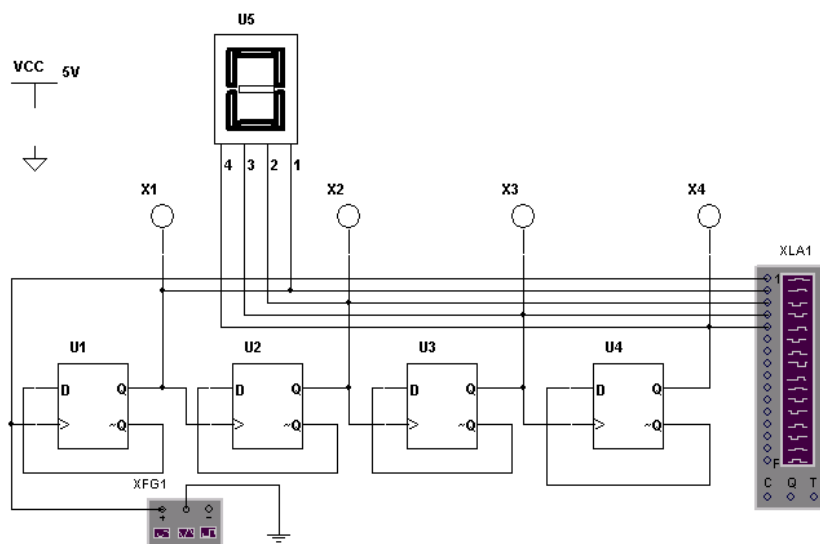



Рис. 42. Четырехразрядный вычитающий счетчик

2. Открыть окно логического анализатора, щелкнув по иконке логического анализатора.

3. Запустить процесс моделирования, нажав кнопку  на панели инструментов.

4. Зарисовать диаграммы сигналов с экрана логического анализатора.

5. Зарисовать все состояния светодиодов и сравнить их с временными диаграммами.

6. Наблюдать за показаниями 16-ричного индикатора и сравнить его показания с соответствующими состояниями светодиодов.

### *Исследование дешифратора*

1. Нарисовать схему дешифратора  $3 \times 8$  (рис. 43). Для этого поместить на схему восемь логических схем ЗИ, восемь светодиодов, пять логических схем НЕ, три переключателя на два направления и три вольтметра.

2. Присвоить переключателям управляющие клавиши.

3. С помощью переключателей подать на вход дешифратора все возможные комбинации сигналов и записать для каждого входного сигнала выходной сигнал (восьмиразрядную комбинацию), т.е. заполнить таблицу истинности.

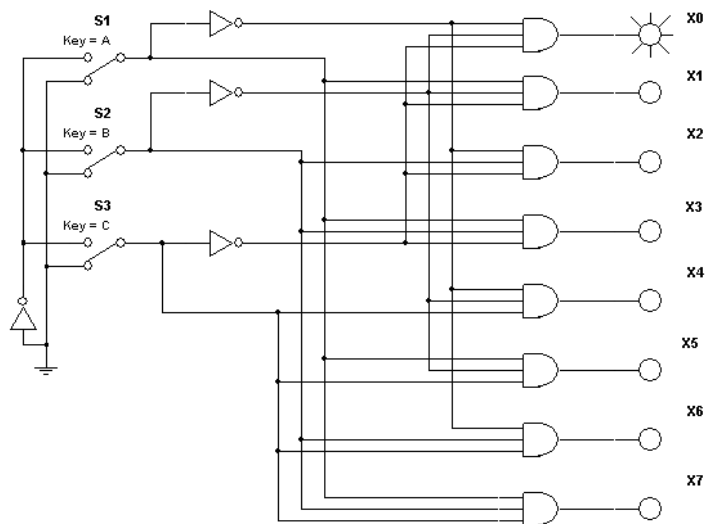


Рис. 43. Дешифратор  $3 \times 8$

### **Содержание отчёта**

1. Название пункта работы.
2. Исследуемая схема.
3. Результат моделирования: для регистра и счетчиков привести временные диаграммы; для дешифратора привести таблицу истинности.

## ИССЛЕДОВАНИЕ ГЕНЕРАТОРА ПСЕВДОСЛУЧАЙНОЙ ПОСЛЕДОВАТЕЛЬНОСТИ

**Цель работы** – исследовать генератор псевдослучайной последовательности.

### Краткие сведения из теории

Регистры сдвига (сдвигающие регистры) являются одними из основных узлов аппаратуры передачи дискретных сообщений. Любой сдвигающий регистр, состоящий из  $n$  триггеров, является запоминающим устройством, способным принимать  $2^n$  различных состояний.

Если в регистр сдвига ввести обратные связи, каждая из которых формируется логической схемой, называемой схемой сложения по модулю два (или Исключающее ИЛИ), то с помощью такого устройства можно построить генератор псевдослучайной последовательности или устройство деления одного многочлена на другой (кодирующие и декодирующие устройства).

Рассмотрим построение генератора псевдослучайной последовательности.

Случайной последовательностью называется последовательность из единиц и нулей, в которой каждый элемент появляется в последовательности независимо от предыдущих символов.

Циклическим сдвигом последовательности называется последовательность, составленная следующим образом. Один или несколько символов в начале последовательности переносятся в её конец, а оставшиеся сдвигаются влево на соответствующее число символов.

10011010000110101 – исходная последовательность,  
10100001101011001 – циклический сдвиг этой последовательности на четыре символа влево.

Всего может быть  $2^n - 1$  циклических сдвигов последовательности, где  $n$  длина последовательности.

Случайные последовательности должны удовлетворять тестам на случайность:

1) уравновешенности, когда число единиц и нулей в последовательности равно;

2) корреляции, когда при почленном сравнении случайной последовательности с любым ее циклическим сдвигом число совпадающих символов равно числу несовпадающих символов;

3) вероятности появления каждого набора из  $m$  элементов для  $m < n$  (где  $n$  – длина последовательности) равна  $2^{-m}$ .

Случайная последовательность не имеет ни начала, ни конца. Получить ее можно, подбрасывая монетку и рассматривая, какая её сторона выпадет. Если выпал “орел”, то записывается единица, если выпала “решка”, то записывается ноль. На практике такие последовательности получить нельзя. Поэтому используют ограниченные (конечные) последовательности, которые обладали бы некоторыми свойствами случайных. Такие последовательности называются псевдослучайными (ПСП).

Получить ПСП можно с помощью регистра сдвига с логическими обратными связями. Простейшая схема такого регистра изображена на рис. 44.

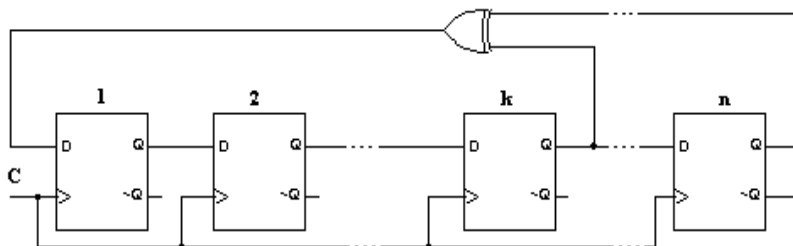


Рис. 44. Генератор псевдослучайной последовательности

На этой схеме на один вход сумматора по модулю два подается сигнал с  $k$ -го разряда регистра, а на второй вход – сигнал с  $n$ -го разряда. Если в первый разряд регистра записать единицу, а в остальные – нули, то под воздействием тактовых импульсов эта единица будет продвигаться по регистру. Когда  $k$ -й разряд примет состояние 1, то на выходе сумматора появится единица. Эта единица поступит на вход первого разряда и запишется в него. С этого момента в регистре сдвига будут продвигаться две единицы. Всякий раз, когда на входы сумматора по модулю два будут подаваться различные символы, в первый разряд регистра сдвига будет записана единица.

Таким образом, под воздействием каждого тактового импульса разряды регистра сдвига принимают определенные состояния, причем последовательность этих состояний зависит от способа включения обратной связи. Общее число таких различных состояний равно  $2^n - 1$ , следовательно, период ПСП равен  $2^n - 1$ . ПСП можно снимать с выхода любого разряда регистра, в частности с последнего. Любая

выходная последовательность, имеющая такой период, называется линейной последовательностью максимальной длины.

Регистр с логической обратной связью описывается характеристическим многочленом вида

$$F(x) = x^n + A_{n-1}x^{n-1} + \dots + A_2x^2 + A_1x + 1.$$

Степень многочлена равна числу разрядов регистра, а коэффициенты  $A_i$  равны 1 или 0. В зависимости от количества и места подключения элементов обратной связи можно получить последовательность максимальной или меньшей длины. Для того чтобы последовательность имела максимальную длину, характеристический многочлен должен быть неприводимым, т.е. не разлагаться на произведение многочленов степеней, меньших  $n$  (или делиться только на 1 и на самого себя).

Таким образом, чтобы построить генератор ПСП, надо знать неприводимый многочлен степени  $n$ . Если для данного  $n$  известен неприводимый многочлен, то процесс создания генератора ПСП состоит из следующих трех шагов:

1. Число триггеров в регистре сдвига равно степени характеристического многочлена.

2. Число элементов обратной связи (сумматоров по модулю два) равно числу ненулевых слагаемых многочлена минус два.

3. Номера разрядов регистра, выходы которых подаются на входы сумматоров по модулю два, определяются из сопряженных многочленов.

