

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БАРАНОВИЧСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

**ОРГАНИЗАЦИЯ
И ФУНКЦИОНИРОВАНИЕ ЭВМ**

**Лабораторный практикум
для студентов II курса
специальности 1-40 01 02
Информационные системы и технологии**

**Барановичи
РИО БарГУ
2011**

УДК 681.3.06:378.244(076)
ББК 22.2:74.58я73
О-64

Рекомендовано к печати методической комиссией
инженерного факультета

С о с т а в и т е л ь

М. В. Филатов

Р е ц е н з е н т ы:

М. В. Нерода, кандидат технических наук,
заведующий кафедрой технологии машиностроения БарГУ;
О. А. Синдель, инженер-программист ЗАО «Атлант»

О-64 **Организация и функционирование ЭВМ** [Текст] : лаб. практикум
для студентов II курса специальности 1-40 01 02 Информационные
системы и технологии / сост. М. В. Филатов. — Барановичи : РИО
БарГУ, 2011. — 76, [4] с. : ил. — 70 экз. — ISBN 978-985-498-424-7.

Включает в себя лабораторные работы, содержащие краткий теоретический материал,
задания для самостоятельного выполнения, а также контрольные вопросы.

Предназначено для студентов II курса инженерного факультета дневной формы обуче-
ния специальности «Информационные системы и технологии» БарГУ.

Рис. 52.

УДК 681.3.06:378.244(076)
ББК 22.2:74.58я73

ISBN 978-985-498-424-7

© БарГУ, 2011

ОГЛАВЛЕНИЕ

Введение	4
Описание программного пакета Electronics Workbench	5
Лабораторная работа 1 Логические элементы. Логический элемент с тремя состояниями	19
Часть 1 Логические элементы	19
Часть 2 Логический элемент с тремя состояниями	25
Лабораторная работа 2 Цифровые компараторы. Устройство контроля четности	29
Часть 1 Цифровые компараторы	29
Часть 2 Устройство контроля четности	30
Лабораторная работа 3 Мультиплексоры и демультиплексоры	34
Лабораторная работа 4 Шифраторы и дешифраторы	39
Лабораторная работа 5 Арифметические сумматоры. Арифметико-логическое устройство	43
Часть 1 Арифметические сумматоры	43
Часть 2 Арифметико-логическое устройство	46
Лабораторная работа 6 Триггерные схемы	50
Лабораторная работа 7 Счетчики. Регистры	54
Часть 1 Счетчики	54
Часть 2 Регистры	60
Лабораторная работа 8 Оперативное запоминающее устройство. Постоянное запоминающее устройство	66
Часть 1 Оперативные запоминающие устройства	66
Часть 2 Постоянное запоминающее устройство	71
Список использованных источников	77

ВВЕДЕНИЕ


Дисциплина «Организация и функционирование ЭВМ» ориентирована в первую очередь на формирование у студентов теоретических основ электроники и методологии и методики решения различных задач, получение теоретических знаний и практических навыков использования и проектирования современных электронных средств управления и контроля технологическими объектами машиностроения.


Издание включает лабораторные работы, направленные на изучение принципов действия, характеристик и особенностей работы радиоэлектронных устройств с помощью программного пакета Electronic Workbench (EWB), который позволяет достаточно легко и быстро собирать схемы разной степени сложности. Наглядность представления способствует лучшему усвоению материала, позволяет получить первичные навыки работы с инженерным программным пакетом. На примере EWB происходит знакомство с компьютерными методами проектирования и анализа схем, все шире используемыми в современной инженерной практике. С помощью EWB можно не только собирать схемы и просматривать входные и выходные сигналы, но и проводить различные виды анализа (спектральный, частотный, шумовой), рассматривать влияние различных дестабилизирующих факторов, например, изменение емкостей, сопротивлений и индуктивностей элементов, изменение температурных коэффициентов для различных точек схемы.


ОПИСАНИЕ ПРОГРАММНОГО ПАКЕТА ELECTRONICS WORKBENCH

Меню программы

File Menu — меню работы с файлами:

New — создает новый файл, эквивалентна кнопке  управления программой;

Open — открывает уже существующий файл, эквивалентна кнопке  управления программой;


Save — сохраняет рабочий файл, эквивалентна кнопке  управления программой;

Save As — сохраняет рабочий файл под новым именем;

Revert to Saved (Revert) — восстанавливает на рабочем поле последнюю записанную версию файла;

Import — импортирует файл с расширением .net или .cir и преобразование его в схему;

Export — экспортирует рабочий файл с расширением .net, .scr, .cmp, .cir, .plc;


Print — печать схемы и инструментов, эквивалентна кнопке  управления программой;


Print Setup (Windows) — установка параметров печати;


Exit (Windows) — закрытие файла и выход из программы (эквивалентна кнопке);

Install (Windows) — установка дополнительных программных продуктов EWB.

Edit Menu — меню редактирования схемы:

Cut — вырезать в буфер выделенные элементы схемы (при этом выделенные элементы удаляются из схемы), эквивалентна кнопке  управления программой;

Copy — копировать в буфер выделенные элементы схемы (при этом выделенные элементы сохраняются в схеме), эквивалентна кнопке  управления программой;

Paste — вставить из буфера выделенные элементы в схему, эквивалентна кнопке  управления программой;


Delete — стереть выделенные элементы из схемы, эквивалентна клавише <Delete> на клавиатуре компьютера;


Select All (Windows) — выделить все;


Copy as Bitmap (Windows) — копировать участок рабочего поля в буфер в формате .bmp;


Show Clipboard — показать содержимое буфера.


Circuit Menu — меню редактирования элементов схемы:


Rotate — повернуть выделенный элемент на 90° по часовой стрелке, эквивалентна кнопке  управления программой;


Flip Vertical — отобразить выделенный элемент зеркально относительно вертикальной оси, эквивалентна кнопке  управления программой;

Flip Horizontal — отобразить выделенный элемент зеркально относительно горизонтальной оси, эквивалентна кнопке  управления программой;

Component Properties — открыть окно со свойствами выделенного элемента схемы, эквивалентна кнопке  управления программой;


Create Subcircuit — компоновка части схемы в отдельную подсхему, эквивалентна кнопке  управления программой;


Zoom In — увеличение масштаба схемы, эквивалентна кнопке  управления программой;

Zoom Out — уменьшение масштаба схемы, эквивалентна кнопке  управления программой;

Schematic Options — настройка параметров схемы.

Analysis Menu — меню анализа:

Activate — включение схемы (элемент включения тумблера питания), математическое вычисление текущих значений сигналов (симуляция), эквивалентна кнопке  управления программой;

Pause — останов схемы, эквивалентна кнопке  управления программой;

Resume — продолжение симуляции с момента останова;

Stop — прекращение симуляции и сброс текущих значений сигналов в «0»;

Analysis Options — установка параметров симуляции;

DC Operating Point — анализ работы схемы по постоянному току;

AC Frequency — анализ работы схемы в режиме малых сигналов;

Transient — анализ переходных процессов в схеме;

Fourier — анализ Фурье сигналов в схеме;

Noise — анализ шумовых характеристик схемы;

Distortion — анализ нелинейных искажений в схеме;

Parameter Sweep — анализ характеристик схемы при изменении одного из параметров в определенных пределах;

Temperature Sweep — температурный анализ схемы;


Pole-Zero — определение нулей и полюсов передаточной функции схемы;

Transfer Function — анализ передаточной функции схемы;

Sensitivity — анализ чувствительности схемы;

Worst Case — анализ устойчивости схемы;

Monte Carlo — анализ Монте-Карло;

Graphs — активизация окна диаграмм анализа, эквивалентна кнопке  управления программой.

Window Menu — меню окон:

Arrange — располагает рабочее окно по площади экрана;

Circuit — располагает рабочее окно на передний план;

Description — открывает окно примечаний;

Help Menu — меню справки.

Кнопки управления программой



— создает новый файл, соответствует выбору меню **File Menu/New**;



— открывает уже существующий файл, соответствует выбору меню **File Menu/Open**;



— сохраняет рабочий файл, соответствует выбору меню **File Menu/Save**;



— печать схемы и инструментов, соответствует выбору меню **File Menu/Print**;



— вырезать в буфер выделенные элементы схемы (при этом выделенные элементы удаляются из схемы), соответствует выбору меню **Edit Menu/Cut**;



— копировать в буфер выделенные элементы схемы (при этом выделенные элементы сохраняются в схеме), соответствует выбору меню **Edit Menu/Copy**;



— вставить из буфера выделенные элементы в схему, соответствует выбору меню **Edit Menu/Paste**;



— повернуть выделенный элемент на 90° по часовой стрелке, соответствует выбору меню **Circuit Menu/Rotate**;



— отобразить выделенный элемент зеркально относительно вертикальной оси, соответствует выбору меню **Circuit Menu/Flip Vertical**;



— отобразить выделенный элемент зеркально относительно горизонтальной оси, соответствует выбору меню **Circuit Menu/Flip Horizontal**;



— открыть окно свойств выделенного элемента схемы, соответствует выбору меню **Circuit Menu/Component Properties**;



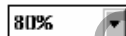
— компоновка части схемы в отдельную подсхему, соответствует выбору меню **Circuit Menu/Create Subcircuit**;



— увеличение масштаба схемы, соответствует выбору меню **Circuit Menu/Zoom In**;



— уменьшение масштаба схемы, соответствует выбору меню **Circuit Menu/Zoom Out**;



— окно текущего масштаба схемы;



— окно выбора справки о программе, свойствах элементов и приборов;



— включение схемы (эквивалент включения тумблера питания), математическое вычисление текущих значений сигналов (симуляция), соответствует выбору меню **Analysis Menu/Activate**;



— останов схемы, соответствует выбору меню **Analysis Menu /Pause**.

Библиотеки элементов



— библиотека микросхем, которые использовались в данной схеме;



— **SOURCES** — библиотека источников питания и источников сигналов;



— **BASIC** — библиотека базовых элементов: резисторов, конденсаторов, индуктивностей и др.;



— **DIODES** — библиотека диодов;



— **TRANSISTORS** — библиотека транзисторов;



— **ANALOG ICs** — библиотека микросхем операционных усилителей;



— **MIXED ICs** — библиотека микросхем аналого-цифровых и цифро-аналоговых преобразователей;



— **DIGITAL ICs** — общая библиотека цифровых микросхем;



— **LOGIC GATES** — библиотека функциональных логических элементов и микросхем на базе логических элементов;



— **DIGITAL** — библиотека функциональных элементов триггеров, микросхем на базе триггеров, микросхем мультиплексоров и др.;



— **INDICATORS** — библиотека индикаторов;



— **CONTROLS** — библиотека функциональных аналоговых устройств: сумматоров, умножителей, ограничителей и др.;



— **MISCELLANEOUS** — библиотека смешанных элементов: предохранителей, кварцевых генераторов, моторов и др.;

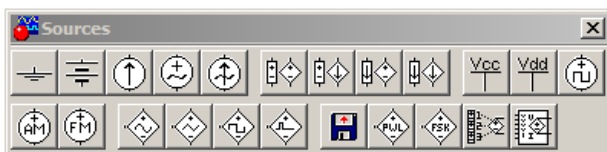




— **INSTRUMENTS** — библиотека инструментов: генераторов различных сигналов и измерительных приборов.

Состав библиотек

В составе библиотек описаны элементы, которые необходимы или могут понадобиться при выполнении лабораторных работ по курсу «Цифровые устройства и микропроцессоры». Полное описание состава библиотек можно найти в HELP программы Electronics Workbench.






Библиотека SOURCES



-  — **Ground** — «земля» для микросхем и подача на схему логического нуля.
-  — **+5 V Voltage Source** — источник питания +5 В для микросхем серии ТТЛ.

Библиотека BASIC




-  — **Connector** — точка соединения проводников;
-  — **Resistor** — резистор (по умолчанию 1 кОм);
-  — **Capacitor** — конденсатор (по умолчанию 1 мкФ);
-  — **Switch** — переключатель;
-  — **Pull-up Resistor** — логическая 1 (резистор 1 кОм на +5 В).


Библиотека DIGITAL ICs



В состав библиотеки **DIGITAL ICs** входят микросхемы 74 серии ТТЛ (аналоги отечественных серий: LS-555, F-1531, ALS-1533, AC-1554, HC-1564, АСТ-1594).

-  — **74XX Template** — микросхемы 7400...7493:
- 7400 4 элемента 2И—НЕ;
- 7402 4 элемента 2ИЛИ—НЕ;
- 7403 4 элемента 2И—НЕ с открытым коллектором;
- 7404 6 элементов НЕ;
- 7405 6 элементов НЕ с открытым коллектором;
- 7406 6 буферных инверторов;
- 7407 6 буферных формирователей с открытым коллектором;
- 7408 4 элемента 2И;
- 7409 4 элемента 2И с открытым коллектором;
- 7410 3 элемента 3И—НЕ;
- 7411 3 элемента 3И;
- 7412 3 элемента 3И—НЕ с открытым коллектором;
- 7414 6 триггеров Шмитта с инверторами;
- 7415 3 элемента 3И;
- 7416 6 буферных элементов НЕ;

7417 6 буферных элементов с открытым коллектором;
 7420 2 элемента 4И—НЕ;
 7421 2 элемента 4И;
 7422 2 элемента 4И—НЕ с открытым коллектором и повышенной нагрузочной способностью;
 7425 2 элемента 4ИЛИ—НЕ со стробированием;
 7426 4 высоковольтных элемента 2И—НЕ с открытым коллектором;
 7427 3 элемента 3ИЛИ—НЕ;
 7428 4 буферных элемента 2ИЛИ—НЕ;
 7430 элемент 8И—НЕ;
 7432 4 элемента 2ИЛИ;
 7433 4 элемента 2ИЛИ—НЕ с открытым коллектором;
 7437 4 буферных элемента 2И—НЕ;
 7438 4 буферных элемента 2И—НЕ с открытым коллектором;
 7439 4 элемента 2И—НЕ с открытым коллектором;
 7440 2 элемента 4И—НЕ с большим коэффициентом разветвления по выходу;
 7442 дешифратор 4×10 ;
 7445 дешифратор двоично-десятичного кода в унитарный;
 7447 дешифратор двоично-десятичного кода в код семисегментного индикатора;
 7451 2 элемента 4—2—3—2И—ИЛИ—НЕ;
 7454 элемент 2—3—3—2И—4ИЛИ—НЕ;
 7455 элемент 4—4И—2ИЛИ—НЕ с возможностью расширения по ИЛИ;
 7469 2 асинхронных 4-разрядных двоичных счетчика;
 7472 триггер JK с логическим элементом И на выходе;
 7473 2 динамических JK-триггера со сбросом;
 7474 2 динамических D-триггера со сбросом и установкой;
 7475 4 статических D-триггера с прямым и инверсным выходами;
 7476 2 динамических JK-триггера со сбросом и установкой;
 7477 4 статических D-триггера;
 7478 двойной JK-триггер;
 7486 4 2-входовых элемента ИСКЛЮЧАЮЩЕЕ ИЛИ;
 7490 двоично-десятичный асинхронный 4-разрядный счетчик;
 7491 8-разрядный сдвигающий регистр;
 7492 асинхронный счетчик-делитель на 12;
 7493 асинхронный двоичный счетчик.