Сопряженные многочлены определяются следующим образом: в характеристическом многочлене производят замену переменных  $x = y^{-1}$ ,

$$1/y^n + A_{n-1}/y^{n-1} + \dots + A_2/y^2 + A_1/y + 1,$$

вновь полученный многочлен умножают на  $y^n$ :

$$\begin{aligned} (y^n / y^n)y^n + A_{n-1}(y^n / y^{n-1}) + \dots + A_2(y^n / y^2) + A_1(y^n / y) + y^n = \\ = 1 + A_{n-1}y + \dots + A_2y^{n-2} + A_1y + y^n. \end{aligned}$$

Показатели степени переменной  $y$  сопряженного многочлена, у которых коэффициенты  $A_i$  не равны нулю, соответствуют номерам разрядов регистра, выходы которых подаются на сумматор по модулю два.

Для некоторых  $n$  неприводимые многочлены вычислены и сведены в таблицы. Некоторые из этих многочленов приведены в табл. 10.



Т а б л и ц а 10

Число разрядов $n$	Многочлен	Максимальная длина ПСП	Номера разрядов, выходы которых подаются на сум. по модулю 2
2	$x^2+x+1$	3	1, 2
3	$x^3+x+1$	7	2, 3
	$x^3+x^2+1$	7	1, 3
4	$x^4+x+1$	15	3, 4
	$x^4+x^3+1$	15	1, 4
5	$x^5+x^2+1$	31	3, 5
	$x^5+x^3+1$	31	2, 5
6	$x^6+x+1$	63	5, 6
	$x^6+x^5+1$	63	1, 6
7	$x^7+x+1$	127	6, 7
	$x^7+x^6+1$	127	1, 7
8	$x^8+x^4+x^3+x^2+1$	255	4, 5, 6, 8
	$x^8+x^6+x^5+x^4+1$	255	2, 3, 4, 8

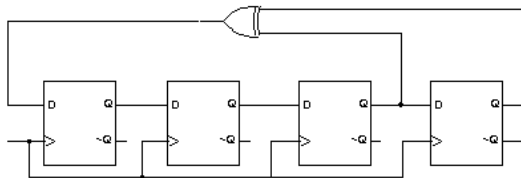
Рассмотрим построение генератора ПСП при  $n = 4$ . В табл. 10 найдем характеристический многочлен степени 4 –  $g(x) = x^4 + x + 1$ .

Произведем замену переменной и найдем сопряженный многочлен:

$$1/y^4 + 1/y + 1,$$

$$y^4(1/y^4 + 1/y + 1) = 1 + y^3 + y^4.$$

Следовательно, регистр сдвига имеет четыре разряда. Число сумматоров по модулю два, включаемых в обратную связь, равно  $3 - 2 = 1$ . Номера разрядов, выходы которых подаются на входы сумматора по модулю два, равны 3 и 4. Схема генератора ПСП приведена на рис. 45.

Рис. 45. Генератор ПСП для  $n = 4$

Если записать в первый разряд 1, то под воздействием тактовых импульсов содержимое регистра будет меняться следующим образом:

1. 1000
2. 0100
3. 0010
4. 1001
5. 1100
6. 0110
7. 1011
8. 0101
9. 1010
10. 1101
11. 1110
12. 1111
13. 0111
14. 0011
15. 0001
16. 1000

Начиная с 16-го такта последовательность состояний регистра повторяется. ПСП можно снимать с выхода любого разряда регистра, а также с выхода схемы сложения по модулю два. Псевдослучайная последовательность, генерируемая данным устройством, имеет вид

000100110101111.

В данном случае эта последовательность получена с выхода четвертого разряда регистра сдвига (четвертый столбец таблицы).

### **Последовательность выполнения работы**

#### ***Исследование генератора ПСП для $n = 4$***

1. Нарисовать схему генератора ПСП для  $n = 4$ , приведенную на рис. 46. Поместить на схему функциональный генератор (*Function Generator*) и осциллограф (*Oscilloscope*).

На этой схеме логический элемент 4И-НЕ введен для автоматической записи 1 в первый разряд регистра, если в момент включения питания все триггеры регистра сдвига установятся в состояние 0.

2. Открыть окно функционального генератора и установить частоту (1 КГц) и вид генерируемого сигнала (прямоугольные импульсы).

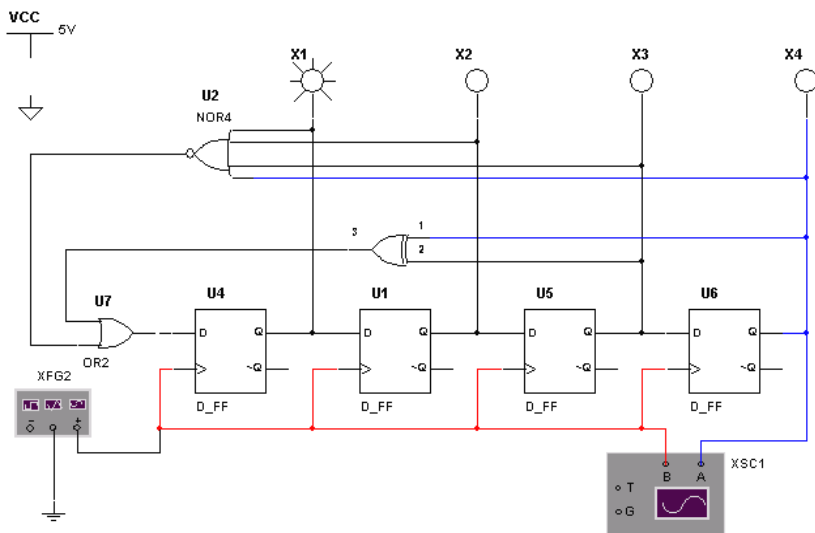



Рис. 46. Четырехразрядный генератор ПСП

3. Запустить процесс моделирования, нажав кнопку  на панели инструментов, и в появившемся меню выбрать команду Run.

4. Щелкнуть по иконке осциллографа, чтобы открыть окно осциллографа, и зарисовать осциллограмму.

### ***Описание осциллографа (Oscilloscope)***

Принцип работы осциллографа основывается на применении электронно-лучевой трубки (ЭЛТ), содержащей катод, излучающий поток (луч) электронов за счет термоэмиссии, модулятор для управления интенсивностью этого потока, систему его фокусировки, отклоняющую систему в виде двух пар пластин и экран из кристалликов люминофора, которые под действием падающего на них потока электронов высвечивают траекторию движения электронного луча в пространстве, определяемую потенциалами на пластинах отклоняющей системы. Горизонтальные пластины этой системы называются *Y*-каналом, вертикальные – *X*-каналом.

На пластины *Y*-канала подается исследуемый сигнал после его усиления и масштабирования вертикальным *Y*-усилителем. Для одновременного наблюдения *n* сигналов используют *n* усилителей, выходные сигналы которых с помощью аналоговых коммутаторов поочередно подаются на вход усилителя мощности *Y*-канала.

На пластины X-канала подается пилообразный сигнал развертки, длительность (период) которого с помощью системы синхронизации поддерживается равным или кратным периоду исследуемого сигнала в одном из Y-каналов, что позволяет получить устойчивое, а не хаотично «бегающее» изображение контролируемого сигнала.

Осциллограф имеет два канала (*Channel*): *A* и *B* с отдельной установкой чувствительности в диапазоне от 10 мкВ / дел ( $V/Div$ ) до 5 кВ / дел ( $kV/Div$ ) и регулировкой смещения по вертикали (*Y POS*) (рис. 47).

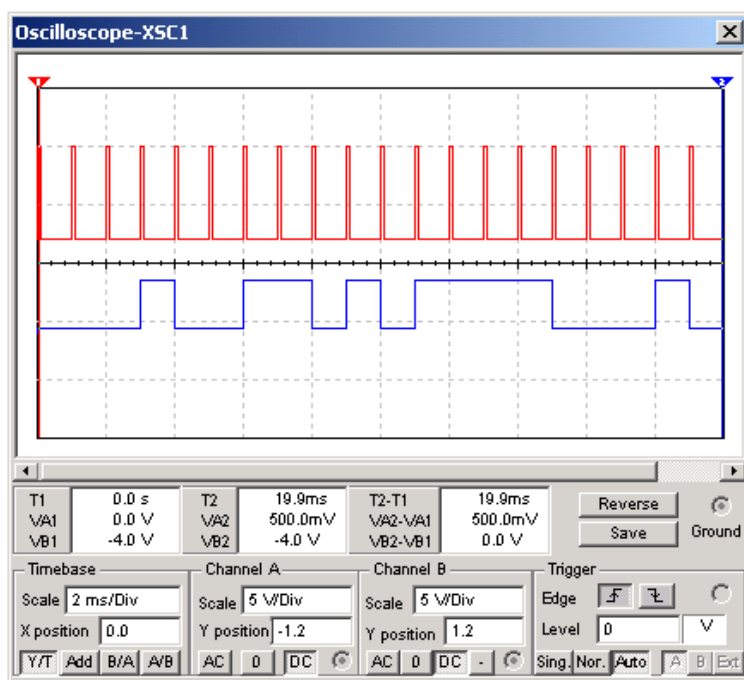




Рис. 47. Лицевая панель осциллографа

На панелях *Channel A* и *Channel B* с помощью кнопок осуществляется выбор режима по входу. Режим *AC* предназначен для наблюдения только сигналов переменного тока. В режиме *0* входной зажим замыкается на землю. В режиме *DC* (включен по умолчанию) можно наблюдать сигналы как постоянного, так и переменного тока. С правой стороны от кнопки *DC* расположен входной зажим канала.

На этих панелях в окне *Scale* устанавливается величина развертки, а в окне *Y position* – величина смещения относительно оси *X*.

На панели *Timebase* с помощью кнопок *Y/T*, *Add*, *B/A*, *A/B* выбирается режим развертки. В режиме *Y/T* (обычный режим, включен по умолчанию) устанавливается следующий режим развертки: по вертикали – напряжение сигнала, по горизонтали – время; в режиме *Add* сигналы обоих каналов накладываются друг на друга; в режиме *B/A*: по вертикали – сигнал канала *B*, по горизонтали – сигнал канала *A*; в режиме *A/B*: по вертикали – сигнал канала *A*, по горизонтали – сигнал канала *B*.

В режиме *Y/T* длительность развертки (*Timebase*) может быть задана в диапазоне от 0,1 нс / дел (*ns/div*) до 1 с / дел (*s/div*) с возможностью установки смещения в тех же единицах по горизонтали, т.е. по оси *X* (*X POS*).


В режиме *Y/T* можно установить ждущий режим в поле *Trigger*. В строке *Edge* устанавливается режим запуска по переднему (кнопка ) или по заднему (кнопка ) фронту запускающего сигнала. В окне *Lever* (уровень) устанавливается уровень запускающего сигнала.

В этом поле с помощью кнопок *Auto*, *A*, *B* или *Ext* ждущий режим запускается от каналов *A* или *B*, от канала *A*, от канала *B* или от внешнего источника, подключаемого к находящемуся на этой панели зажиму.

С помощью кнопки *Reverse* можно инвертировать изображение, с помощью кнопки *Save* – записать данные в файл.


С помощью визирных линеек (синего и красного цвета) можно просканировать изображение. Результаты измерения напряжения, временных интервалов и их приращений (между визирными линиями) отображаются в индикаторных окошках, расположенных под экраном.

### **Исследование генератора ПСП для $n = 5$ и $g(x) = x^5 + x^3 + 1$**

1. Нарисовать схему генератора ПСП для  $n = 5$  и  $g(x) = x^5 + x^3 + 1$ .
2. Запустить процесс моделирования, нажав кнопку  на панели инструментов.
3. Щелкнуть по иконке осциллографа, чтобы открыть окно осциллографа, и зарисовать осциллограмму.

### **Исследование генератора ПСП для $n=5$ и $g(x)=x^5+x^4+1$**

1. Нарисовать схему генератора ПСП для  $n = 5$  и  $g(x) = x^5 + x^4 + 1$ . Для этого предварительно найти сопряженный многочлен.

2. Запустить процесс моделирования, нажав кнопку  на панели инструментов.

3. Щелкнуть по иконке осциллографа, чтобы открыть окно осциллографа, и зарисовать осциллограмму.

### Содержание отчёта

1. Название пункта работы.
2. Исследуемая схема.
3. Результат моделирования: все различные состояния разрядов регистра и полученную ПСП.

## ЛАБОРАТОРНАЯ РАБОТА № 4

### ИССЛЕДОВАНИЕ АРИФМЕТИЧЕСКОГО СУММАТОРА

**Цель работы** – исследовать арифметический сумматор, полусумматор и полный сумматор.

#### Краткие сведения из теории

Арифметический сумматор предназначен для сложения двух двоичных  $n$ -разрядных чисел.

Арифметические сумматоры являются составной частью арифметико-логических устройств (АЛУ) микропроцессоров (МП). Арифметический сумматор состоит из двух устройств: полусумматора и  $n - 1$  полных сумматоров. Полный сумматор имеет три входа:  $A$ ,  $B$  – входы суммируемых операндов,  $C_i$  – вход переноса из предыдущего разряда сумматора, и два выхода:  $S$  – выход суммы и  $C_o$  – выход переноса. Полусумматор отличается от полного сумматора тем, что у него нет входа переноса из предыдущего разряда. Полусумматор используется в качестве первого разряда арифметического сумматора, а в качестве остальных разрядов арифметического сумматора используются полные сумматоры (рис. 48).

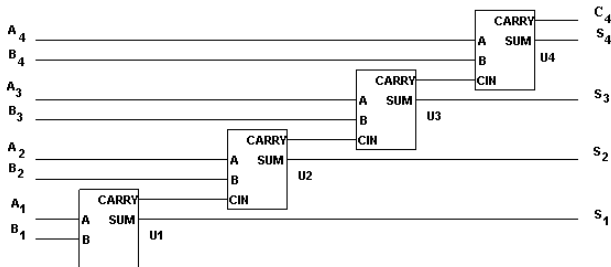


Рис. 48. Четырехразрядный арифметический сумматор

Полусумматор является одной из простейших комбинационных логических схем. Таблица истинности полусумматора имеет вид табл. 11.

Т а б л и ц а 11

Входы		Выходы	
<i>A</i>	<i>B</i>	<i>S</i>	<i>C</i>
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Рассматривая эту таблицу, можно заметить, что выход *S* полусумматора выполняет функции элемента ИСКЛЮЧАЮЩЕЕ ИЛИ, а выход переноса *C* полусумматора – функции элемента И. Таким образом, логические выражения для функций *S* и *C* будут такими:

$$S = AB' + A'B,$$

$$C = AB.$$

Схема полусумматора представлена на рис. 49.

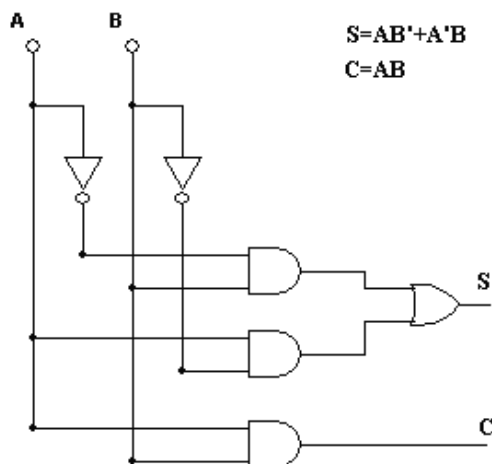


Рис. 49. Структура полусумматора

Таблица истинности полного сумматора имеет вид табл.12.

Т а б л и ц а 12

Входы			Выходы	
$A$	$B$	$C_{i-1}$	$S$	$C_i$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Из этой таблицы можно получить логические выражения для  $S$  (суммы) и  $C$  (переноса в следующий разряд). Логическое выражение для  $S$  будет иметь четыре слагаемых, соответствующих строкам таблицы, в которых выход  $S$  равен 1 (строки 2, 3, 5, 8):

$$S = A'B'C_{i-1} + A'BC_{i-1}' + AB'C_{i-1}' + ABC_{i-1}.$$

Логическое выражение для  $C$  также будет иметь четыре слагаемых (строки 6, 8, 9, 10):

$$C_i = A'BC_{i-1} + A'BC_{i-1}' + ABC_{i-1}' + ABC_{i-1}.$$

С помощью законов булевой алгебры (см. лаб. раб. №1) это выражение можно упростить. Тогда оно будет иметь вид

$$C_i = AC_{i-1} + BC_{i-1} + AB.$$

Схема полного сумматора изображена на рис. 50.

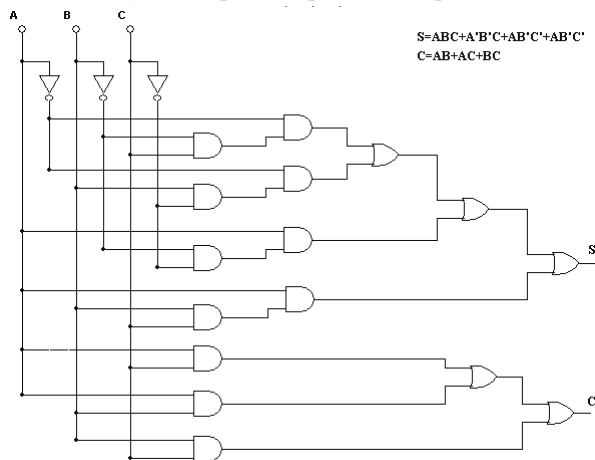




Рис. 50. Структура полного сумматора



- 1) построить таблицу истинности этого устройства,
- 2) по таблице истинности составить логическое выражение для выходного сигнала этого устройства,
- 3) по логическому выражению нарисовать принципиальную схему устройства.

*Truth Table to Boolean Expression* (булево выражение по таблице истинности) , *Boolean Expression to Circuit* (создание схемы по булеву выражению) , получить:

- таблицу истинности полусумматора,
- логические выражения для выходов  $S$  и  $C$ ,
- схемную реализацию полученных логических выражений.