 — 741XX Template — микросхемы 74107...74199:

74107 2 JK-триггера типа MS со сбросом;
 74109 2 динамических JK-триггера;
 74112 2 JK-триггера типа MS с входами сброса и установки;
 74113 2 JK-триггера типа MS с установкой;
 74114 2 сдвоенных JK-триггера типа MS с установкой и общим сбросом;
 74116 2 счетверенных статических D-триггера со сбросом;
 74125 4 буферных элемента с тремя состояниями на выходе;
 74126 4 буферных элемента с тремя состояниями на выходе;
 74132 4 триггера Шмитта;

10

- 74133 элемент 13И—НЕ;
- 74134 элемент 12И—НЕ с тремя состояниями;
- 74138 двоичный дешифратор на восемь направлений;
- 74139 2 дешифратора-демультиплексора;
- 74145 дешифратор на 10 выходов с открытым коллектором;
- 74147 приоритетный шифратор 10—4;
- 74148 приоритетный шифратор 8—3;
- 74150 селектор-мультиплексор данных на 16 каналов со стробированием;
- 74151 селектор-мультиплексор на 8 каналов со стробированием;
- 74153 сдвоенный цифровой селектор-мультиплексор 4—1;
- 74154 дешифратор-демультиплексор 4—16;
- 74155 сдвоенный дешифратор-демультиплексор 2—4;
- 74156 сдвоенный дешифратор 2—4 с открытым коллектором;
- 74157 селектор-мультиплексор 2—1;
- 74158 4 мультиплексора 2—1 с инверсными выходами;
- 74159 дешифратор-демультиплексор 4—16 с открытым коллектором;
- 74160 синхронный двоично-десятичный 4-разрядный счетчик со сбросом;
- 74162 синхронный двоично-десятичный 4-разрядный счетчик;
- 74163 синхронный 4-разрядный двоичный счетчик с синхронным сбросом;
- 74164 8-разрядный последовательный сдвигающий регистр с параллельным выходом;
- 74165 8-разрядный регистр сдвига с параллельным вводом информации;
- 74166 8-разрядный регистр сдвига с возможностью синхронного параллельного ввода информации;
- 74169 синхронный 4-разрядный двоичный реверсивный счетчик;
- 74173 4-разрядный регистр с тремя состояниями;
- 74174 6 динамических D-триггеров;
- 74175 4 динамических D-триггера;
- 74181 арифметико-логическое устройство;
- 74190 синхронный реверсивный двоично-десятичный счетчик;
- 74191 синхронный реверсивный двоичный счетчик;
- 74192 синхронный двоично-десятичный реверсивный счетчик;
- 74194 4-разрядный универсальный регистр сдвига;
- 74195 4-разрядный регистр сдвига с параллельным вводом информации;
- 74198 реверсивный 8-разрядный регистр сдвига с параллельной загрузкой;
- 74199 8-разрядный регистр сдвига с параллельной загрузкой.



— 742XX Template — микросхемы 74238...74298:

- 74238 дешифратор—демультиплексор 3—8;
- 74240 2-канальный 8-разрядный формирователь с тремя состояниями на выходе и инверсией сигнала;
- 74241 2-канальный 8-разрядный формирователь с тремя состояниями;
- 74244 8-канальный однонаправленный формирователь с симметричным управлением;
- 74251 8-ходовый селектор-мультиплексор с тремя состояниями на выходе;
- 74253 2-разрядный 4-канальный коммутатор с тремя состояниями на выходе;
- 74257 4-разрядный селектор 2—1 с тремя состояниями;
- 74258 4-разрядный селектор 2—1 с тремя состояниями и инверсией сигналов на выходе;

74266 4 элемента 2ИСКЛЮЧАЮЩЕЕ ИЛИ—НЕ;
74273 8-разрядный регистр с установкой в 0;
74279 4 RS-триггера;
74280 9-разрядная схема контроля четности и нечетности;
74290 асинхронный двоично-десятичный счетчик;
74293 асинхронный двоичный счетчик;
74298 4 2-входовых мультиплексора с запоминанием.



— 743XX Template — микросхемы 74350...74395:

74350 4-разрядный статический D-триггер с тремя состояниями на выходе;
74352 двоянный селектор-мультиплексор 4—1;
74353 двоянный селектор-мультиплексор 4—1 с тремя состояниями на выходе;
74365 6 формирователей с тремя состояниями с элементом управления по входу;
74367 6 формирователей с тремя состояниями с раздельным управлением по входу;
74368 6 формирователей с тремя состояниями;
74373 8-разрядный регистр на триггерах-зашелках с тремя состояниями на выходе;
74374 8-разрядный регистр на триггерах-зашелках с тремя состояниями на выходе;
74375 4-разрядный статический D-триггер;
74377 8-разрядный регистр с регулированием записи информации;
74378 6 динамических D-триггера с разрешением записи;
74379 4 динамических D-триггера с разрешением записи;
74393 2 4-разрядных двоичных счетчика;
74395 4-разрядный параллельный регистр сдвига.



— 744XX Template — микросхемы 74445...74466:

74445 дешифратор двоично-десятичного кода в унитарный;
74465 8 буферов тремя состояниями на выходе;
74466 8 буферов с инверсией и с тремя состояниями на выходе.

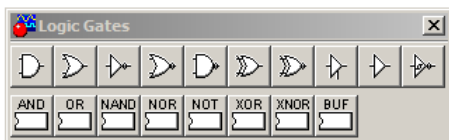


— 4XXX Template — микросхемы 4000...4556 серии КМОП (аналог серии 561, 1561):

4000 2 элемента 3ИЛИ—НЕ и элемент НЕ;
4001 4 элемента 2ИЛИ—НЕ;
4002 2 элемента 4ИЛИ—НЕ;
4008 4-разрядный полный сумматор;
4009 6 преобразователей уровня с инверсией;
4010 6 преобразователей уровня без инверсии;
4011 4 элемента 2И—НЕ;
4012 2 элемента 4И—НЕ;
4013 2 D-триггера с установкой в 0 и 1;
4014 8-разрядный статический D-триггер;
4015 двоянный 4-разрядный регистр сдвига;
4017 десятичный счетчик с дешифратором;
12

- 4019 4 элемента И—ИЛИ;
- 4023 3 элемента ЗИ—НЕ;
- 4024 6-разрядный двоичный счетчик;
- 4025 3 элемента ЗИЛИ—НЕ;
- 4027 2 JK-триггера;
- 4028 дешифратор 4×10 ;
- 4030 4 элемента ИСКЛЮЧАЮЩЕЕ ИЛИ;
- 4040 12-каскадный двоичный счетчик;
- 4041 4 буфера с прямыми и инверсными выходами;
- 4042 4 D-триггера;
- 4043 4 RS-триггера;
- 4044 4 RS-триггера с тремя состояниями;
- 4049 6 элементов НЕ;
- 4050 6 преобразователей уровня;
- 4066 счетверенный двунаправленный переключатель;
- 4068 элемент 8И—НЕ;
- 4069 6 инверторов;
- 4070 4 элемента ИСКЛЮЧАЮЩЕЕ ИЛИ;
- 4071 4 элемента 2ИЛИ;
- 4072 2 элемента 4ИЛИ;
- 4073 3 элемента ЗИ;
- 4075 3 элемента ЗИЛИ;
- 4076 4 динамических D-триггера с третьим состоянием;
- 4077 4 элемента ИСКЛЮЧАЮЩЕЕ ИЛИ—НЕ;
- 4078 элемент 8ИЛИ—НЕ;
- 4081 4 элемента 2И;
- 4082 2 элемента 4И;
- 4085 2 комбинированных устройства на элементах И, ИЛИ, НЕ;
- 4086 комбинированное устройства на элементах И, ИЛИ, НЕ;
- 4093 ТЛ1 4 триггера Шмитта с элементами 2И—НЕ;
- 4502 ЛН1 6 стробируемых элементов НЕ;
- 4503 — 6 буферов с тремя состояниями;
- 4508 — 2 4-разрядных статических D-триггера;
- 4510 — синхронный двоично-десятичный реверсивный счетчик с параллельной загрузкой;
- 4511 дешифратор двоично-десятичного кода в код семисегментного индикатора с защелками;
- 4512 мультиплексор 8-1 с тремя состояниями;
- 4514 дешифратор-демультиплексор 4-16 с защелками на входе;
- 4515 дешифратор-демультиплексор 4-16 с защелками на входе;
- 4516 4-х разрядный двоичный реверсивный счетчик;
- 4518 2 двоично-десятичных счетчика;
- 4520 2 4-разрядных двоичных счетчика;
- 4532 приоритетный шифратор на 8 входов;
- 4556 2 дешифратора-демультиплексора 2-4.

Библиотека LOGIC GATES



— **2-Input AND** — элемент 2И;



— **2-Input OR** — элемент 2ИЛИ;



— **NOT Gate** — элемент НЕ;



— **2-Input NOR Gate** — элемент 2ИЛИ—НЕ;



— **2-Input NAND Gate** — элемент 2И—НЕ;



— **2-Input XOR Gate** — элемент ИСКЛЮЧАЮЩЕЕ ИЛИ;



— **2-Input XNOR Gate** — элемент ИСКЛЮЧАЮЩЕЕ ИЛИ—НЕ;



— **Tristate Buffer** — буфер с тремя состояниями;



— **Buffer** — буферный элемент;



— **Schmitt Trigger** — триггер Шмитта;



— **And Gates** — микросхемы, в состав которых входят элементы И;



— **OR Gates** — микросхемы, в состав которых входят элементы ИЛИ;



— **NAND Gates** — микросхемы, в состав которых входят элементы И—НЕ;



— **NOR Gates** — микросхемы, в состав которых входят элементы ИЛИ—НЕ;



— **NOT Gates** — микросхемы, в состав которых входят элементы НЕ;



— **XOR Gates** — микросхемы, в состав которых входят элементы ИСКЛЮЧАЮЩЕЕ ИЛИ;

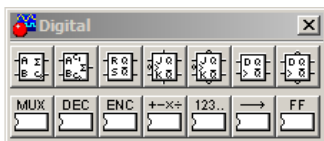


— **XNOR Gates** — микросхемы, в состав которых входят элементы ИСКЛЮЧАЮЩЕЕ ИЛИ—НЕ;



— **Buffers** — микросхемы, в состав которых входят буферные элементы.

Библиотека DIGITAL





— **Half Adder** — полусумматор;



— **Full Adder** — полный сумматор;



— **RS Flip-Flop** — RS-триггер;



— **JK Flip-Flop with Active High Asynchronous Inputs** — JK-триггер с асинхронными входами установки, активными по «1»;



— **JK Flip-Flop with Active Low Asynchronous Inputs** — JK-триггер с асинхронными входами установки, активными по «0»;



— **D Flip-Flop** — динамический D-триггер;



— **D Flip-Flop with Active Low Asynchronous Inputs** — D-триггер с асинхронными входами установки, активными по «0»;



— **Multiplexer ICs** — микросхемы мультиплексов;



— **Demultiplexer ICs** — микросхемы демультимплексов;



— **Encoder ICs** — микросхемы декодеров;



— **Arithmetic ICs** — микросхемы арифметико-логических устройств;



— **Counter ICs** — микросхемы счетчиков;



— **Shift Register ICs** — микросхемы регистров;



— **Flip-Flops ICs** — микросхемы триггеров.

Библиотека INDICATORS



— **Voltmeter** — вольтметр;



— **Ammeter** — амперметр;



— **Bulb** — лампочка;



— **Probe** — светодиод;



— **7-Segment Display** — семисегментный индикатор;



— **Decoded 7 Segment Display** — семисегментный индикатор с декодером;



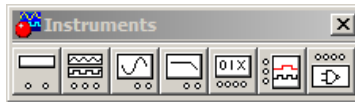
— **Buzzer** — автоматический прерыватель;





— **Bargraph Display** — матричный дисплей;





— **Decoded Bargraph Display** — матричный дисплей с декодером.





 — **Multimeter** — комбинированный прибор, позволяющий измерять постоянную и переменную составляющие тока, напряжения, а также сопротивление и разницу в дБ между двумя различными точками цепи;


 — **Function Generator** — источник напряжения, который обеспечивает на выходе синусоидальный, треугольный или прямоугольный сигнал. Можно изменять форму, частоту, амплитуду, коэффициент заполнения и постоянную составляющую сигнала;

 — **Oscilloscope** — двухканальный осциллограф, показывающий величину и временные диаграммы напряжения одного или двух сигналов, а также позволяющий сравнивать один сигнал с др.;

 — **Bode Plotter** — графопостроитель, позволяющий измерять АЧХ и ФЧХ исследуемой схемы;

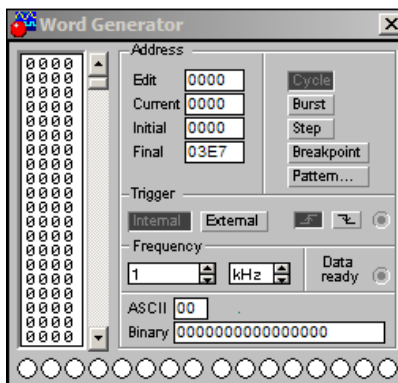
 — **Word Generator** — генератор логических сигналов, позволяющий выводить на 16 выходов различные комбинации логических сигналов;

 — **Logic Analyzer** — логический анализатор, позволяющий наблюдать временные диаграммы до 16 логических сигналов;

 — **Logic Converter** — логический конвертор, позволяющий взаимно преобразовывать логические выражения в таблицы истинности и схемные решения.

Word Generator

WORD GENERATOR — генератор логических сигналов. После двойного щелчка на значке инструмента на схеме генератор имеет следующий вид:



С левой стороны генератора располагается окошко задания сигналов с полосой прокрутки.

В центре сверху — поле **Address**, которое показывает номер строки (в шестнадцатеричных числах) из окошка задания сигналов.

В верхнем правом поле находятся кнопки управления выводом сигналов.

В центре — поле **Trigger** — поле выбора режима синхронизации генератора и входная клемма для ввода внешнего сигнала синхронизации.

Ниже находится поле задания частоты изменения выходных сигналов и выходная клемма сигнала синхронизации.

Еще ниже располагается поле с отображением текущей строки сигнала в коде **ASCII** и двоичном коде **Binary**.

Внизу генератора находятся 16 выходных клемм.

В окошке задания сигналов располагаются строки из четырех шестнадцатеричных чисел, что соответствует двоичному 16-разрядному слову (окошко **Binary**). Величины шестнадцатеричных чисел могут быть в пределах от 0000 до FFFF. Когда включается схема, в начальный момент времени первая строка битов появляется на соответствующих выходных клеммах, т. е. старший (левый) бит на левой клемме. В следующий момент времени на выходные клеммы генератора выводится вторая строка, затем третья и т. д. Менять частоту следования слов можно в окошке **Frequency**. По умолчанию частота равна 1 кГц.

Для изменения величины числа в генераторе нужно изменить:

- двоичное число в окошке **Binary**

или

- шестнадцатеричное число в окошке задания сигналов,

или

- число в коде **ASCII** в окошке **ASCII** (при этом изменятся шестнадцатеричные и двоичные числа в соответствующих окошках).

Можно менять режим вывода 16-битного слова на выходные клеммы генератора с помощью кнопок **Step** (пошаговый режим), **Burst** (однократный режим) или **Cycle** (цикловый режим):

- чтобы вывести одно слово нужно нажать кнопку **Step** (строка выводимого слова выделяется инверсией);

- при нажатии кнопки **Burst** на выходах генератора будут появляться последовательно 16-битные слова начиная со строки, номер которой записан в окошке **Initial**, и заканчивая словом, номер которого записан в окошке **Final**.

- нажатие кнопки **Cycle** аналогично нажатию кнопки **Burst**, только сигнал на выходах генератора будет непрерывный. Остановить вывод сигнала можно кнопкой включения схемы или кнопкой **Pause**.

Задать точку останова можно также с помощью кнопки **Breakpoint**. Для этого нужно поставить маркер на нужной строке и нажать на кнопку **Breakpoint**. Строка при этом выделяется серым цветом. Для того чтобы убрать точку останова, необходимо снова установить маркер в строке останова и нажать на кнопку **Breakpoint**.

Нажав на кнопку **Pattern**, можно выбрать из стандартных вариантов сигнал, который появится в окошке задания сигнала. При необходимости его можно отредактировать.

Logic Analyzer

LOGIC ANALYZER — логический анализатор, который показывает уровни (а не измеряет их; этим он и отличается от обычного осциллографа) до 16 цифровых сигналов в схеме.

Основное центральное окошко — информационное, где выводится графическое изображение исследуемых сигналов.

Слева на панели LOGIC ANALYZER — 16 входных клемм.

Внизу расположены:

- кнопка **Stop**, останавливающая вывод информации на экран анализатора;
- кнопка **Reset**, сбрасывающая информацию с экрана анализатора;
- поле **Time Base** — чувствительность развертки анализатора. Размерность — время/деление.
- поле **Clock** — поле выбора и установки режима синхронизации;
- поле **Triggering** — поле выбора комбинации входных сигналов.

По краям экраны расположены красная и синяя измерительные линии. Нажимая на верхние треугольники и передвигая линии влево и вправо, можно менять положение линий на экране и, соответственно, появится информация о текущем времени в секундах в центральном нижнем окошке и величине 16-битного слова данных (в шестнадцатеричной форме).

Репозиторий Барнауль

ЛАБОРАТОРНАЯ РАБОТА 1

ЛОГИЧЕСКИЕ ЭЛЕМЕНТЫ.

ЛОГИЧЕСКИЙ ЭЛЕМЕНТ С ТРЕМЯ СОСТОЯНИЯМИ

Цели:

- познакомиться с программой для моделирования электронных схем Electronics Workbench, научиться создавать в ней простейшие электронные схемы, исследовать основные логические элементы с ее помощью;
- провести исследование логического элемента с тремя состояниями, изучить особенности каждого из них.

Часть 1

Логические элементы

Теоретические сведения

Известно, что математической основой цифровых вычислительных устройств является двоичная арифметика, в которой используется всего два числа «0» и «1». Выбор двоичной системы счисления диктовался требованиями простоты технической реализации самых сложных задач с использованием всего одного базового элемента — ключа, который имеет два состояния: включен (замкнут) или выключен (разомкнут). Если первое состояние ключа принять за условную (логическую) «1», то второе будет отражать условный (логический) «0» или наоборот. Возможные комбинации показаны на рисунках 1.1, 1.2 и 1.3.

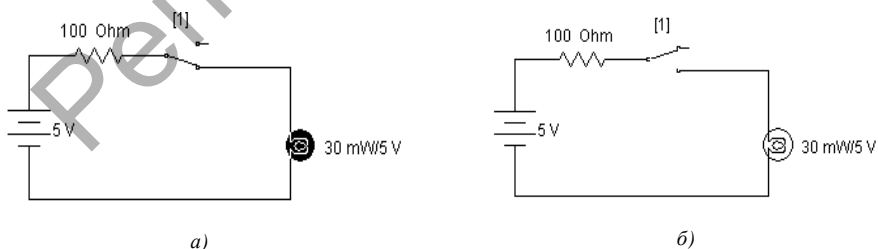


Рисунок 1.1 — Схемы электромеханических имитаторов единицы (а) и нуля (б)

На рисунке 1.1 показаны ключи «1» и «0», управляемые клавишами <1> и <0> соответственно, и вспомогательное устройство в виде батареи 5 В с внутренним сопротивлением 100 Ом и лампа накаливания на 6 В с мощностью 30 мВт, которые позволяют судить о состоянии ключа: если он находится в положении «1», лампа горит (рис. 1.1, а), или не горит, если он — в положении «0» (рис. 1.1, б).

Возможно другое расположение ключей по отношению к вспомогательным устройствам, показанное на рисунке 1.2. В этих схемах состояние индикатора «0» или «1» противоположно показанному на рисунке 1.1. При нажатии на клавишу <1> индикатор фиксирует состояние «0» (рис. 1.2, а) и наоборот (рис. 1.2, б). Следовательно, схемы на рисунке 1.2 по выходному сигналу (состоянию индикаторных лампочек) обратны (инверсны) по отношению к схемам на рисунке 1.1, поэтому такие ключи называют инверторами.

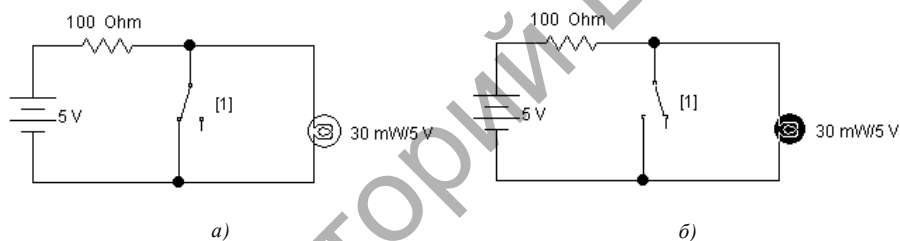


Рисунок 1.2 — Электромеханические имитаторы логической единицы (а) и нуля (б) в инверсном режиме

Поскольку в цифровых системах содержится огромное количество ключей (только в одном микропроцессоре их несколько миллионов) и они не могут сообщать друг другу о своем состоянии миганием лампочек, то для взаимного обмена информацией используются электрические сигналы напряжений. При этом ключи, как правило, применяются в инверсном режиме в соответствии со схемами (рис. 1.3).

На рисунке 1.3 сопротивление 490 Ом имитирует внутреннее сопротивление нагрузки ключа (аналог коллекторного сопротивления в транзисторном ключе), сопротивление 10 Ом — сопротивление замкнутого электронного ключа, сопротивление 500 Ом — сопротивление разомкнутого ключа с учетом внешней нагрузки. Как видно из рисунка 1.3, наличие на выходе логического нуля (инверсия 1) индуцируется

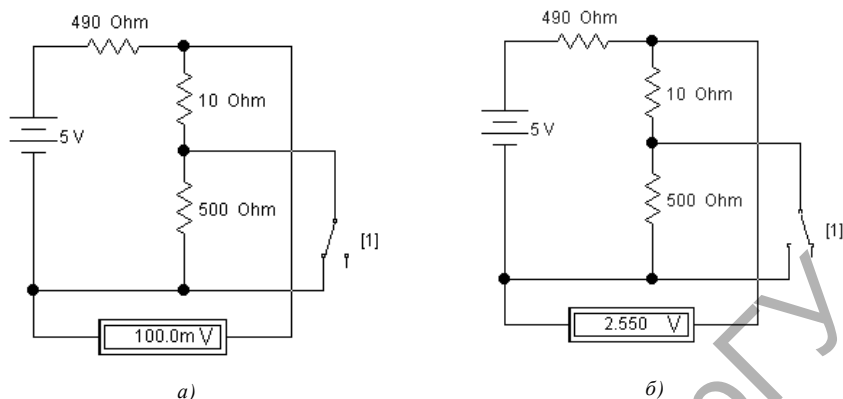


Рисунок 1.3 — Электромеханические имитаторы логической «1» (а) и «0» (б) в инверсном режиме с индикаторами выходного напряжения

напряжением 100 мВ (в практических конструкциях может быть и больше), а наличие логической единицы — напряжением 2,55 В (нормируется на уровне 2,4 В). Электронные ключи проектируются таким образом, чтобы при наихудших сочетаниях входных и выходных параметров ключи могли различать сигналы логической «1» и «0».

В цифровой технике практические аналоги рассмотренных схем принято называть логическими элементами. При этом в зависимости от выполненных функций каждый элемент имеет свое название и соответствующее графическое обозначение. На рисунке 1.4 показаны обозначения базовых логических элементов, принятые в программе EWB.

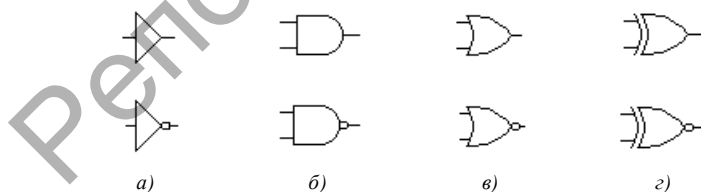


Рисунок 1.4. — Графические обозначения буферного элемента (а), элементов И (AND) (б), ИЛИ (OR) (в), ИСКЛЮЧАЮЩЕЕ ИЛИ (XOR) (г) и их инверсные варианты во втором ряду (NOT, NAND, NOR, XNOR соответственно)

Электромеханическим аналогом буферного элемента являются имитаторы на рисунке 1.3, а логического элемента НЕ (NOT) — на рисунке 1.2 и 1.3. Электромеханические аналоги двухвходовых элементов И и НЕ показаны на рисунке 1.5.

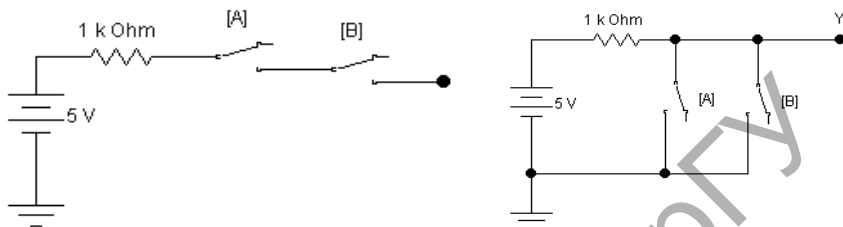


Рисунок 1.5 — Электромеханические имитаторы двухвходовых элементов

При наличии в программе EWB такого инструмента, как логический преобразователь исследования логических схем целесообразно проводить с его помощью. В качестве примера на рисунке 1.6 приведена схема исследования элемента ИСКЛЮЧАЮЩЕЕ ИЛИ.

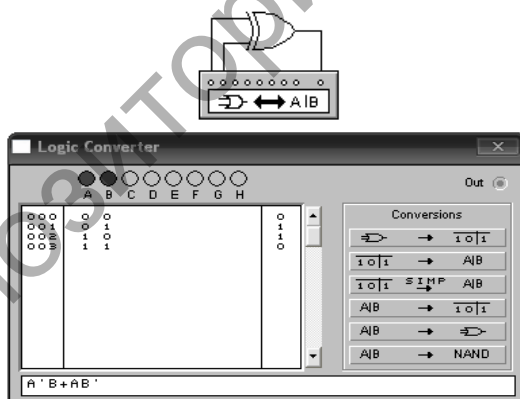


Рисунок 1.6 — Исследование логического элемента ИСКЛЮЧАЮЩЕЕ ИЛИ с помощью логического преобразователя

Подключение исследуемого элемента к логическому преобразователю очевидно из рисунка 1.6. При наличии двух входов возможны только четыре комбинации входных сигналов, что отображается на

экране преобразователя в виде таблицы истинности, которая генерируется после нажатия клавиши  \rightarrow $\bar{1}0\bar{1}$.

Для получения булева выражения исследуемого элемента необходимо нажать клавишу $\bar{1}0\bar{1} \rightarrow A\bar{B}$. Это выражение приводится на дополнительном дисплее, расположенном в нижней части лицевой панели, в виде двух слагаемых, соответствующих выходному сигналу **ИСТИНА** (сигнал логической «1» на выходе **OUT**). Сопоставление полученного выражения с таблицей истинности убеждает нас в том, что таких комбинаций действительно две, если учесть, что в полученном выражении приняты следующие обозначения: $A\bar{}$ — инверсия $A = 1$, \bar{B} — инверсия B , знак «+» соответствует логической операции **ИЛИ**.

С помощью логического преобразователя можно проводить не только анализ логических устройств, но и их синтез. Допустим, что нам требуется составить схему и булево выражение для логического элемента, у которого выходная комбинация в таблице истинности не 0110, как на рисунке 1.6, а 1101. Для внесения необходимых изменений отмечаем курсором в столбце **OUT** подлежащий изменению символ, изменяем его с помощью клавиатуры и затем, перемещаясь по столбцу клавишами управления курсором, изменяем по необходимости символы в других строках. После внесения всех изменений последовательно на-

жимаем на клавиши $\bar{1}0\bar{1} \rightarrow A\bar{B}$, $\bar{1}0\bar{1} \rightarrow \bar{A}B$, $A\bar{B} \rightarrow \bar{1}0\bar{1}$ и получаем результат, представленный на рисунке 1.7, а его булево выражение — на дополнительном дисплее.

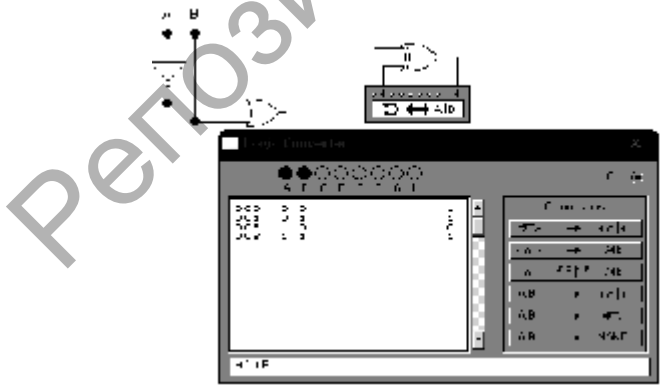


Рисунок 1.7 — Результат синтеза логического устройства по заданной таблице истинности

В более общем случае для выполнения синтеза целесообразно действовать следующим образом. Щелчком курсора по иконке логического преобразователя непосредственно на линейке приборов раскрываем его лицевую панель. Активизируем курсором клеммы кнопки **A**, **B...H** (начиная с **A**), количество которых равно количеству входов синтезируемого устройства. Вносим необходимые изменения в столбец **OUT** и после нажатия на панели преобразователя указанных выше клавиш управления получаем результат в виде схемы на рабочем поле программы и булево выражение в дополнительном дисплее.

В заключение заметим, что для двухвходовых элементов на рисунке 1.4 можно увеличить количество входов до восьми, открывая двойным щелчком по значку компонента диалоговое окно (рис. 1.8). По умолчанию в этом окне указано минимально возможное число входов, равное двум.