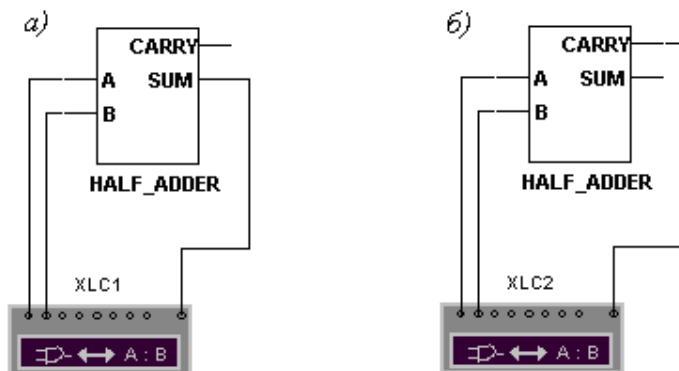


Рис. 52. Схема исследования полусумматора:  $a$  – выход  $S$ ,  $b$  – выход  $C$

5. Собрать схему, изображенную на рис. 53, и с помощью логического преобразователя получить таблицу истинности полного сумматора, логические выражения для выходов  $S$  и  $C$ , схемную реализацию полученных логических выражений (см. п. 4).

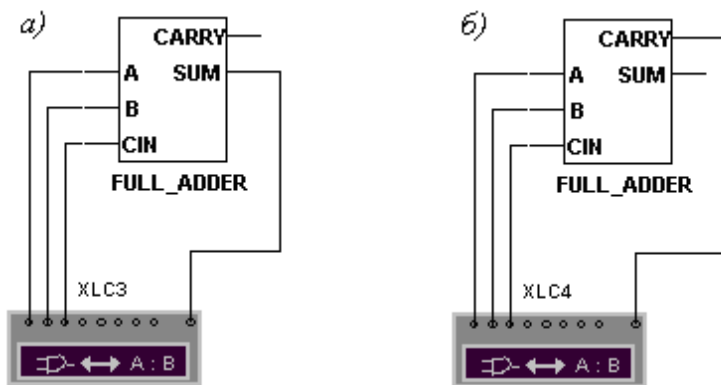


Рис. 53. Схема исследования полного сумматора:  $a$  – выход  $S$ ,  $b$  – выход  $C$

## Описание используемых контрольно-измерительных приборов

### Генератор слова (*Word Generator*)



Генератор слова (или кодовый генератор) предназначен для генерации 32-разрядных двоичных слов. На рис. 54 изображена лицевая панель генератора слова. В окне, расположенном справа лицевой панели генератора слова, отображаются 8-разрядные 16-ричные числа от 00000000 до FFFFFFFF (от 0 до 4294967265). Каждая горизонтальная строка представляет 32-разрядное двоичное число. Эти числа поступают в параллельном виде на выходные клеммы генератора, расположенные внизу лицевой панели.

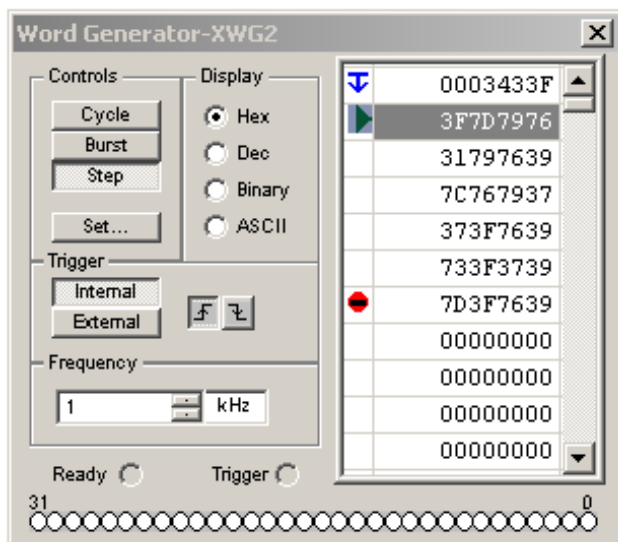


Рис. 54. Лицевая панель генератора слова

Чтобы изменить значение любого бита кодового слова, надо выбрать число (т.е. щелкнуть по нему, при этом оно меняет цвет), которое необходимо изменить, и ввести новое значение.

Двоичные числа могут набираться пользователем в 16-ричном (*Hex*), десятичном (*Dec*), двоичном кодах (*Binary*) или в коде *ASCII*. Код, в котором набираются двоичные числа, выбирается с помощью переключателей *Hex*, *Dec*, *Binary* или *ASCII*, расположенных на панели *Display* (см. рис. 54). Для выбора кода надо поставить черную точку в один из переключателей.

На панели *Controls* устанавливается режим выдачи кодовых слов. Чтобы выдать 32-разрядное слово на выход прибора, надо щелкнуть по одной из кнопок: *Step*, *Burst* или *Cycle*.

Если необходимо выдать только одно слово, щелкнуть по кнопке *Step*.

Если необходимо выдать все кодовые слова множества, щелкнуть по кнопке *Burst*.

Если щелкнуть по кнопке *Cycle*, то будут выдаваться все кодовые слова множества непрерывно циклически. Остановить выдачу слов можно, повторно щелкнув по кнопке *Cycle*.

Для того чтобы ограничить число выдаваемых на выход прибора слов, надо щелкнуть правой клавишей мыши в крайне левом столбце напротив слова, которым ограничивается список выдаваемых слов. Откроется всплывающее меню (рис. 55). В этом меню выбрать строку *Set Break-Point*, напротив выбранного слова появится красная точка.

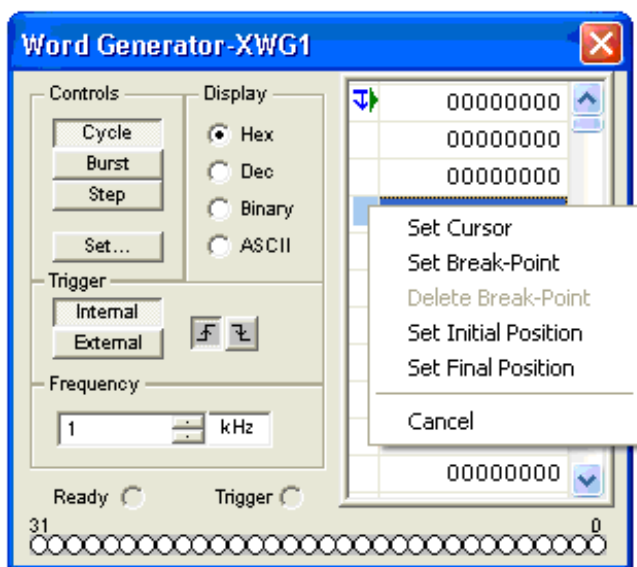


Рис. 55. Всплывающее меню генератора слова

Для того чтобы удалить точку останова, во всплывающем меню выбрать строку *Delete Break-Point*.

Перед началом процесса моделирования установить курсор на первой строке таблицы. Для этого выбрать первую строку таблицы и во всплывающем меню выбрать строку *Set Cursor*. В крайнем левом столбце появится стрелка, направленная острием вниз (см. рис. 55).

На панели *Triggering* расположены четыре кнопки, с помощью которых можно установить источник запускающего сигнала (внутренний (*Internal*) или внешний (*External*)) и фазу запускающего сигнала (по переднему или заднему фронту).

На панели *Frequency* устанавливается тактовая частота генератора слова в Гц, кГц или МГц. Кодовые слова поступают на выход генератора с каждым тактом генератора. Рядом расположена клемма *Ready*, на которую выдается сигнал готовности выдавать данные.



### Логический преобразователь (*Logic Converter*)

На лицевой панели преобразователя (рис. 56) расположены клеммы-индикаторы входов *A*, *B*, *C*, ..., *H* и клемма выхода *Out*, окно для отображения таблицы истинности исследуемой схемы, строка для отображения ее булева выражения и панель *Conversions*.

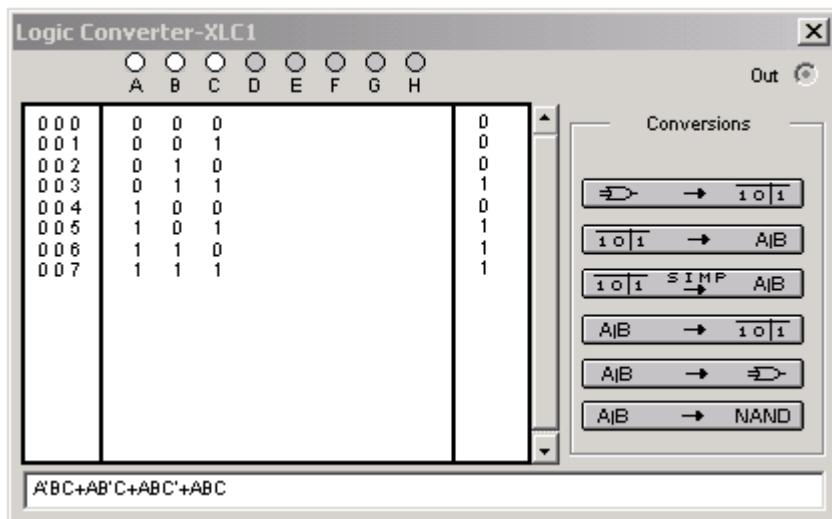


Рис. 56. Лицевая панель логического преобразователя

На панели *Conversions* расположены пять кнопок:

- кнопка – для получения таблицы истинности исследуемого устройства,
- кнопка – для получения булева выражения, реализуемого исследуемым устройством,

- кнопка – для получения минимизированного булева выражения,
- кнопка – для получения таблицы истинности по булевому выражению,
- кнопка – для создания схемы устройства по логическому выражению на логических элементах без ограничения их типа,
- кнопка – для создания схемы устройства только на логических элементах И-НЕ.

### Содержание отчёта

1. Название пункта работы.
2. Исследуемая схема.
3. Результат моделирования.