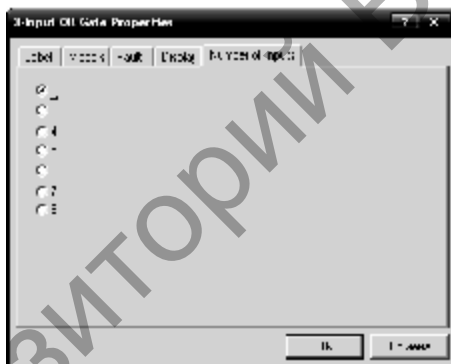


Рисунок 1.8 — Окно установки количества входов логического элемента

Задания

1. Проведите моделирование оставшихся без рассмотрения двухвходовых логических элементов на рисунке 1.4 с использованием логического преобразователя и установите для каждого из них соответствие таблицы истинности и булева выражения.
2. Разработайте схемы электромеханических имитаторов двухвходовых логических элементов на рисунке 1.4 (за исключением элемента И).

3. Проведите синтез трехвходового логического устройства с выходной комбинацией 10011110 в таблице истинности.

4. Составьте схему для реализации выражения $(\bar{A} \cdot C + B \cdot \bar{C})$ с помощью логических элементов. A, B, C — входные сигналы.

Часть 2

Логический элемент с тремя состояниями

Теоретические сведения

За основу логического элемента с тремя состояниями взят базовый элемент серии 134 и к нему добавлен элемент, обеспечивающий возможность реализации режима третьего состояния или так называемого **Z**-состояния. Следует отметить, что рассматриваемый элемент является важным компонентом многих цифровых ИМС, начиная от простейших логических элементов и кончая сложными регистрами и шинными формирователями, обеспечивающими возможность реализации наиболее распространенных архитектур ЭВМ и микропроцессорных систем управления с общей шиной.

Принципиальная схема логического элемента с тремя состояниями представлена на рисунке 1.9. Она содержит базовый логический элемент серии 134 на транзисторах **VT1...VT4**, резисторах **R1...R4** и диоде **VD2**. В базовом элементе в качестве **VT1** используется так называемый мно-гоэмиттерный транзистор, однако ввиду отсутствия такового в библиотеке EWB он представлен обычным транзистором. Ко входу **In** логического элемента подключен имитатор входного сигнала на переключателе **D**, управляемый с клавиатуры клавишей **D**, резистора R_d , имитирующего выходное сопротивление 1 кОм, имитирующего источник входного сигнала в режиме генерации логической единицы. К точке **A** схемы подключены диод **VD1** имитатор источника управления состоянием выхода логического элемента на переключателе **E**, управляемого клавишей **E**. Все элементы дополнительной схемы и компоненты из библиотек **Passive** и **Control**. В исходном состоянии диод **VD1** закрыт напряжением положительной полярности на его катоде и он не оказывает влияния на работу схемы. К эмиттеру транзистора **VT1** подключен резистор R_d , на котором создается падение напряжения

$$U_{in} = \frac{R_d(U_{cc} - U_{be})}{R_1 + R_d},$$

где U_{cc} — напряжение питания (5 В);

U_{be} — напряжение база-эмиттер открытого транзистора (0,7 В).

При $U_{in} \leq 0,4$ В логический элемент воспринимает входной сигнал как сигнал логического «0». При этом напряжение на коллекторе транзистора **VT1** и базе **VT2** недостаточно для открывания последнего. В результате падение напряжения на резисторе **R3** близко к нулю и транзистор **VT4** будет закрыт, а база транзистора **VT3** соединена с источником питания U_{cc} через резистор **R2** и **VT3** будет открыт. При этом выходное напряжение U_y , измеряемое мультиметром, равно

$$U_{in} = \frac{(U_{cc} U_{кбнас} - U_{пр}) R_y}{R_y + R_4},$$

где $U_{кбнас} < 1$ В — напряжение насыщения транзистора **VT4**;

$U_{пр} < 1$ В — падение напряжения на диоде **VD2**;

R_y — сопротивление нагрузки, равное в данном случае входному сопротивлению мультиметра.

Если выбрать $R_y \gg R_4$, то $U_y > 3$ В, т. е. при подаче на вход сигнала логической единицы. Таким образом, схема на рисунке 1.9 является логическим элементом НЕ (инвертором).

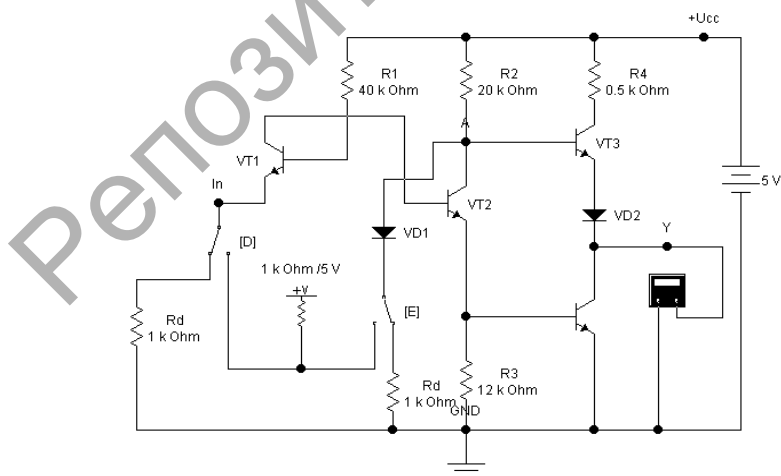


Рисунок 1.9 — Логический элемент с тремя состояниями

При переводе клавишей **D** одноименного переключателя во второе положение напряжение коллектора транзистора **VT2** (точка **A**) будет равно

$$U_a = U_{\text{пр}} + \frac{(U_{cc} - U_{\text{пр}})R_d}{R_d + R_2}.$$

При выполнении неравенства $R_2 \gg R_d$ напряжение $U_{\text{пр}} < 1\text{В}$, что недостаточно для открытия двух переходов (эмиттерного и диодного), и транзистор **VT3** будет закрыт вне зависимости от состояния транзистора **VT2**. Если этот транзистор открыт (наихудший случай), то с учетом его напряжения насыщения, сравнимого с $U_{\text{пр}}$, падения напряжения на резисторе **R3** будет ничтожно малым, следовательно, транзистор **VT4** будет закрыт. Таким образом, выход схемы полностью отключается от нагрузки, что может быть зафиксировано мультиметром в режиме омметра — он будет измерять очень большое сопротивление.

Заметим, что в составе библиотеки компонентов имеется неинвертирующий элемент с тремя состояниями. Он показан на рисунке 1.10, где обозначено: **In**, **Out**, **Enable** — вход, выход и вход сигнала разрешения. При подаче логического «0» на вход разрешения выход буфера переводится в третье состояние.

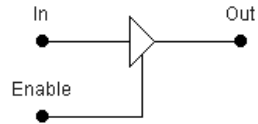


Рисунок 1.10 — Буферный элемент с тремя состояниями

Задание

Проведите моделирование схемы на рисунке 1.9. Изучите все возможные состояния и зафиксируйте показания мультиметра (используйте режимы вольтметра и омметра, в зависимости от положения ключей). Проанализируйте полученные результаты.

Контрольные вопросы

1. Известно, что единицей измерения информации является бит. Какие значения может принимать эта единица?
2. Какая разница между элементами И и И—НЕ, ИЛИ и ИЛИ—НЕ?
3. Напишите логические выражения и составьте таблицы истинности для элементов ЗИ и ЗИЛИ—НЕ.

4. Как использовать в качестве схемы НЕ (инвертора) элементы 2И—НЕ, 2ИЛИ—НЕ?
5. Установите различия в булевых выражениях и графических обозначениях логических элементов программы EWB от принятых в отечественной научно-технической литературе.
6. Какими замечательными свойствами обладает логический элемент с тремя состояниями, какие задачи цифровой техники он позволил решить?
7. Как включить режим, в котором выход полностью отключается от нагрузки? Пояснить, что происходит в схеме на рисунке 1.9 в этом случае.

Репозиторий Баргу

ЛАБОРАТОРНАЯ РАБОТА 2

ЦИФРОВЫЕ КОМПАРАТОРЫ.

УСТРОЙСТВО КОНТРОЛЯ ЧЕТНОСТИ

Цели:

- исследовать цифровые компараторы, выяснить их принцип действия;
- познакомиться с устройством контроля четности, провести моделирование простейшего устройства контроля четности с помощью логических элементов, исследовать его работу.

Часть 1

Цифровые компараторы

Теоретические сведения

Цифровые компараторы (от англ. *compare* — сравнивать) выполняют сравнение двух чисел A и B одинаковой разрядности, заданных в двоичном или двоично-десятичном коде. В зависимости от схемного исполнения компараторы могут определять равенство $A = B$ или неравенства $A < B$, $A > B$. Результат сравнения отображается в виде логического сигнала на одноименных выходах.

Цифровые компараторы применяются для выявления нужного числа (слова) в цифровых последовательностях, отметки времени в часовых приборах, выполнения условных переходов в вычислительных устройствах, а также адресных селектора [4].

Схема одноразрядного компаратора приведена на рисунке 2.1. Компаратор состоит из двух элементов НЕ, четырех элементов И и одного элемента ИЛИ—НЕ.

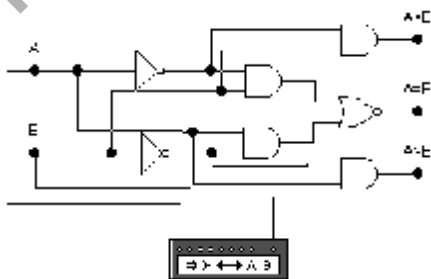


Рисунок 2.1 — Схема одноразрядного цифрового компаратора

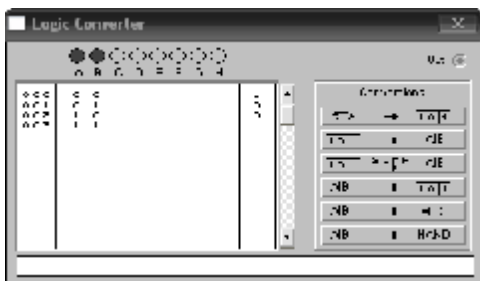


Рисунок 2.2 — Результаты моделирования цифрового компаратора

Для исследования компаратора к нему подключен логический преобразователь. Подсоединяя его клемму **OUT** к каждому выходу компаратора, можно получить таблицу истинности и булево выражение для каждого режима работы компаратора. Для случая $A = B$, показанного на рисунке 2.1, результаты моделирования пред-

ставлены на рисунке 2.2, откуда видно, что условию $A = B$ соответствуют комбинации сигналов на входе: $A = B = 1$ или $A = B = 0$. Этому условию и отвечает булево выражение на дополнительном дисплее.

Задание

Используя схему на рисунке 2.1, проведите ее исследование в режимах $A > B$, $A < B$ и проанализируйте полученные результаты.

Часть 2

Устройство контроля четности

Теоретические сведения

Операция контроля четности двоичных чисел позволяет повысить надежность передачи и обработки информации. Ее сущность заключается в суммировании по модулю 2 всех разрядов с целью выяснения четности числа, что позволяет выявить наиболее вероятную ошибку в одном из разрядов двоичной последовательности. Например, если при передаче кода 1001 произойдет сбой во втором разряде, то на приемном пункте получим код 1101 — такую ошибку определить в общем случае затруднительно. Если же код относится к двоично-десятичному (8-4-2-1), то ошибку легко обнаружить, поскольку полученный код (десятичный эквивалент — число 13) не может в принципе принадлежать к двоично-десятичному.

Обнаружение ошибок путем введения дополнительного бита четности происходит следующим образом. На передающей стороне передаваемый код анализируется и дополняется контрольным битом до четного или нечетного числа единиц в суммарном коде. Соответственно суммарный код называется четным или нечетным. В случае нечетного кода дополнительный бит формируется таким образом, чтобы сумма всех единиц в передаваемом коде, включая контрольный бит, была нечетной. При контроле четности все наоборот. Например, в числе 0111 число единиц нечетно, поэтому при контроле нечетности дополнительный бит должен быть «0», а при контроле четности — «1». На практике чаще всего используется **контроль нечетности**, поскольку он позволяет фиксировать полное пропадание информации (случай нулевого кода во всех информационных разрядах). На приемной стороне производится проверка кода четности. Если он правильный, то прием разрешается, в противном случае включается сигнализация ошибки или посылается передатчику запрос на повторную передачу.

Схема формирования бита четности [1] для четырехразрядного кода показана на рисунке 2.3. Она содержит четыре элемента ИСКЛЮЧАЮЩЕЕ ИЛИ, выполняющих функции сумматоров по модулю 2 (без переноса) и состоит из трех ступеней. На первой ступени попарно суммируются все биты исходного кода на входах **A, B, C, D**. На второй ступени анализируются сигналы первой ступени и устанавливается четность или нечетность суммы входного кода. На третьей ступени полученный результат сравнивается с контрольным сигналом на входе **E**, задающим вид используемого контроля, в результате чего на выходе **F** формируется пятый дополнительный бит четности, сопровождающий информационный сигнал в канале передачи.

Результаты моделирования формирователя показаны на рисунке 2.3 в виде таблицы истинности из 32 возможных двоичных комбинаций и булева выражения (из 32 комбинаций на рисунке 2.4 видны только первые 16, остальные просматриваются с помощью линейной прокрутки). Для просмотра

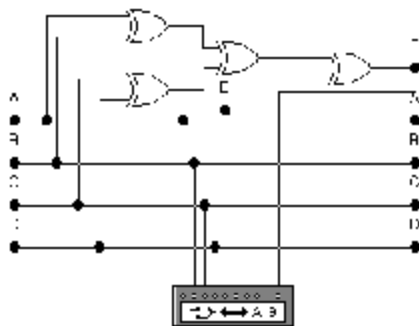


Рисунок 2.3 — Схема формирования бита четности четырехразрядного кода

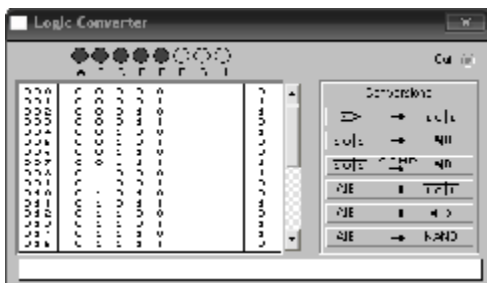


Рисунок 2.4 — Результаты моделирования схемы

составляющих булева выражения необходимо мышью поместить курсор в дополнительный дисплей и передвигать ее клавишами управления курсором.

В библиотеке программы EWB схема проверки на четность и нечетность представлена ИМС 74280, схема ее включения показана на рисунке 2.5. ИМС 74280 имеет 9 входов (A, B...I)

и два выхода (EVEN, ODD), один из которых инверсный. Вход I используется для управления видом контроля («0» — контроль четности, «1» — контроль нечетности) и управляется переключателем Z (управляется с клавиатуры одноименной клавишей). Вывод NC — *not connection* — пустой, т. е. внутри ИМС к нему ничего не подключено.

Правильность функционирования на рисунке 2.5 проверяется с помощью генератора слова, при этом тип контроля (четности или нечетности) выбирается переключателем Z; на входы рассматриваемого устройства подаются различные двоичные комбинации; состояние выходов ИМС контролируется подключенными к ним светоиндикаторами (логическими пробниками).

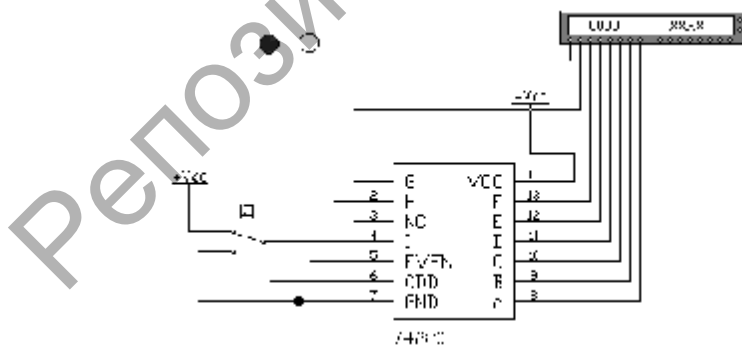


Рисунок 2.5 — Схема включения ИМС 74280

Задания

1. Исследуйте схему на рисунке 2.3.
2. Проверьте правильность функционирования схемы на рисунке 2.5, подавая на входы двоичные комбинации с генератора слова.
3. Выясните, каким образом с помощью схемы на рисунке 2.3 можно добавлять контрольный бит (в случае использования контроля четности, контроля нечетности).

Контрольные вопросы

1. Какие функции выполняет цифровой компаратор, в каких устройствах он может быть использован?
2. Приведите таблицу переключений и схему одноразрядного компаратора.
3. Какое назначение имеют формирователи кода четности, где они могут быть использованы?
4. Какая форма контроля четности чаще всего используется на практике, в частности, в вашем компьютере, если в нем установлены модули оперативного запоминающего устройства (ОЗУ) с нечетным числом микросхем?
5. Из представленных на экране логического преобразователя данных (см. рис. 2.4) выберите комбинации, относящиеся к контролю четности и нечетности, а также соответствующие им слагаемые булева выражения.

ЛАБОРАТОРНАЯ РАБОТА 3

МУЛЬТИПЛЕКСОРЫ И ДЕМУЛЬТИПЛЕКСОРЫ

Цели:

- познакомиться с мультиплексорами и демультимплексорами;
- провести их исследование в программе EWB;
- научиться составлять многоуровневые схемы мультиплексоров и демультимплексоров на основе логических элементов.

Теоретические сведения

Назначение мультиплексоров (от англ. *multiplex* — многократный) — коммутировать в заданном порядке сигналы, поступающие с нескольких входных шин на одну выходную. У мультиплексора может быть, например, 16 входов и 1 выход. Это означает, что если к этим входам присоединены 16 источников цифровых сигналов (генераторов последовательных цифровых слов), то байты от любого из них можно передавать на единственный выход. Для выбора любого из 16 каналов необходимо иметь 4 входа селекции ($2^4 = 16$), на которые подается двоичный адрес канала. Так, для передачи данных от канала номер 9 на входах селекции необходимо установить код 1001. В силу этого мультиплексоры часто называют селекторами или селекторами-мультиплексорами.

Мультиплексоры применяются, например, в МП 18088 для выдачи на одни и те же выходы МП адреса и данных, что позволяет существенно сократить общее количество выводов микросхемы. В микропроцессорных системах управления мультиплексоры устанавливаются на удаленных объектах для возможности передачи информации по одной линии от нескольких установленных на них датчиков.

На рисунке 3.1 приведена схема двухканального мультиплексора, состоящего из элементов ИЛИ, НЕ и двух элементов И.

Результаты моделирования двухканального мультиплексора с помощью логического преобразователя показаны на рисунке 3.2, из которого видно, что его выходной сигнал описывается булевым выражением $OUT = BC + AC$, т. е. сигнал из канала А проходит на выход при адресном входе $C = 0$, а из канала И — при $C = 1$, что и соответствует логике работы мультиплексора.

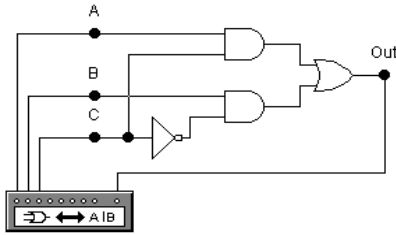


Рисунок 3.1 — Схема двухканального мультиплексора

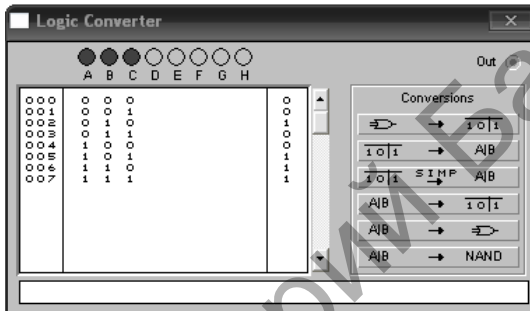


Рисунок 3.2 — Результаты моделирования двухканального мультиплексора

На рисунке 3.3 приведена схема четырехканального мультиплексора. Для исследования выходного сигнала использовался светодиодный индикатор, так как это наиболее простой элемент, который не ухудшает наглядности в случае изучения одного выходного сигнала. Для изучения того, каким именно образом происходит коммутация входных каналов на один выход, подаем единицы на входы **C2** и **C3**, и нули на входы **C0** и **C1**. Далее с помощью генератора слова изменяются значения адресных входов **A** и **B**, последовательно **00**, **10**, **01**, **11**. В результате, варьируя значениями входов **C0...C3**, можно отметить те варианты входов, при которых загорается либо потухает индикатор. Таким образом устанавливаем, с какого именно информационного входа сигнал идет на выход.

Демультимплексоры в функциональном отношении противоположны мультиплексорам. С их помощью сигналы с одного информационного входа распределяются в требуемой последовательности по нескольким выходам.

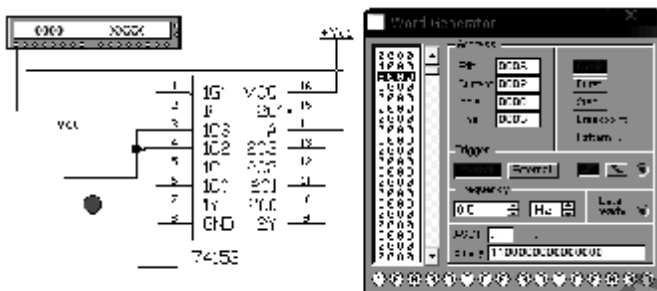


Рисунок 3.3 — Схема исследования сдвоенного четырехканального мультиплексора

Выбор нужной выходной шины, как и в мультиплексоре, обеспечивается установкой соответствующего кода на адресных входах. При m адресных входах демультиплексор может иметь до 2^m выходов.

Принцип работы демультиплексора поясним с помощью схемы на рисунке 3.4, на котором обозначено: X — информационный вход, A — вход адреса, Y_0 , Y_1 — выходы. Схема содержит два элемента И и один элемент НЕ. Из рисунка 3.4 нетрудно увидеть, что при $A = 0$ сигнал информационного входа передается на выход Y_0 , а при $A = 1$ — на выход Y_1 . Следует отметить, что промышленностью демультиплексоры как таковые не выпускаются, поскольку режим демультиплексора может быть реализован как частный случай в других устройствах — дешифраторах.

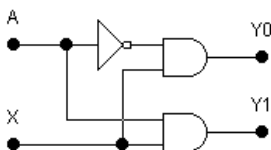


Рисунок 3.4 — Схема демультиплексора

Исследование сдвоенного демультиплексора на 4 выхода ИМС 74139 (рис. 3.5) производится с помощью логического анализатора, который позволяет более наглядно получать данные о выходных сигналах.

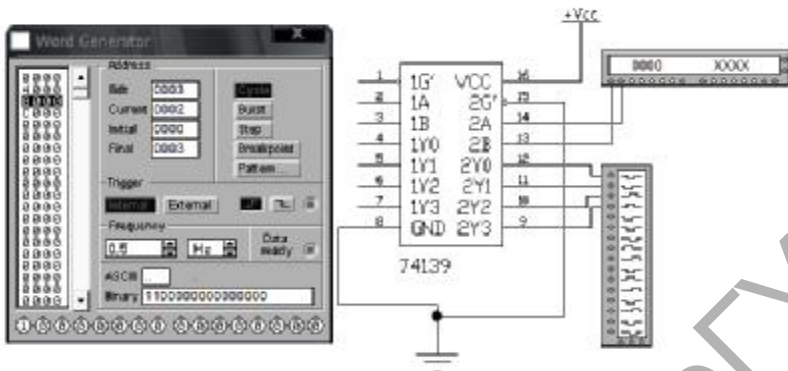


Рисунок 3.5 — Схема для исследования сдвоенного демультиплексора (ИМС 74139) на 4 выхода

Задания

1. Соберите схему мультиплексора на 2 входа (см. рис. 3.1), проанализируйте его работу.
2. Используя методику анализа мультиплексора с помощью логического преобразователя, исследуйте внутреннюю структуру сдвоенного четырехканального мультиплексора 74153. Из сопоставления обозначений выводов этой ИМС и ее отечественного аналога К155КП2 следует, что их функциональное назначение таково: **A, B** — адресные входы, **1G, 2G** — инверсные входы разрешения первого и второго мультиплексоров, **1C0...1C3** и **2C0...2C3**, **1Y** и **2Y** — входы и выходы первого и второго мультиплексоров соответственно.
3. Проведите исследование демультиплексора на 2 выхода (см. рис. 3.4) с помощью логического конвертера (исследуйте отдельно каждый выход).
4. Исследуйте принцип работы сдвоенного демультиплексора на 4 выхода, наблюдая выходные сигналы с помощью логического анализатора (LOGIC ANALYZER), меняйте адресный код (можно использовать генератор слова). Проверив работу схемы, опишите результаты эксперимента, приведите временные диаграммы изменения выходных сигналов (см. рис. 3.5).

Контрольные вопросы

1. Что такое мультиплексор, каково его назначение?
2. Какая закономерность количества входов и выходов мультиплексора? Сколько должно быть входов и выходов, если передаются 7-разрядные данные? 24-разрядные?
3. Составьте схему мультиплексора на 4 входа с использованием только логических элементов.
4. Сколько понадобится 4-входовых мультиплексоров для составления 16-входового мультиплексора? Как будет выглядеть его схема?
5. Что такое демультимплексор, для решения каких задач его можно применить?
6. Каков принцип работы демультимплексора? Пояснить на схеме рисунок 3.3.
7. Чем будет отличаться демультимплексор на 3 выхода, составленный из логических элементов, от демультимплексора на рисунке 3.3?

Репозиторий БарГУ

ЛАБОРАТОРНАЯ РАБОТА 4 ШИФРАТОРЫ И ДЕШИФРАТОРЫ

Цели:

- познакомиться с шифраторами и дешифраторами;
- провести их исследование в программе EWB.

Теоретические сведения

Шифраторы (кодеры) используются для преобразования десятичных чисел в двоичный или двоично-десятичный код, например, в микрокалькуляторах, в которых нажатие десятичной клавиши соответствует генерация соответствующего двоичного кода. Поскольку возможно нажатие сразу нескольких клавиш, в шифраторах используется принцип приоритета старшего разряда, т. е. при нажатии клавиш <9>, <5> и <2> на выходе шифратора будет код 1001, соответствующий цифре «9». Следует отметить, что шифраторы как отдельный класс функционирования устройств представлены в наиболее богатой ТТЛ — серии всего двумя ИМС — 74147 и 74148, причем последняя ИМС имеется в библиотеке EWB. Ее схема включения показана на рисунке 4.1.

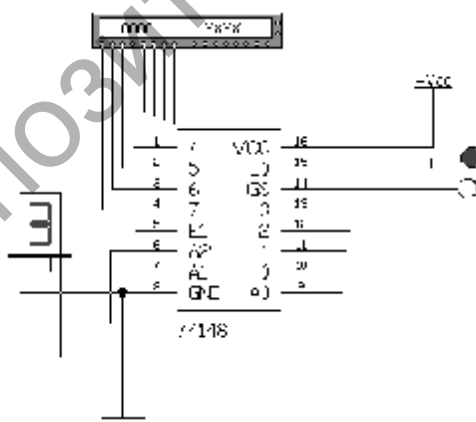


Рисунок 4.1 — Схема включения ИМС 74148

Назначение выводов ИМС 74148:0...7 — входы: **A0, A1, A2** — выходы: **E1** — вход разрешения, **E0, GS** — выходы для каскадирования дешифратора. При моделировании необходимо обратить внимание на реализацию принципа приоритета, при этом следует учесть, что все входы и выходы инверсные (на функциональной схеме ИМС в программе EWB они ошибочно показаны прямыми).

Режим работы используемого в схеме на рисунке 4.1 генератора слова показан на рисунке 4.2. **Дешифратор** (декодер) — устройство с несколькими входами и выходами, у которого определенным комбинациям входных сигналов соответствует активное состояние одного из выходов, т. е. дешифратор является обращенным по входам демультиплексором, у которого адресные входы стали информационными, а бывший информационный вход стал входом разрешения. Поэтому дешифраторы часто называют дешифраторами-демультиплексорами и наоборот.

Дешифраторы и демультиплексоры в виде серийных ИМС средней степени интеграции широко используется в информационно-измерительной технике и микропроцессорных системах управления, в частности, в качестве коммутаторов-распределителей информационных сигналов и синхроимпульсов, для демультиплексирования данных и адресной логики в запоминающих устройствах, а также для преобразования двоично-десятичного кода в десятичный с целью управления индикаторами и печатающими устройствами.

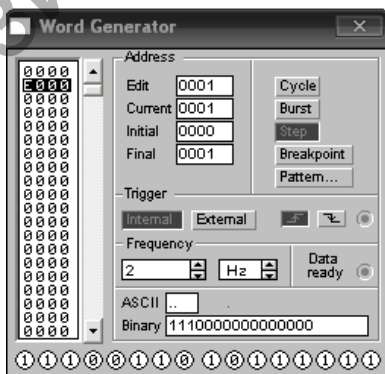


Рисунок 4.2 — Панель генератора слова с установками для схемы на рисунке 4.1

Дешифраторы как самостоятельные изделия электронной техники имеют 4, 8 или 16 выходов. Если требуется большее число выходов, дешифраторы наращиваются в систему.

В качестве примера на рисунке 4.3 приведена схема включения дешифратора 74154. ИМС 74154 имеет четыре адресных входа **A, B, C, D**, два входа разрешения **G1, G2** и шестнадцать выходов **0...15** (выходы не прямые, как ошибочно обозначено в EWB, а инверсные, т. е. в исходном состоянии на выходах сигнал логической «1»). В режиме дешифратора с генератора слова на входы **G1, G2** подается «0», а на адресные входы — код в диапазоне 0000...1111.