## ЛАБОРАТОРНАЯ РАБОТА №5

### ИССЛЕДОВАНИЕ РАБОТЫ МУЛЬТИПЛЕКСОРА И ДЕМУЛЬТИПЛЕКСОРА

**Цель работы** – исследовать мультиплексор и демultipлексор.

#### Краткие сведения из теории

##### Мультиплексоры

Мультиплексором называется функциональный узел, предназначенный для поочередной коммутации (переключения) информации от одного из  $n$  входов на общий выход. Номер конкретной входной линии, подключаемой к выходу, определяется адресным кодом  $A_0, A_1, \dots, A_{n-1}$ .

Обобщенная схема мультиплексора (MUX) состоит из коммутатора и управляющей входной логической схемы (рис. 57).

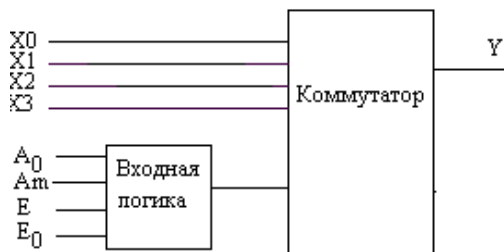


Рис. 57. Обобщенная схема мультиплексора

Входные логические сигналы  $X_i$  поступают на входы коммутатора и через него передаются на выход  $Y$ . Управление коммутатором осуществляется управляющей логической схемой. На вход логической схемы подаются адресные сигналы  $A_k$ .

Мультиплексор может иметь дополнительный управляющий вход  $E$ , который может выполнять стробирование выхода  $Y$ . Кроме того, некоторые мультиплексоры могут иметь выход с тремя состояниями: два состояния 0 и 1 и третье состояние – отключенный выход (выходное сопротивление равно бесконечности). Перевод мультиплексора в третье состояние производится сигналом  $OE$ .

Мультиплексоры могут быть односторонними (от входа к выходу) или двусторонними (от входа к выходу и от выхода к входу). Мультиплексоры со стробирующим входом  $E$  выполняют функцию передачи сигнала от входа к выходу только при поступлении сигнала строба  $E$ . Мультиплексоры с тремя состояниями можно каскадировать.

В зависимости от числа информационных входов  $n$  и адресных входов  $m$ , мультиплексоры делятся на полные и неполные. Если  $n = 2^m$ , то мультиплексоры называются полными. Если  $n < 2^m$ , то мультиплексор называется неполным.

Наибольшее распространение получили мультиплексоры  $(2 \rightarrow 1)$  с  $n = 2$  и  $m = 1$ ,  $(4 \rightarrow 1)$  с  $n = 4$  и  $m = 1$ ,  $(8 \rightarrow 1)$  с  $n = 8$  и  $m = 1$ ,  $(16 \rightarrow 1)$  с  $n = 16$  и  $m = 1$ .

В качестве примера рассмотрим работу мультиплексора  $(4 \rightarrow 1)$ . Логика его работы представлена в табл. 13, где  $A_0, A_1$  – входы адреса,  $F_0, F_1, F_2, F_3$  – выходы внутреннего дешифратора,  $X_0, X_1, X_2, X_3$  – входная информация,  $Q$  – общий информационный выход.

Т а б л и ц а 13

$A_1$	$A_0$	$F_3$	$F_2$	$F_1$	$F_0$	$Q$
0	0	0	0	0	1	$F_0X_0$
0	1	0	0	1	0	$F_1X_1$
1	0	0	1	0	0	$F_2X_2$
1	1	1	0	0	0	$F_3X_3$

На основе табл. 13 выражение для выходной функции  $Q$  можно представить с использованием выходов  $F_0, F_1, F_2, F_3$  внутреннего дешифратора в виде

$$Q = F_0X_0 + F_1X_1 + F_2X_2 + F_3X_3, \text{ или, с учетом адресных входов,}$$

$$Q = \bar{A}_1\bar{A}_0X_0 + \bar{A}_1A_0X_1 + A_1\bar{A}_0X_2 + A_1A_0X_3.$$

Соответствующая схема мультиплексора изображена на рис. 58.

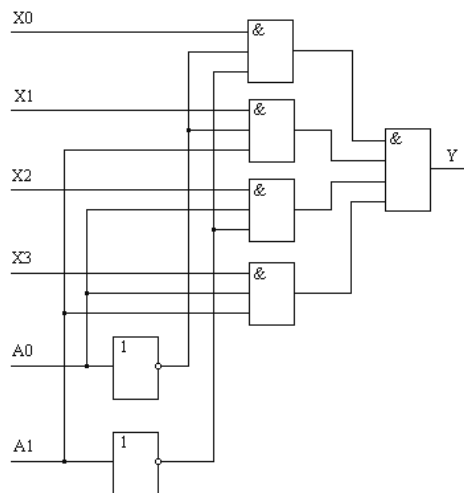


Рис. 58. Схема мультиплексора ( $4 \rightarrow 1$ )

### Демультимплексор

Демультимплексором называется функциональный узел, который обеспечивает передачу цифровой информации, поступающей по одной входной линии, на несколько выходных линий. Выбор выходной линии осуществляется при помощи сигналов, поступающих на адресные входы. То есть демультимплексор осуществляет преобразование, обратное действию мультиплексора.

Обобщенная схема демультимплексора состоит из коммутатора и входной логики (рис. 59).



Рис. 59. Схема демультимплексора ( $1 \rightarrow 4$ )

В качестве примера рассмотрим работу демультимплексора ( $1 \rightarrow 4$ ). Логика его работы представлена в табл. 14, где  $A_0, A_1$  – входы адреса,  $F_0, F_1, F_2, F_3$  – выходы внутреннего дешифратора адреса,  $X$  – общий информационный вход,  $Y_0, Y_1, Y_2, Y_3$  – выходная информация.



Т а б л и ц а 14

$A_1$	$A_0$	$F_3$	$F_2$	$F_1$	$F_0$	$Y_3$	$Y_2$	$Y_1$	$Y_0$
0	0	0	0	0	1	$F_3X_3$			
0	1	0	0	1	0		$F_2X_2$		
1	0	0	1	0	0			$F_1X_1$	
1	1	1	0	0	0				$F_0X_0$

На основе табл. 14 выражения для выходных функций  $Y_i$  можно представить с использованием адресных входов  $A_1, A_0$ :

$$Y_0 = \bar{A}_1 \bar{A}_0 X ,$$

$$Y_1 = A_1 \bar{A}_0 X ,$$

$$Y_2 = \bar{A}_1 A_0 X ,$$

$$Y_3 = A_1 A_0 X .$$

Соответствующая схема демультиплексора изображена на рис. 60.

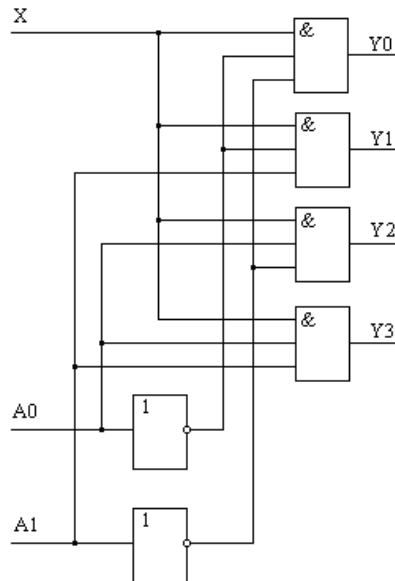


Рис. 60. Схема демультиплексора (1→4)

В табл. 15 приведен алгоритм работы схемы управления.

Т а б л и ц а 15

<i>A</i>	<i>B</i>	<i>Q</i>
0	0	$Y_0$
0	1	$Y_1$
1	0	$Y_2$
1	1	$Y_3$

### Последовательность выполнения работы

#### Исследование мультиплексора

1. Нарисовать схему исследования мультиплексора  $4 \rightarrow 1$  (рис. 61).  
Схема содержит коммутатор (*U1, U2, U3, U4, U5*), схему управления (три элемента НЕ и два переключателя), четыре источника и входной информации (выходы разрядов генератора ПСП). Генератор ПСП состоит из элементов *U6, U7, U8, U9*.

Кроме того, на схеме расположены два прибора: функциональный генератор и логический анализатор.

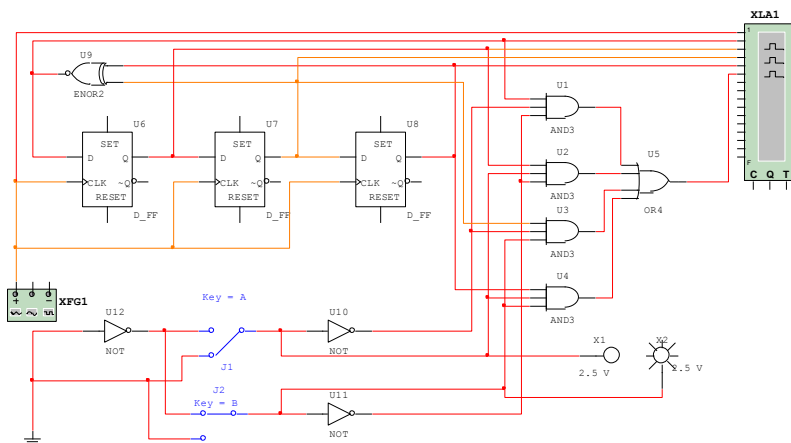




Рис. 61. Схема исследования мультиплексора ( $4 \rightarrow 1$ )

2. Открыть окно функционального генератора и установить вид генерируемых сигналов (прямоугольные импульсы), генерируемую частоту 1000 Гц, амплитуду генерируемых импульсов 5 В.

3. Открыть окно логического анализатора, дважды щелкнув по иконке логического анализатора.

4. С помощью схемы управления установить номер входа, информация которого должна появиться на выходе.

5. Запустить процесс моделирования. Сделать это можно тремя способами:

- нажав кнопку  на панели инструментов,
- в меню Simulate выбрать команду *Run*,
- перевести переключатель  в положение 1.

6. Выключить процесс моделирования, затем вернуть движок полосы прокрутки в крайне левое положение и зарисовать диаграмму сигналов с экрана логического анализатора (рис. 62).

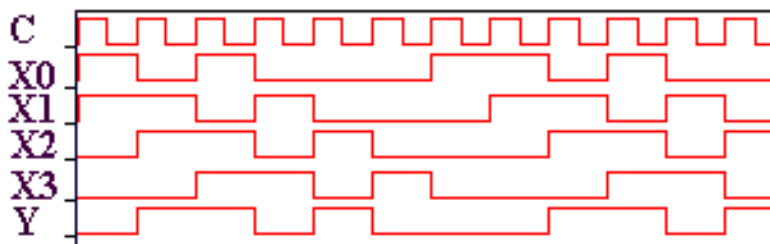


Рис. 62. Диаграмма работы мультиплексора

На устройстве управления мультиплексором набрано двоичное число 1, информация, поступившая на второй вход мультиплексора, должна появиться на выходе мультиплексора.

### ***Исследование демultipлексора***

1. Нарисовать схему исследования демultipлексора (1→4) (рис. 63).

Схема содержит коммутатор ( $U1$ ,  $U2$ ,  $U3$ ,  $U4$ ), схему управления (три элемента НЕ и два переключателя), четыре источника и входной информации (выходы разрядов генератора ПСП). Генератор ПСП состоит из элементов  $U6$ ,  $U7$ ,  $U8$ ,  $U9$ .

Кроме того на схеме расположены два прибора: функциональный генератор и логический анализатор.

2. Открыть окно функционального генератора и установить вид генерируемых сигналов (прямоугольные импульсы), генерируемую частоту 1000 Гц, амплитуду генерируемых импульсов 5 В.

3. Открыть окно логического анализатора, дважды щелкнув по его иконке.

4. С помощью схемы управления установить номер выхода, на котором должна появиться информация.

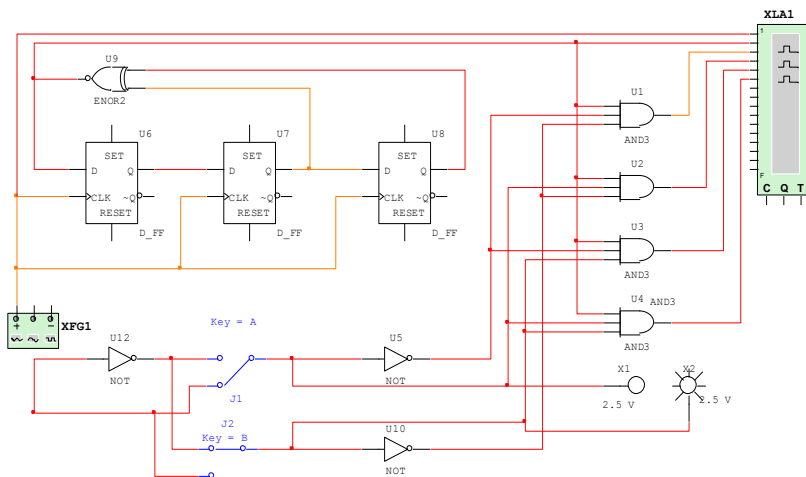




Рис. 63. Схема исследования демультиплексора (1→4)

5. Запустить процесс моделирования. Сделать это можно тремя способами:

- нажав кнопку  на панели инструментов,
- в меню Simulate выбрать команду *Run*,
- перевести переключатель  в положение 1.

6. Выключить процесс моделирования, затем вернуть движок полосы прокрутки в крайнее левое положение и зарисовать диаграмму сигналов с экрана логического анализатора (рис. 64).

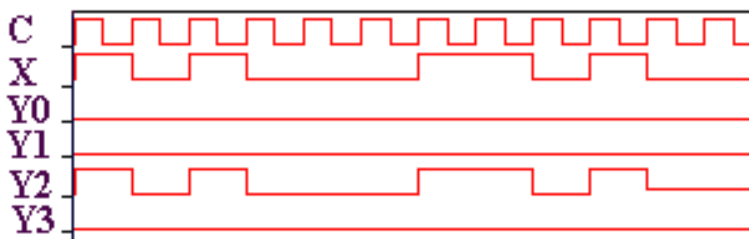


Рис. 64. Диаграмма работы демультиплексора

На устройстве управления демультиплексором набрано число 2, и информация появилась на третьем выходе демультиплексора.

## Содержание отчёта

1. Название пункта работы.
2. Исследуемая схема.
3. Результат моделирования.

### ЛАБОРАТОРНАЯ РАБОТА № 6

## ВЫВОД СТАТИЧЕСКОГО СЛОВА

**Цель работы** – вывести на дисплей, состоящий из четырех семисегментных индикаторов, слово из четырех букв.

### Краткие сведения из теории

Дисплей состоит из четырех индикаторов, состоящих из восьми светодиодных сегментов, обозначенных латинскими буквами *a, b, c, d, e, f, g* и *h* (рис. 65). Каждому индикатору в памяти ЭВМ отводится своя ячейка памяти, из которой информация выводится на индикаторы.

Каждый бит ячейки памяти управляет состоянием соответствующего ему сегмента индикатора (табл. 16). Для того чтобы заставить светиться какой-либо сегмент индикатора, надо в соответствующий ему разряд ячейки памяти записать единицу, чтобы погасить этот сегмент, надо в этот разряд записать ноль.

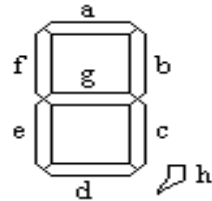


Рис. 65. Семисегментный индикатор

Связь между сегментами индикатора и битами соответствующей ему ячейки памяти приведена в табл. 16.

Т а б л и ц а 16

Бит ячейки памяти	7	6	5	4	3	2 <sub>0</sub>	1	0
Сегмент индикатора	h	g	f	e	d	c	b	a

Таким образом, на индикатор можно вывести некоторые буквы русского и латинского алфавитов и десятичные цифры от 0 до 9, если зажечь соответствующие сегменты индикатора. Но прежде надо определить коды этих букв и цифр.

Рассмотрим определение кодов букв на примере слова СЕРЕБРО.

Для того чтобы вывести на индикатор букву С, надо зажечь сегменты *a, d, e* и *f*, т.е. поставить 1 в разряды 0, 3, 4 и 5, а в остальные разряды – 0.

Для того чтобы вывести на индикатор букву Е, надо зажечь сегменты а, d, е, f и g, т.е. поставить 1 в разряды 0, 3, 4, 5, 6, а в остальные разряды – 0.

Для того чтобы вывести на индикатор букву Р, надо зажечь сегменты а, b, е, f и g, т.е. поставить 1 в разряды 0, 1, 4, 5 и 6, а в остальные разряды – 0.

Для того чтобы вывести на индикатор букву Б, надо зажечь сегменты а, с, d, е, f и g, т.е. поставить 1 в разряды 0, 2, 3, 4, 5 и 6, а в остальные разряды – 0.

Для того чтобы вывести на индикатор букву О, надо зажечь сегменты а, b, с, d, е и f, т.е. поставить 1 в разряды 0, 1, 2, 3, 4 и 5, а в остальные разряды – 0.

Изображение букв и соответствующие им коды приведены на рис. 66 и в табл. 17.

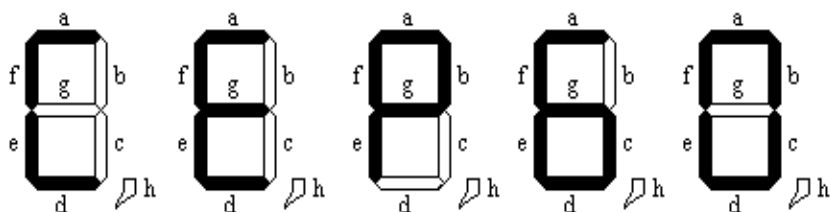


Рис. 66. Изображение букв

Т а б л и ц а 17

Буква	7	6	5	4	3	2	1	0	Коды букв
	h	g	f	e	d	c	b	a	
С	0	0	1	1	1	0	0	1	39
Е	0	1	1	1	1	0	0	1	79
Р	0	1	1	1	0	0	1	1	73
Б	0	1	1	1	1	1	0	1	7d
О	0	0	1	1	1	1	1	1	3f

Светодиоды в семисегментных индикаторах обычно скомпонованы в виде матриц с объединенными катодами или анодами. Управление такими индикаторами осуществляется высокими или низкими уровнями логических сигналов соответственно. Структуры элементов индикации названного типа приведены на рис. 67 и 68.