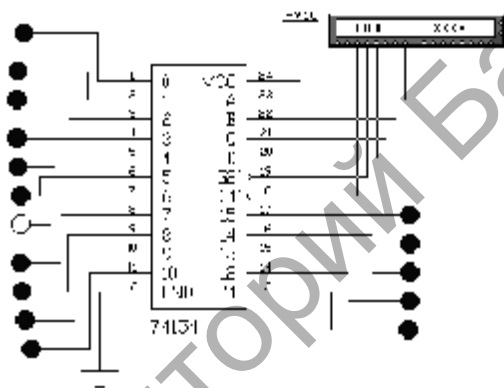


Рисунок 4.3 — Схема включения дешифратора 74154

В режиме демультиплексора один из разрешающих входов, например **G1**, используется в качестве информационного. Информационный сигнал в виде логического «0» с этого выхода распределяется по выходам **0...15** в соответствии с состоянием адресных входов, т. е. режимы дешифратора и демультиплексора практически неразличимы.

Задания

1. Меняя кодовые комбинации в генераторе слова схемы на рисунке 4.1, покажите, в чем заключается смысл слова «приоритетный» в названии шифратора типа 74148.

2. Подготовьте выходные кодовые комбинации генератора слов в схеме на рисунке 4.3, обеспечивающие поочередное включение светоиндикаторов на выходе дешифратора, начиная с выхода **0**. Составьте таблицу истинности, сравните ее с библиотечной.

3. Переведите дешифратор в схеме 4.3 в режим демультиплексирования и выполните задание 2 (чтобы перевести дешифратор в режим демультиплексора, используйте вход **G2** в качестве информационного). Сопоставьте полученные данные с результатами предыдущей лабораторной работы.

Контрольные вопросы

1. Что такое шифратор, при решении каких задач он используется?
2. Каким образом в шифраторах реализован принцип приоритета старшего разряда?
3. При решении каких задач цифровой техники используется дешифратор?
4. Каков принцип работы дешифратора, чем он схож с демультиплексором?
5. Сколько корпусов ИМС 74148 потребуется для построения шифратора на 16 входов? На 32 входа? Приведите схему на 16 входов.
6. Как использовать в режиме дешифратора на 8 выходов ИМС 74155? Составьте принципиальную схему.

ЛАБОРАТОРНАЯ РАБОТА 5

АРИФМЕТИЧЕСКИЕ СУММАТОРЫ.

АРИФМЕТИКО-ЛОГИЧЕСКОЕ УСТРОЙСТВО

Цели:

- смоделировать арифметические сумматоры;
- изучить различные операции, и способы их включения, арифметико-логического устройства.

Часть 1

Арифметические сумматоры

Теоретические сведения

Арифметические сумматоры являются основной частью арифметико-логических устройств (АЛУ) микропроцессоров (МП). Они используются для формирования физического адреса ячеек памяти в МП с сегментной организацией памяти. В программе EWB арифметические сумматоры представлены в библиотеке **Comb1** двумя базовыми устройствами, показанными на рисунке 5.1: полусумматором и полным сумматором. Они имеют следующие названия выводов: **A**, **B** — входы слагаемых, Σ — результат суммирования, C_0 — выход переноса, C_i — вход переноса. Многоразрядный сумматор создается на базе одного полусумматора и n полных сумматоров. В качестве примера на рисунке 5.2 приведена структура трехразрядного сумматора [6]. На входы **A1**, **A2**, **A3** и **B1**, **B2**, **B3** подаются первое и второе слагаемые соответственно, а с выходов **S1**, **S2**, **S3** снимается результат суммирования.



а)



б)

Рисунок 5.1 — Схемы полусумматора (а) и полного сумматора (б)

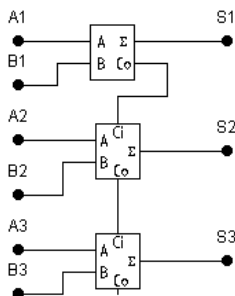
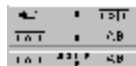


Рисунок 5.2 — Трехразрядный сумматор

Для исследования внутренней структуры и логики функционирования сумматора как нельзя лучше подходит логический преобразователь. После подключения полусумматора к преобразователю согласно



рисунку 5.3, а, последовательно нажимаем кнопки \rightarrow и в результате получаем таблицу истинности и булево выражение. Сравнивая полученные результаты с результатами исследования базовых логических элементов в предыдущем разделе, приходим к выводу, что при подключении вывода (полусумматора к зажиму **OUT** преобразователя (как показано на рисунке 5.4, а)) он выполняет функции элемента ИСКЛЮЧАЮЩЕЕ ИЛИ. Подключив клемму **OUT** преобразователя к выходу C_0 полусумматора и проделав аналогичные действия, приходим к выводу, что в таком включении полусумматор выполняет функции элемента И. Следовательно, эквивалентная схема полусумматора имеет вид, показанный на рисунке 5.5.

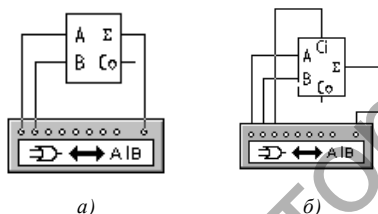


Рисунок 5.4 — Схемы подключения полусумматора (а) и полного сумматора (б) к логическому преобразователю

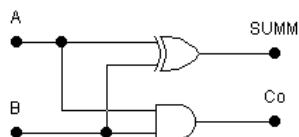


Рисунок 5.5 — Структура библиотечного полусумматора

В каталоге программы EWB 5.2 имеется схема включения четырехразрядного АЛУ (файл alu181.ca4) на базе серийной микросхемы 74181. В несколько переработанном виде она показана на рисунке 5.6. ИМС 74181 обеспечивает 32 режима работы АЛУ в зависимости от состояния управляющих сигналов на входах **M**, **S0...S3**, а также допускает наращивание разрядности (вход **CN** и выход **CN < 4** для переносов). Показанная на рисунке 5.6 схема включения ИМС соответствует режиму сумматора без переноса. Значения четырехразрядных операндов **A** и **B** на входе задаются с помощью генератора слова и в шестнадцатеричном коде отображаются одноименными алфавитно-цифровыми индикаторами.

На выходах **F0...F3** результат суммирования отображается индикатором **F**. Изменяя состояния сигналов на управляющих входах, можно промоделировать большинство функций АЛУ, используемых в микропроцессорах. Режимы работы генератора слова в схеме рисунок 5.6 и его кодовый набор показаны на рисунке 5.7.

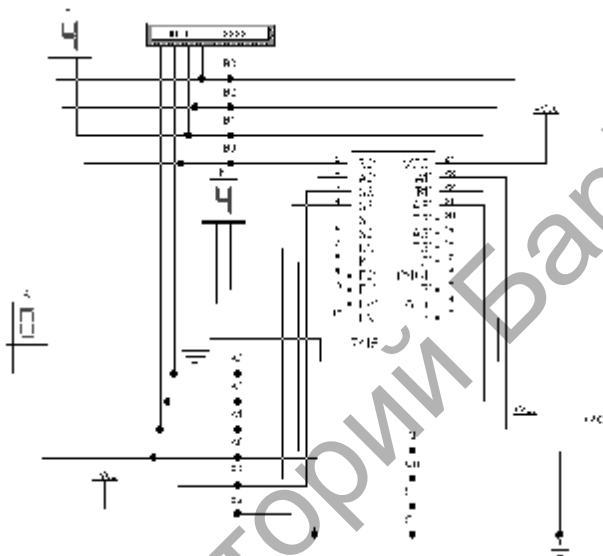


Рисунок 5.6 — Схема включения ИМС 74181 в режиме сумматора без переноса

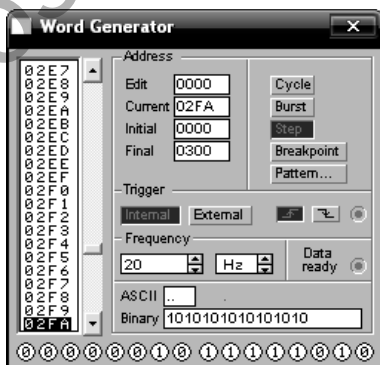


Рисунок 5.7 — Настройка генератора слова для схемы на рисунке 5.6

Задания

1. Соберите схему для исследования полусумматора (см. рис. 5.4, а). Составьте таблицу истинности для его выходов, подключая логический конвертер (Logic Converter) отдельно к каждому выходу.

2. Выясните внутреннюю структуру полного сумматора, пользуясь схемой его подключения к логическому преобразователю на рисунке 5.4, б, и принимая во внимание методику решения аналогичной задачи для полусумматора.

3. Используя опыт работы со схемой на рисунке 5.6, подключите ко входам 3-разрядного сумматора на рисунке 5.3 генератор слова, а к выходам — алфавитно-цифровой индикатор с дешифратором и проверьте правильность его функционирования.

4. Проверьте работу ИМС 74181 в режиме сумматора с переносом (на вход **Cn** подайте сигнал логического «0»). Проведите аналогию между работой АЛУ в режиме сумматора и схемой 3-разрядного сумматора на рисунке 5.3.

Часть 2

Арифметико-логическое устройство

Теоретические сведения

Показанная на рисунке 5.8 схема на базе ИМС 74181 (К155ИПЗ) позволяет оперативно реализовать многие режимы. Возможные режимы задаются с помощью переключателей **0**, **1**, **2**, **3** для подачи сигналов «0» («земля») или «1» (+5 В) на входы управления **S0**, **S1**, **S2**, **S3**. В положении переключателя **M**, показанном на рисунке 13.1 (сигнал «0» на входе **M**), выполняются 16 арифметических операций (16 комбинаций сигналов **S0...S3**) с учетом переноса по входу **Cn** (переключатель **C** в показанном на рисунке 5.8 положении) или без учета переноса (сигнал «0» на входе **Cn** переключателя **C**). При переводе ключа **M** в другое положение (на входе **M** сигнал «1») выполняются 16 логических операций, задаваемых теми же переключателями **0...3**.

Значения четырехразрядных операндов **A** и **B** задаются с помощью генератора слова и в шестнадцатеричном коде отображаются на алфавитно-цифровых индикаторах. На выходах результат суммирования отображается индикатором **F**. При коде 1111 на этих выходах и при

равенстве операндов $A = B$ переводится в единичное состояние. Поскольку этот выход представляет собой каскад с открытым коллектором, то на него подается питание +5 В через резистор 1 кОм. Выход $A = B$ совместно с выходом переноса $Cn + 4$ и выходом P подтверждения переноса используются для формирования признаков $A > B$ и $A < B$ с помощью элементов $U1, U2, U3$.

Изменяя состояния сигналов на управляющих входах, можно промоделировать большинство функций АЛУ, используемых в микропроцессорах. Приведем перечень этих функций:

1. Логические функции (на входе M сигнал «1»), которые выполняются поразрядно, переносы не учитываются.

Код 0000 на входах $S1, S2, S3, S0$: при этом выполняется логическая функция $A \ll n$ — данные со входов A передаются на выходы F с инверсией; может быть использована в команде CMA (здесь и далее используется мнемоника команд микропроцессоров семейства 80xx фирмы Intel).

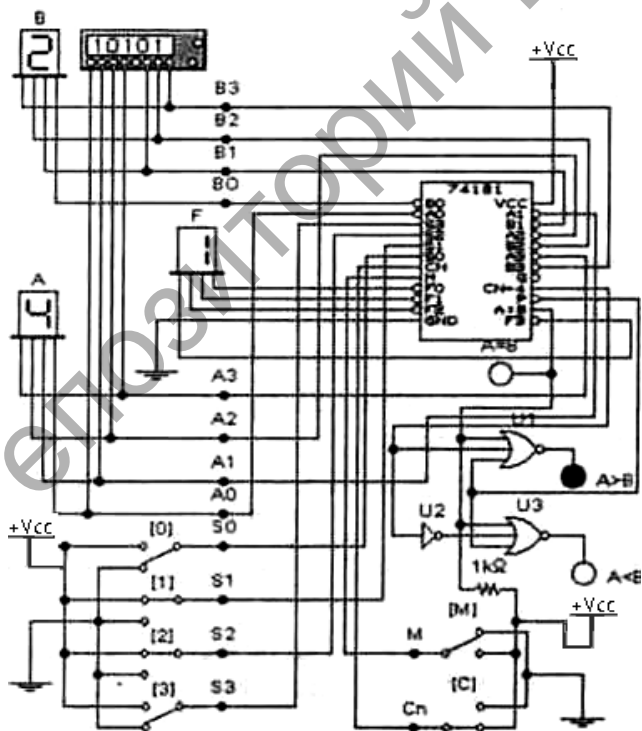


Рисунок 5.8 — АЛУ на ИМС 74181

0001 — $(A+B)'$ — поразрядная операция ИЛИ с инверсией над операндами A и B;

0010 — $A'B$ — операция И инвертированного операнда A и операнда B;

0011 — 0 — нет операции;

0100 — $(AB)'$ — операция И с инверсией;

0101 — B' — инверсия операнда B;

0110 — $A \oplus B$ — операция ИСКЛЮЧАЮЩЕЕ ИЛИ, команда XRA;

0111 — AB' — операция И над операндами A и инверсией B;

1000 — $A'+B$ — операция ИЛИ над инверсией A и операндом B;

1001 — $(A+B)'$ — операция ИЛИ с инверсией;

1010 — B — передача на выход операнда B;

1011 — AB — операция И, команда ANA;

1100 — 1;

1101 — $A + B'$ — операция ИЛИ над инверсией B и операндом A;

1110 — $A + B$ — операция ИЛИ, команда ORA;

1111 — A — передача на выход операнда A.

2. Арифметические операции ($M = 0$) без переноса ($C_n = 1$) и с переносом ($C_n = 0$, данные приводятся в круглых скобках):

0000 — A — передача на выход операнда ($A+1$ — суммирование операнда с 1 переноса, команда инкремента);

0001 — $A + B$ — операция суммирования без учета переноса, команда **ADD** ($(A + B) + 1$ — суммирование с учетом переноса, команда **ADC**);

0010 — $A + B'$ — операция суммирования операнда A с инверсией операнда B без учета переноса ($(A + B') + 1$ — то же, но с учетом переноса);

0011 — -1 (0);

0100 — $A + AB'$ ($A + (AB)'$ + 1).

0101 — $(A + B) + AB'$ ($(A + B) + AB' + 1$);

0110 — $A - B - 1$, команда **SBB** ($A - B$, команда **SUB**);

0111 — $AB' - 1$ ($(AB)'$);

1000 — $A + AB$ ($A + B + 1$);

1001 — $A + B$, команда **ADD** ($A + B + 1$);

1010 — $(A + B') + AB$ ($(A + B') + AB + 1$);

1011 — $AB - 1$ (AB);

1100 — $A + A$ ($A + A + 1$);

1101 — $(A + B) + A$ ($(A + B) + A + 1$);

1110 — $(A + B') + A$ ($(A + B') + A + 1$);

1111 — $A - 1$ (A).

Задание

Проведите моделирование всех перечисленных выше режимов АЛУ (см. рисунок 5.8), предварительно составив неповторяющиеся комбинации на выходе генератора слова. Проверить, как осуществляется переключение АЛУ в режим выполнения логических и арифметических операций, как работает перенос для каждого из этих режимов. Проведите исследование выходов, используемых для сравнения чисел (режим компаратора).