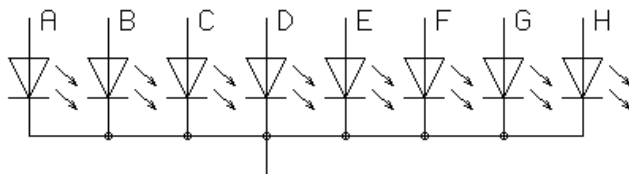


Рис. 67. Индикатор с общими анодами

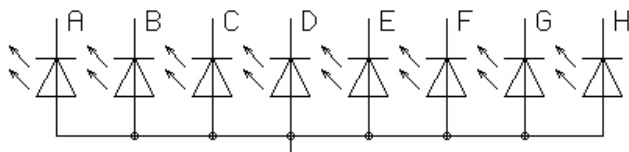


Рис. 68. Индикатор с общими катодами

Схемы включения индикаторов с общими анодами и катодами показаны на рис. 69 и 70.

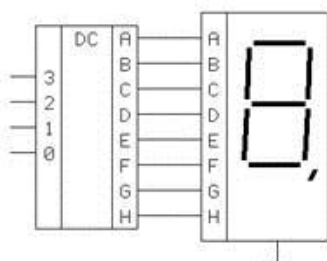


Рис. 69

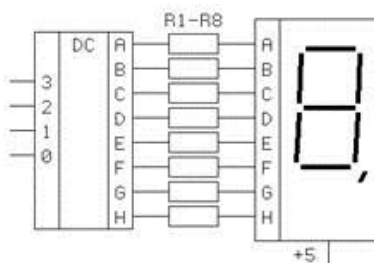



Рис. 70

### ***Последовательность выполнения работы***

1. Собрать схему, изображенную на рис. 71. Поместить на схему четыре семисегментных индикатора с общими катодами и генератор слов.

2. Для того чтобы поместить индикаторы на схему, надо на панели инструментов (расположенной вертикально слева в окне схемы) выбрать клавишу «индикаторы» . Появится диалоговое окно *Select a Component* (рис. 72). В этом диалоговом окне выбрать строку *HEX\_DISPLAY* (окно *Family*).

3. Во втором окне *Component* выбрать строку *SEVEN\_SEG\_COM\_K*. В окне *Symbol(ANSI)* появится изображение компонента. Нажать клавишу *OK*. Компонент прикрепится к курсору. Поместить компонент на схему и щелкнуть левой клавишей мыши. Компонент будет помещен на схему.

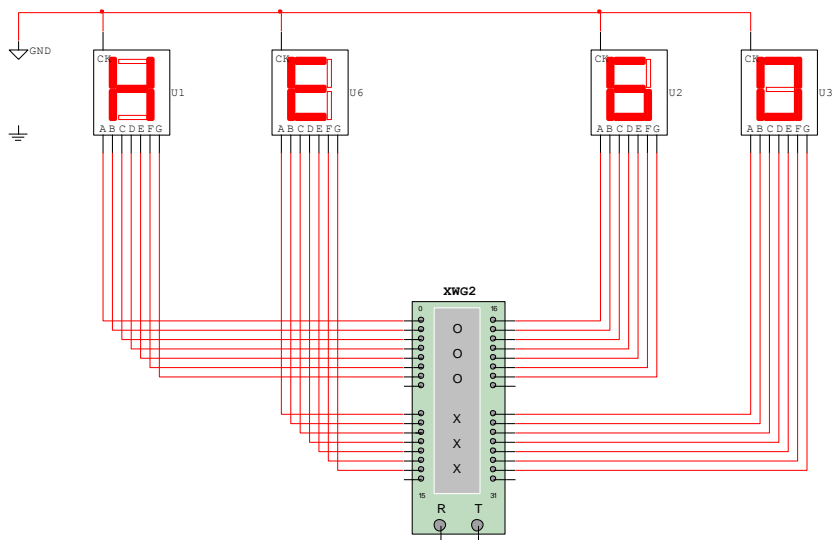


Рис. 71. Схема вывода статического слова на дисплей

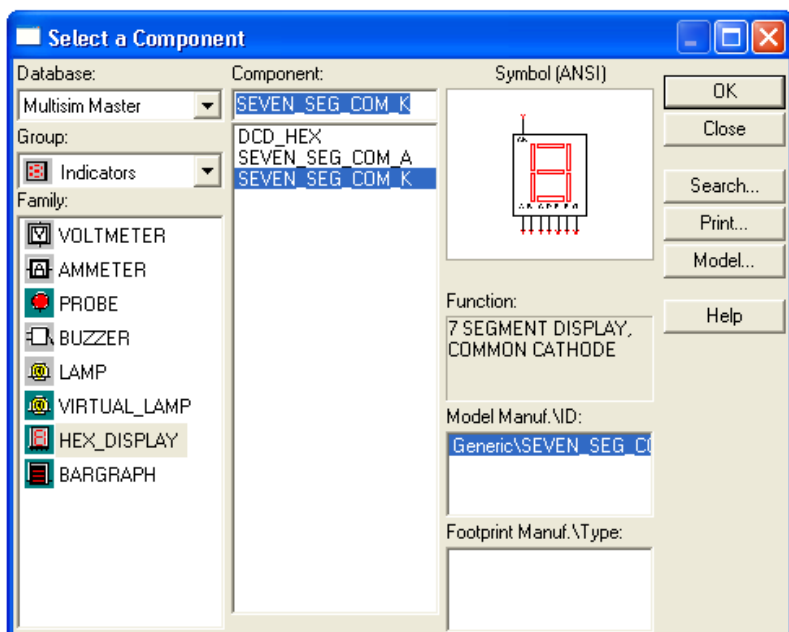


Рис. 72. Диалоговое окно *Select a Component*



4. После того как все четыре индикатора размещены на схеме, надо поместить на неё генератор слов *Word Generator*. Для этого на панели приборов (крайне правый столбец в окне схемы) выбрать *Word Generator* и поместить на схему. Затем соединить компоненты проводниками.

5. После того как схема нарисована, надо открыть лицевую панель генератора слов (рис. 73), дважды щелкнув по иконке левой клавишей мыши.

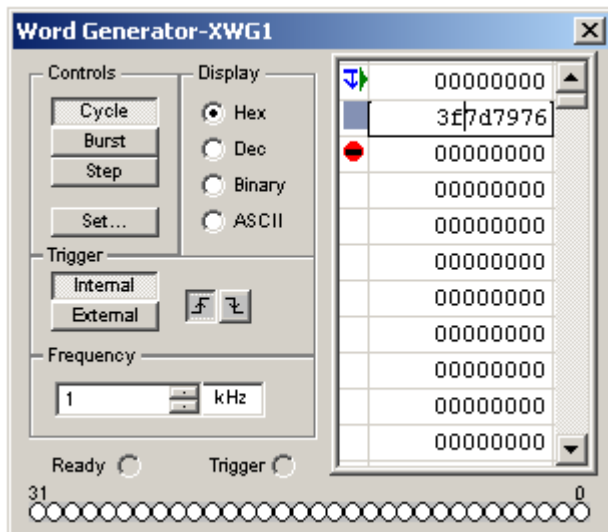



Рис. 73. Лицевая панель генератора слов (*Word Generator*)

6. В белом окне, расположенном слева на этой панели, установить точку останова (*Break-Point*) во второй строке. Установить курсор на первой строке. Во второй строке ввести коды букв, составляющих слово. Например, НЕБО – 3F7D7976. Включить процесс моделирования, переведя выключатель , расположенный справа во второй строке окна Multisim, в положение 1.

7. Нажать два раза клавишу *Step* на панели *Controls*. На индикаторах отобразится слово НЕБО.

### Содержание отчёта

1. Название пункта работы.
2. Исследуемая схема.
3. Результат моделирования.

## ВЫВОД «БЕГУЩЕЙ СТРОКИ» НА ДИСПЛЕЙ

**Цель работы** – на дисплее, состоящем из четырех семи-сегментных индикаторов, создать иллюзию «бегущей строки».

### Краткие сведения из теории

Предположим, что необходимо вывести на дисплей, состоящий из четырех семисегментных индикаторов, определенный текст произвольной длины. Прежде всего этот текст необходимо закодировать в семисегментных кодах.

Слова в тексте разделяются нулями. На эту строку будем смотреть через окно, в котором можно видеть только четыре буквы.

Создать иллюзию «бегущей строки» можно двумя способами: двигать строку относительно окна или двигать окно относительно строки.

В первый момент в окне можно видеть первые четыре буквы. Затем надо сдвинуть окно на одну букву, т.е. теперь в окне можно увидеть буквы с 2 по 5. Затем снова сдвинуть окно на одну букву и так далее, пока не дойдем до конца сообщения.

Кроме того, надо согласовать темп вывода текста на экран (вывод окна) с восприятием человеческого глаза. Текст (или окно) сдвигается на одну позицию влево (вправо) каждые 0,5 с.

Таким образом, процедура вывода «бегущей строки» должна содержать следующие шаги:

1. Вывод четырех букв в окно (дисплей).
2. Ввод задержки на 0,5 с.
3. Сдвиг окна на одну позицию влево.

В дальнейшем эти три шага повторяются. Пример выполнения первых трех шагов приведен на рис. 74. В данном примере в окне можно увидеть восемь букв.

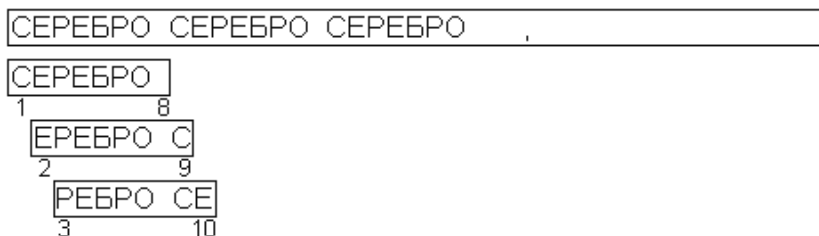


Рис. 74. Выводимая строка сообщения

Дисплей состоит из четырёх индикаторов, состоящих из восьми светодиодных сегментов, обозначенных латинскими буквами *a, b, c, d, e, f, g* и *h* (рис. 75). Каждому индикатору в памяти ЭВМ отводится своя ячейка памяти, из которой информация выводится на индикаторы.

Каждый бит ячейки памяти управляет состоянием соответствующего ему сегмента индикатора (см. лаб. работу №6). Для того чтобы заставить светиться какой-либо сегмент индикатора, надо в соответствующий ему разряд ячейки памяти записать единицу, если надо погасить этот сегмент – ноль.

Связь между сегментами индикатора и битами соответствующей ему ячейки памяти приведена в табл. 18.

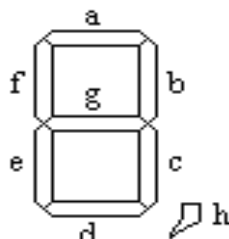


Рис. 75. Семисегментный индикатор

Т а б л и ц а 18

Бит ячейки памяти	7	6	5	4	3	2 <sub>0</sub>	1	0
Сегмент индикатора	h	g	f	e	d	c	b	a

Таким образом, на индикатор можно вывести некоторые буквы русского и латинского алфавитов и десятичные цифры от 0 до 9, если зажечь соответствующие сегменты индикатора. Но прежде надо определить коды этих букв и цифр.

Рассмотрим вывод строки НА ГОРЕ СНЕГ.

Правила определения кодов букв приведены в описании лабораторной работы №6. В табл. 19 приведены коды необходимых букв.

Т а б л и ц а 19

Буква	7	6	5	4	3	2	1	0	Коды букв
	h	g	f	e	d	c	b	a	
Н	0	1	1	1	0	1	1	0	76
А	0	1	1	1	0	1	1	1	77
Г	0	0	1	1	0	0	0	1	31
О	0	0	1	1	1	1	1	1	3f
Р	0	1	1	1	0	0	1	1	73
Е	0	1	1	1	1	0	0	1	79
С	0	0	1	1	1	0	0	1	39
У	0	1	1	0	1	1	1	0	6e
П	0	0	0	1	0	1	1	1	17
Ь	0	1	1	1	1	1	0	0	7c


В табл. 20 приведена программа вывода этой фразы.

Т а б л и ц а 20

Буква	7	6	5	4	3	2	1	0				
	h	g	f	e	d	c	b	a				
Н	7	6	0	0	0	0	0	0	Н	0	0	0
А	7	7	7	6	0	0	0	0	А	Н	0	0
О	0	0	7	7	7	6	0	0	0	А	Н	0
Г	3	1	0	0	7	7	7	6	Г	0	А	Н
О	3	F	3	1	0	0	7	7	О	Г	0	А
Р	7	3	3	F	3	1	0	0	Р	О	Г	0
Е	7	9	7	3	3	F	0	1	Е	Р	О	Г
О	0	0	7	9	7	3	3	F	0	Е	Р	О
С	3	9	0	0	7	9	7	3	С	0	Е	Р
Н	7	6	3	9	0	0	7	9	Н	С	0	Е
Е	7	9	7	6	3	9	0	0	Е	Н	С	0
Г	3	1	7	9	7	6	3	9	Г	Е	Н	С
	0	0	3	1	7	9	7	6	0	Г	Е	Н
	0	0	0	0	3	1	7	9	0	0	Г	Е
	0	0	0	0	0	0	3	1	0	0	0	Г

### Порядок выполнения работы

1. Собрать схему, изображенную на рис. 76. Поместить на неё четыре семисегментных индикатора с общими катодами и генератор слов.

2. Для того чтобы поместить индикаторы на схему, на панели инструментов (расположенной вертикально слева в окне схемы) выбрать клавишу индикаторы , появится диалоговое окно *Select a Component*. В этом диалоговом окне выбрать строку *HEX\_DISPLAY* (окно *Family*).

3. Во втором окне *Component* выбрать строку *SEVEN\_SEG\_COM\_K*. В окне *Symbol(ANSI)* появится изображение компонента. Нажать клавишу *OK*. Компонент прикрепится к курсору. Поместить компонент на схему и щелкнуть левой клавишей мыши. Компонент будет помещен на схему.

4. После того как все четыре индикатора помещены на схему, поместить на схему генератор слов *Word Generator*. Для этого на панели приборов (крайне правый столбец в окне схемы) выбрать *Word Generator* и поместить его на схему. Затем соединить компоненты проводниками.

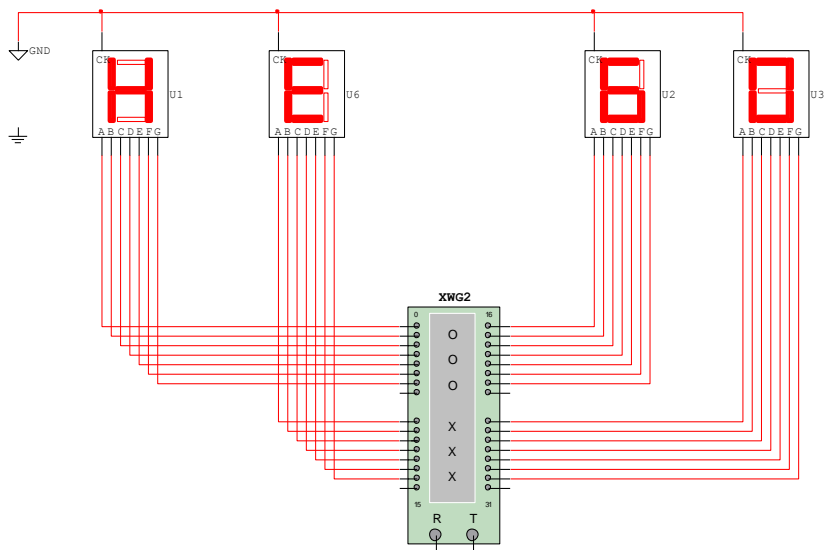


Рис. 76. Схема вывода «бегущей строки» на дисплей

5. После того как схема нарисована, открыть лицевую панель генератора слов (рис. 77), дважды щелкнув по иконке левой клавишей мыши.

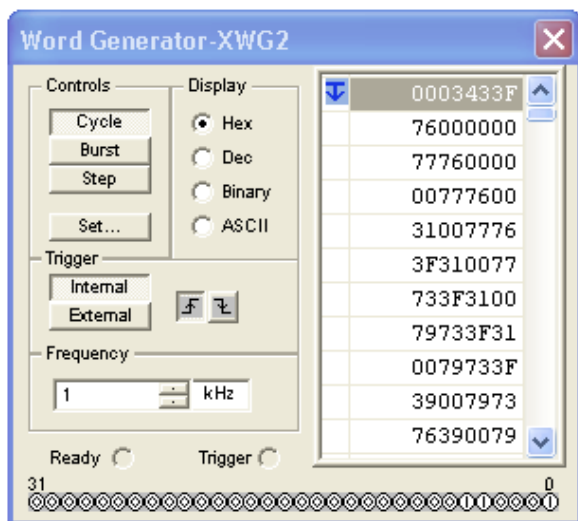


Рис. 77. Лицевая панель генератора слов (Word Generator)

6. В белом окне, расположенном слева на этой панели, установить точку останова (*Break-Point*). Установить курсор на первой строке.

7. Начиная со второй строки, ввести коды букв выводимого сообщения.

8. Включить процесс моделирования, переведя выключатель



, расположенный справа во второй строке окна *Multisim*, в положение 1.

9. Нажать клавишу *Burst* на панели *Controls*. На индикаторах отобразится бегущая строка.

### Содержание отчёта

1. Название пункта работы.
2. Исследуемая схема.
3. Результат моделирования.

## СОДЕРЖАНИЕ

Краткое описание пакета Multisim .....	3
Лабораторная работа №1. Исследование логических схем .....	22
Лабораторная работа №2. Исследование регистров, счетчиков и дешифраторов .....	31
Лабораторная работа № 3. Исследование генератора псевдослучайной последовательности .....	46
Лабораторная работа №4. Исследование арифметического сумматора .....	54
Лабораторная работа №5. Исследование работы мультиплексора и демультиплексора .....	62
Лабораторная работа № 6. Вывод статического слова .....	69
Лабораторная работа № 7. Вывод «бегущей строки» на дисплей .....	74

*Бесперстов Эдуард Александрович*

### **Моделирование цифровых устройств в среде Multisim 7**

Редактор *Г.М. Звягина*

Корректор *Л.А. Петрова*

Компьютерная вёрстка: *О.М. Дмитриева*

Подписано в печать 12.04.2018. Формат 60х84/16. Бумага документная.

Печать трафаретная. Усл. печ. л. 4,6. Тираж 100 экз. Заказ № 43.

Балтийский государственный технический университет

Типография БГТУ

190005, С.-Петербург, 1-я Красноармейская ул., д.1