Контрольные вопросы

1. Что такое полусумматор? Приведите его таблицу истинности, схему и логические выражения для выходных сигналов.
2. Чем отличается полусумматор от полного сумматора?
3. Что нужно сделать с сумматором, чтобы он работал аналогично элементу ИСКЛЮЧАЮЩЕЕ ИЛИ?
4. Зачем используется перенос в полусумматорах, сумматорах и АЛУ?
5. Как можно выполнить вычитание с помощью сумматора?
6. Дополните операции без комментариев описанием выполняемых ими функций.
7. Установите значения $CN + 4$ и $A = B$ сигналов для различных соотношений между сравниваемыми числами ($M = 0$, $CN = 1$, $S = 0110$) при работе АЛУ в режиме компаратора.

ЛАБОРАТОРНАЯ РАБОТА 6. ТРИГГЕРНЫЕ СХЕМЫ

Цель: изучить принципы построения и работы триггерных схем на основе RS, JK и D-триггеров.

Теоретические сведения

Триггерные схемы в программе EWB представлены в библиотеке **Seg`I** тремя типами триггеров **RS**, **JK** и **D**, показанными на рисунке 6.1. Назначение выводов триггеров следующее. Для всех триггеров выходы **Q** — прямой, **Q'** — инверсный. Для **RS**-триггера **R** — установка триггера в «0», при сигнале «1» на этом входе **Q = 0**, **Q' = 1**; **S** — установка в «1», при сигнале «1» на этом входе **Q = 1**, **Q' = 0**; комбинация **R = 1**, **S = 1** не изменяет состояния выходов и относится к запрещенным. Для **JK**-триггера **J**, **K** — информационные входы, «>» — тактовый вход; **вывод сверху** — асинхронная предустановка триггера в единичное состояние (**Q = 1**) вне зависимости от состояния сигналов на входах (функционально аналогичен входу **S** **RS**-триггера); **вывод внизу** — асинхронная предустановка триггера в нулевое состояние (так называемая очистка триггера, после которой **Q' = 1**); наличие кружочков на изображениях выводов обозначает, что активными являются сигналы низкого уровня. Для **D**-триггера вход **D** — информационный, состояние этого входа после подачи тактового импульса запоминается триггером, т. е. при **D = 1** имеем **Q = 1**, при **D = 0** — **Q = 0**.

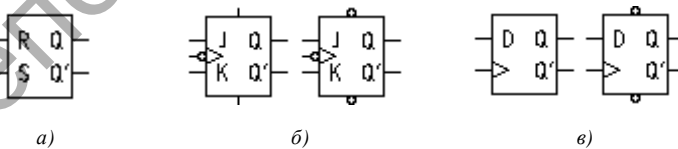


Рисунок 6.1 — Обозначения **RS**-триггеров (а), **JK**-триггеров (б), и **D**-триггеров (в)

Для проведения исследования триггерных схем нельзя использовать логический преобразователь, поскольку триггер является элементом памяти. Для этого необходимо подключить к его входам генератор слова

и светодиодные индикаторы к выходам. Схема для исследования наиболее сложного **JK**-триггера показана на рисунке 6.2. Заметим, что тактовый вход триггера необходимо соединить с выходом синхронизации генератора.

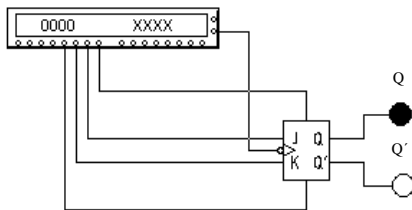


Рисунок 6.2 — Схема для исследования **JK**-триггера

Конечной задачей исследования является получение таблицы истинности, являющейся одной из основных характеристик триггера. Получение ее целесообразно проводить в следующем порядке:

1) последовательно подать на входы предустановки триггера активные сигналы высокого уровня (сигнал «1») и зафиксировать состояние выхода триггера для случая предустановки в «0» и «1». Для проверки асинхронности этих входов указанные операции повторить при различных состояниях сигналов на тактовом и информационных входах. В дальнейшем на эти входы подавать только сигналы логического «0»;

2) подать на тактовый и информационные входы сигналы «0» и «1» в различных комбинациях (на входах асинхронной предустановки при этом должны быть сигналы логического «0» для всех комбинаций) и зафиксировать для каждой комбинации состояние выходов триггера;

3) на основании полученных результатов составить таблицу истинности.

Для понимания процессов, происходящих в триггерах, приведем схему тактируемого **RS**-триггера, показанную на рисунке 6.3. Собственно **RS**-триггер выполнен на двух элемента 2И—НЕ U_3, U_4 . Установка триггера в «0» или «1» возможна только при наличии разрешающего тактового импульса на тактовом входе **C**, т. е. такой триггер полностью идентичен **JK**-триггеру без цепей предустановки и дополнительных цепей обратной связи. Для введения таких цепей достаточно разорвать соединения в точках **S1, R1** и ввести в разрыв элементы 2ИЛИ и элементы U_1, U_2 заменить на трехвходовые. Если же вход **S** соединить со входом **R** через элемент НЕ, то получится **D**-триггер, в котором **S**-вход будет выполнять роль **D**-входа.

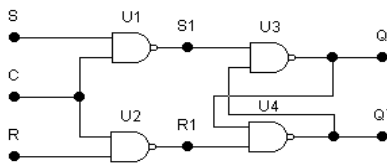


Рисунок 6.3 — Схема **RS**-триггера на логических элементах

Задания

1. Определите кодовые комбинации на выходе генератора слова для исследования триггера (согласно выданному преподавателем типу) по схеме на рисунке 6.3 в соответствии с описанной методикой. Полученную таблицу истинности сравните с таблицей истинности, вызываемой нажатием клавиши помощи <F1> после выделения на схеме триггера.

2. Составьте временную диаграмму работы выбранного триггера с помощью логического анализатора. Подключите к входам триггера периодические последовательности импульсов. Задайте сигналы таким образом, чтобы исключить логически недопустимую комбинацию (например, $R = S = 1$ для **RS**-триггера).

3. Изучите работу **JK**-триггера в асинхронном режиме (подавая сигналы на входы **R** и **S**). Определите, обладают ли входы **R** и **S** приоритетом перед остальными входами, т. е., влияют ли на работу триггера сигналы на остальных входах.

4. Исследуйте работу **D**-триггера в динамическом режиме. Используйте в качестве синхронизирующих импульсов сигнал с частотой в четыре раза выше, чем на входе **D**. Зарисуйте временные диаграммы. Проверьте с помощью анализатора, как работает триггер, если во время синхроимпульсов меняется сигнал **D**, т. е. поменяйте местами сигналы с входов **C** и **D**. Подключите ко входам **C** и **D** соответствующие этому случаю сигналы.

5. Зарисуйте временные диаграммы входных и выходных сигналов.

Контрольные вопросы

1. Что представляют собой триггеры, в каких современных микросхемах и для чего они используются?
2. Какие типы триггеров вы знаете, чем обусловлено их многообразие?
3. В чем состоит различие в работе триггеров со входами **R, S** и **R, S**?
4. Что представляет собой логически недопустимая комбинация входных сигналов **RS**-триггера?
5. Как работает триггер при $J = K$?
6. Как установить триггер в состояние «0» с использованием **J** и **K** входов?
7. Предложите схемы использования **JK**-триггера в качестве **RS**-триггера и **D**-триггера.
8. Спроектируйте **JK**-триггер на основе **RS**-триггера.
9. Что представляет собой **D**-триггер? Спроектируйте **D**-триггер на логических элементах.
10. Приведите схему статического **D**-триггера.

11. Какие сигналы необходимо подать на входы статического синхронного D-триггера для установки его в «1»?

12. В чем заключается отличие асинхронных триггеров от синхронных? Реализованы ли оба типа триггеров в программе EWB?

13. Какие сигналы необходимо подать на входы статического синхронного D-триггера для установки его в «1»?

14. В чем заключается отличие динамического D-триггера от статического?

Репозиторий Баргу

ЛАБОРАТОРНАЯ РАБОТА 7

СЧЕТЧИКИ. РЕГИСТРЫ

Цели:

- изучить различные варианты построения счетчиков;
- исследовать схемы счетчиков;
- провести моделирование счетчиков;
- исследовать различные типы регистров.

Часть 1 Счетчики

Теоретическая часть

Счетчиком называют устройство, сигналы на выходе которого отображают число импульсов, поступивших на счетный вход. Триггер на рисунке 6.3 может служить примером простейшего счетчика, который считает до двух. Счетчик, образованный цепочкой из m триггеров, может подсчитать в двоичном коде 2^m импульсов. Каждый из триггеров такой цепочки называют **разрядом счетчика**. Число m определяет количество разрядов двоичного числа, которое может быть записано в счетчик. Число $K_{сч} = 2^m$ называют коэффициентом (модулем) счета.

Информация снимается с прямых и (или) инверсных входов всех триггеров. В паузах между входными импульсами триггеры сохраняют свои состояния, т. е. счетчик запоминает число входных импульсов.

Нулевое состояние всех триггеров принимается за нулевое состояние счетчика в целом. Остальные состояния нумеруются по числу поступивших входных импульсов. Когда число входных импульсов $N_{вх} > K_{сч}$ происходит переполнение, после чего счетчик возвращается в нулевое состояние и цикл повторяется. Таким образом, коэффициент счета характеризует число входных импульсов, необходимое для выполнения одного цикла и возвращения в исходное состояние. Число входных импульсов и состояние счетчика взаимно определены только для первого цикла.

После завершения каждого цикла на выходах последнего триггера возникают перепады напряжения. Это определяет второе назначение счетчиков: деление числа входных импульсов. Если входные сигналы периодичны и следуют с частотой $F_{вх}$, то частота выходных сигналов равно $F_{вых} = F_{вх} / K_{сч}$. В этом случае коэффициент счета называется коэффициентом деления и обозначается как $K_{дел}$.

Основными эксплуатационными показателями счетчика являются емкость и быстродействие. Емкость счетчика, численно равная коэффициенту счета, равна числу импульсов за один цикл.

Быстродействие счетчика определяется двумя параметрами: разрешающей способностью $T_{\text{раз.сч}}$ и временем установки кода счетчика $T_{\text{уст}}$. Под разрешающей способностью подразумевают минимальное время между двумя входными сигналами, в течение которого не возникают сбои в работе. Обратная величина $F_{\text{макс}} = 1 / T_{\text{раз.сч}}$ называется максимальной частотой счета. Время установки кода $T_{\text{уст}}$ равно времени между моментом поступления входного сигнала и переходом счетчика в новое устойчивое состояние. Эти параметры зависят от быстродействия триггеров и способа их соединения между собой.

Счетчики различаются числом и типами триггеров, способами связей между ними, кодом, организацией счета и другими показателями. Цифровые счетчики классифицируются по следующим параметрам [1;2]:

- коэффициенту счета: двоичные (бинарные); двоично-десятичные (декадные) или с другим основанием счета; с произвольным постоянным и переменным (программируемым) коэффициентом счета;
- направлению счета: суммирующие, вычитающие и реверсивные;
- способу организации внутренних связей: с последовательным, параллельным или комбинированным переносом, кольцевые.

Классификационные признаки независимы и могут встречаться в разных сочетаниях, например, суммирующие счетчики бывают как с последовательным, так и с параллельным переносом, они могут иметь двоичный, десятичный и иной коэффициенты счета.

Введением дополнительных логических связей — обратных и прямых — двоичные счетчики преобразуются в не двоичные. Наибольшее распространение получили десятичные (декадные) счетчики, работающие с $K_{\text{сч}} = 10$ в двоично-десятичном коде (двоичный — по коду счета, десятичный — по числу состояний).

Десятичные счетчики организуются из четырехразрядных двоичных счетчиков. Избыточные шесть состояний исключаются введением дополнительных связей. Возможны два варианта построения схем: счет циклически идет от 0000 до 1001 и исходным состоянием служит 0110B = 6D; счет происходит до 1111B = 15D (**B**, **D** — обозначения двоичного и десятичного чисел). Первый вариант на практике применяется чаще.

В суммирующем счетчике каждый входной импульс увеличивает на единицу число, записанное в счетчик, при этом перенос информации из

одного разряда в другой, более старший, имеет место, когда происходит смена состояния «1» на «0».

Вычитающий счетчик действует обратным образом: двоичное число, хранящееся в счетчике, с каждым поступающим импульсом уменьшается на единицу. Переполнение вычитающего счетчика происходит после достижения им нулевого состояния. Перенос из младшего разряда в старший здесь имеет место при смене состояния младшего разряда с «0» на «1».

Реверсивный счетчик может работать в качестве суммирующего и вычитающего. Эти счетчики имеют дополнительные входы для задания направления счета. Режим работы определяется управляющими сигналами на этих входах. В программе EWB такие счетчики представлены ИМС 74163 и 74169.

Счетчики с последовательным переносом представляют собой цепочку триггеров, в которой импульсы, подлежащие счету, поступают на вход первого триггера, а сигнал переноса передается последовательно от одного разряда к другому.

Главное достоинство счетчиков с последовательным переносом — простота схемы. Увеличение разрядности осуществляется подключением дополнительных триггеров к выходу последнего триггера. Основным недостатком счетчиков с последовательным переносом — сравнительно низкое быстродействие, поскольку триггеры срабатывают последовательно, один за другим. Счетчики этого класса в библиотеке EWB не представлены.

Счетчики с параллельным переносом состоят из синхронных триггеров. Счетные импульсы подаются одновременно на все тактовые входы, а каждый из триггеров цепочки служит по отношению к последующим только источником информационных сигналов. Срабатывание триггеров параллельного счетчика происходит синхронно, и задержка переключения всего счетчика равна задержке одного триггера. В таких счетчиках используются **JK** и **D**-триггеры. В схемном отношении они сложнее счетчиков с последовательным переносом. Число разрядов у этих счетчиков обычно невелико (4...6), поскольку с повышением числа разрядов число внутренних логических связей быстро растет.

Счетчики с параллельным переносом (их чаще называют синхронными) в библиотеке EWB представлены счетчиками 74160, 74162, 74163 и 74169.

В счетчике с параллельно-последовательным переносом триггеры объединены в группы так, что отдельные группы образуют счетчики с параллельным переносом, а группы соединяются последовательно. В роли групп могут быть и готовые счетчики. Счетчики этого типа, как

правило, многоразрядные. Общий коэффициент счета равен произведению коэффициентов счета групп. По быстродействию они занимают промежуточное положение.

Счетчики-делители, оформленные как самостоятельные изделия, имеются в составе многих серий микросхем. Номенклатуру счетчиков отличает большое разнообразие. Многие из них обладают универсальными свойствами и позволяют управлять коэффициентом и направлением счета, вводить до начала цикла исходное число, прекращать счет по команде, наращивать число разрядов и т. п. С помощью готовых счетчиков можно решить большинство практических задач, возникающих перед разработчиком аппаратуры.

Проектирование счетчика сводится к определению числа триггеров и организации связей между ними и логическими элементами, а также вычислению разрешающей способности счетчика (максимальной частоты счета).

На первом шаге проектирования заданный коэффициент счета (деления) преобразуется в двоичный код. Число разрядов двоичного числа показывает, сколько триггеров должен иметь счетчик, а число единиц определяет число входов логического элемента. Входы элемента подключаются к прямым выходам Q тех триггеров, которые соответствуют единицам двоичного числа. Следует только учитывать, что первый, входной, триггер отображает младший разряд числа. Выход логического элемента соединяется с входами установки нуля (входы R) входы триггеров, от которых сделаны отводы, а также тех, которые непосредственно за ним следуют.

Порядок разработки устройств на базе счетчиков рассмотрим на примере цифровых часов, функциональная схема которых приведена на рисунке 7.1. Часы содержат три пары индикаторов для отображения часов, минут, секунд и два одиночных индикатора-разделителя. Индикаторы управляются от подсхем $\text{cont}24$ и $\text{cont}60$ с питанием от источника $+5\text{ В}$. В качестве задающего генератора используется функциональный генератор, режимы работы которого показаны на рисунке 7.2.

Следует отметить, что представленная на рисунке 7.1 схема обладает крайне низким быстродействием, поэтому реализовать режим секундомера путем повышения частоты задающего генератора не удалось. Блок $\text{cont}60$ представляет собой счетчик с коэффициентом $K_{\text{сч}} = 60$, его функциональная схема показана на рисунке 7.3. Блок содержит подсхему-счетчик $\text{cont}6$ ($K_{\text{сч}} = 6$) и двоично-десятичный счетчик 74160. Микросхема 74160 — декадный двоично-десятичный счетчик [1]. Он запускается

положительным перепадом тактового импульса и имеет синхронную загрузку (предварительную установку каждого триггера по входам **A**, **B**, **C**, **D**). Несколько счетчиков ИЕ9 образуют синхронный многодекадный счетчик. Сброс всех триггеров — асинхронный по общему входу сброса **R** (**CLR**).

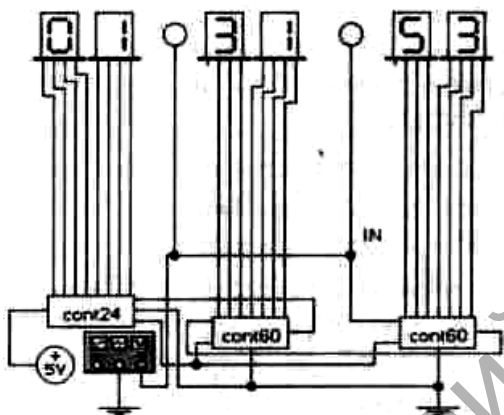


Рисунок 7.1 — Функциональная схема часов

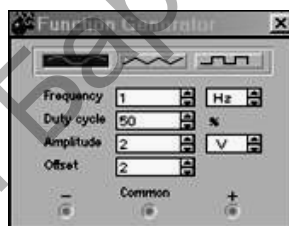


Рисунок 7.2 — Панель функционального генератора в схеме часов

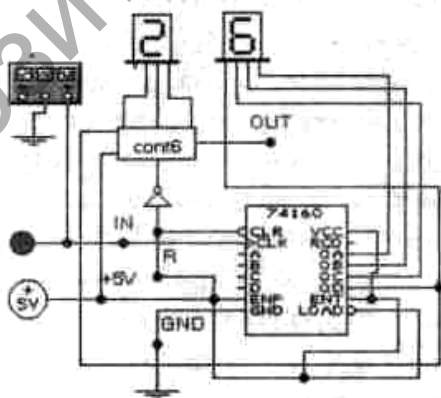


Рисунок 7.3 — Функциональная схема счетчика cont60

Схема счетчика cont6 показана на рисунке 7.4. Счетчик выполнен на трех **JK**-триггерах в счетном режиме (на **J**- и **K**-входы поданы сигналы «1»). Для обеспечения коэффициента счета $K_{сч} = 6$ использована обратная связь на элементе И **U2**, который срабатывает при коде 110 $B = 6D$, при этом сигнал «1» с его выхода через элемент ИЛИ **U1** поступает на **R**-входы триггеров, переводя их в нулевое состояние. Ко второму входу элемента **U1** подключен вход **R** для подачи внешнего сигнала сброса. Поскольку для рассматриваемого счетчика и счетчика 74160 эти сигналы различны (для первого это «1», а для второго — «0»), на входе **R** счетчика cont6 (см. рис. 7.2) включен инвертор. Схема счетчика часовых интервалов cont24 показана на рисунке 7.5. Счетчик выполнен на двух ИМС 74160 и обеспечивает коэффициент $K_{сч} = 24$.

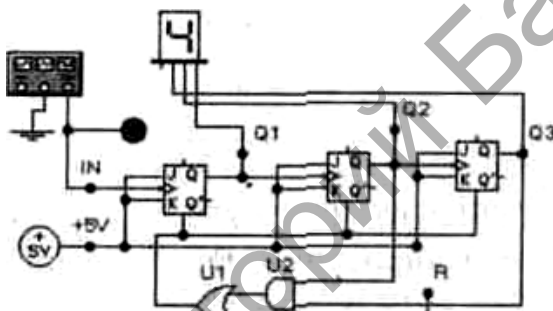


Рисунок 7.4 — Функциональная схема счетчика cont6

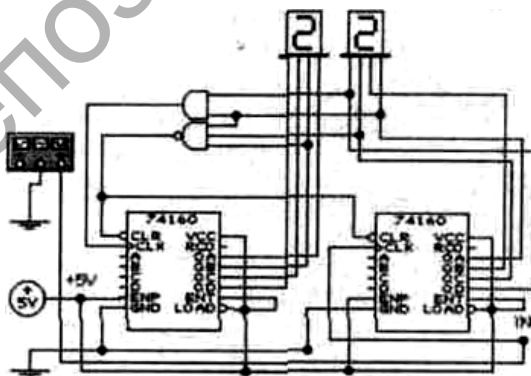


Рисунок 7.5 — Функциональная схема счетчика cont24

Задания

1. Разработайте схему счетчика с коэффициентом счета 3 на **JK**-и **D**-триггерах (см. рис. 7.4).
2. Проведите моделирование всех функциональных узлов часов, представленных на рисунке 7.1, выявите недостатки и устраните их.
3. Проведите моделирование и опишите порядок работы счетчика, представленного на рисунке 7.5.
4. Разработайте схему счетчика с коэффициентом счета согласно выданному преподавателем заданию. Обратите внимание на то, как выполнен счетчик с коэффициентом счета равным $K_{сч} = 6$ (см. рис. 7.4). Что необходимо сделать, чтобы увеличить коэффициент счета до $K_{сч} = 8$. Что будет происходить, если добавить в цепочку еще один **JK**-триггер?

Часть 2 Регистры

Теоретические сведения

Основное назначение регистров — хранение и преобразование многоразрядных двоичных чисел. Регистры, наряду со счетчиками и запоминающими устройствами, являются наиболее распространенными устройствами цифровой техники. При сравнительной простоте регистры обладают большими функциональными возможностями. Они используются в качестве управляющих и запоминающих устройств, генераторов и преобразователей кодов, счетчиков, делителей частоты, узлов временной задержки. Элементами структуры регистров являются синхронные триггеры **D**- или **JK**-типа с динамическим или статическим управлением. Одиночный триггер может запоминать (регистровать) один разряд (бит) двоичной информации. Такой триггер можно считать одnorазрядным регистром. Занесение информации в регистр называют **операцией ввода записи**. Выдача информации к внешним устройствам характеризует операцию вывода или считывания. Запись информации в регистр не требует его предварительного обнуления.

Все регистры в зависимости от функциональных свойств подразделяются на две категории: накопительные (регистры памяти, хранения) и сдвигающие. Последние делятся по способу ввода и вывода информации

на параллельные, последовательные и комбинированные (параллельно-последовательные и последовательно-параллельные), по направлению передачи (сдвига) информации — на однонаправленные и реверсивные.

Наиболее простыми регистрами являются регистры памяти. Их назначение — хранение двоичной информации небольшого объема в течение короткого промежутка времени. Они представляют собой набор синхронных триггеров, каждый из которых хранит один разряд двоичного числа. Ввод (запись) и вывод (считывание) информации производятся параллельным кодом. Ввод обеспечивается тактовым импульсом, с приходом очередного тактового импульса записанная информация обновляется. Считывание производится в прямом или обратном коде (в последнем случае с инверсных выходов).

Регистры хранения представляют собой наборы триггеров с независимыми информационными входами и обычно общим тактовым входом. В таком качестве используются синхронные триггеры, составленные из микросхем, содержащих в одном корпусе несколько самостоятельных триггеров, например, K155TM8 (74175), K155TM9 (74179) и другие, которые можно рассматривать как 4—6-разрядные регистры памяти. Нарращивание разрядности регистров памяти достигается добавлением нужного числа триггеров, тактовые входы которых подсоединяют к шине синхронизации.

Регистр K155ИР15(74173) является библиотечным компонентом EWB и может служить примером устройства хранения с тремя выходными состояниями. Схема его включения приведена на рисунке 7.6.

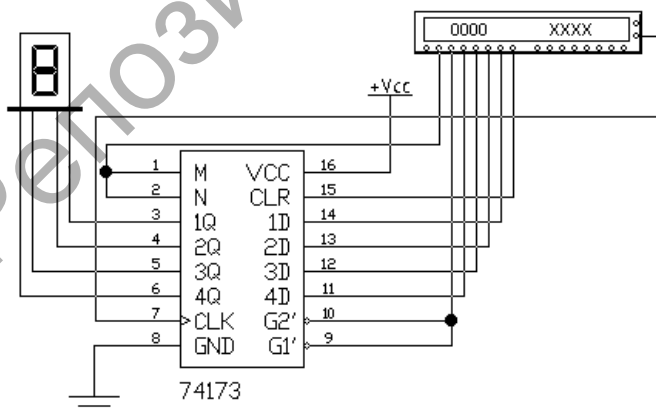


Рисунок 7.6 — Схема включения регистра 74173

Как видно, K155ИР15 — четырехразрядный регистр. Он имеет выходы **1Q...4Q** с третьим **Z**-состоянием (при сигнале «1» на выходах **G2, G1**), а его входы **1D...4D** снабжены логическими элементами разрешения записи путем подачи логического «0» на входы **M, N** (в EWB ошибочно показаны прямыми). Используется регистр как четырехразрядный источник кода, способный обслуживать непосредственно шину данных цифровой системы.

Загрузка информации в регистр производится синхронно с положительным перепадом тактового импульса, если на входах **M, N** присутствуют напряжения низкого уровня. Если на одном из этих входов напряжение высокого уровня, после прихода положительного тактового перепада в регистре должны оставаться прежние данные. Вход сброса **CLR** имеет высокий активный уровень. Если на входы **G2, G1** подано напряжение активного низкого уровня, данные, содержащиеся в регистре, отображаются на выходах **1Q...4Q**, присутствие хотя бы одного напряжения высокого уровня на входах разрешения **G2, G1** вызывает **Z**-состояние (размыкание) для выходных линий. При этом данные из регистра в шину данных систем не проходят, а выходы регистра не влияют на работу других аналогичных выходов, присоединенных к проводникам шины. На работу входов сброса **CLR** и тактового **C** смена уровней на входах разрешения влияния не оказывает. Режимы работы генератора в схеме на рисунке 7.6 и некоторые кодовые комбинации показаны на рисунке 7.7.

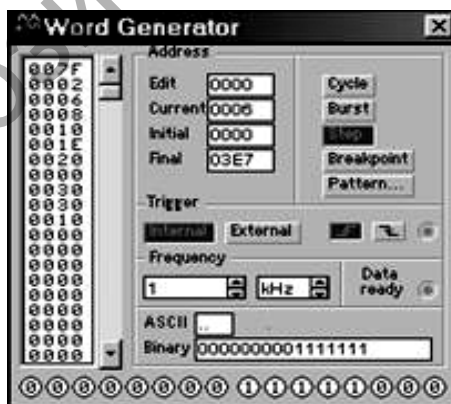


Рисунок 7.7 — Лицевая панель генератора слова с установкой режимов работы

Вторым наиболее распространенным классом регистров являются регистры сдвига, которые отличаются большим разнообразием как в функциональном отношении, так и в отношении схемных решений и характеристик. Регистры сдвига, кроме операции хранения, осуществляют преобразование последовательного двоичного кода в параллельный, а параллельного — в последовательный, выполняют арифметические и логические операции, служат в качестве элементов временной задержки. Своим названием они обязаны характерной для них операции сдвига. С приходом каждого тактового импульса происходит перезапись (сдвиг) содержимого триггера каждого разряда в соседний разряд без изменения порядка следования единиц и нулей. При сдвиге информации вправо после каждого тактового импульса бит из более старшего разряда сдвигается в младший, а при сдвиге влево — наоборот.

Рассмотрим регистр сдвига на примере библиотечного регистра 74195 (K155ИР12), схема включения которого показана на рисунке 7.8. ИМС 74195 — быстродействующий регистр для выполнения операций сдвига, счета, накопления и взаимного параллельно-последовательного преобразования цифровых слов. Через вход **LD/SH** загружаются параллельные данные и производится их сдвиг вправо. Если на этом входе присутствует напряжение высокого уровня, через входы первого триггера **J** и **K** в регистр вводятся последовательные данные. Вход **J** имеет «высокий» активный уровень, вход **K** — «низкий». Если эти входы соединить, получим простой **D**-вход. Данные сдвигаются в направлении от **QA** к **QB**, **QC**, а затем к **QD** после каждого положительного перепада на тактовом входе **CLK**.

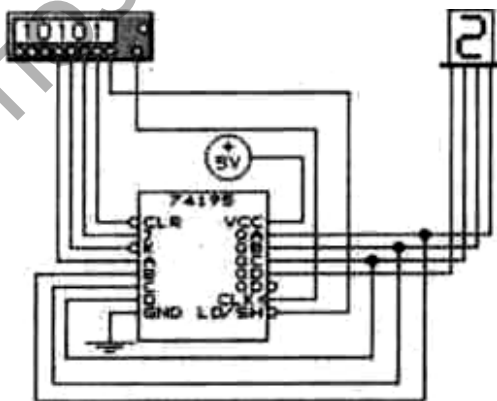


Рисунок 7.8 — Схема включения регистра 74195

Если на входе **LD/SH** присутствует напряжение «низкого» (активного) уровня, все четыре триггеры регистра запускаются одним тактовым перепадом (от «низкого» уровня к «высокому»). Тогда данные от параллельных входов **A...D** передаются на соответствующие выходы **QA...QD**. Сдвиг данных влево обеспечивается в схеме, где каждый выход **Q_n** соединен внешней перемычкой со входом **D_n**, т. е. схема включения на рисунке 7.8 соответствует только режиму приема и хранения данных.

Схема включения ИМС в режиме сдвига показана на рисунке 7.9, режимы работы генератора слова — на рисунке 7.10. Для режима сдвига напряжение на входе **LD/SH** необходимо зафиксировать на высоком уровне. Все операции в регистре ИР12 строго синхронны и запускается он фронтом импульса, поэтому логические уровни на входах **J, K, D_n, LD/SH** можно произвольно изменять до прихода фронта запуска. Низким уровнем на входе **CLR** всем выходным сигналам присваивается низкий уровень.

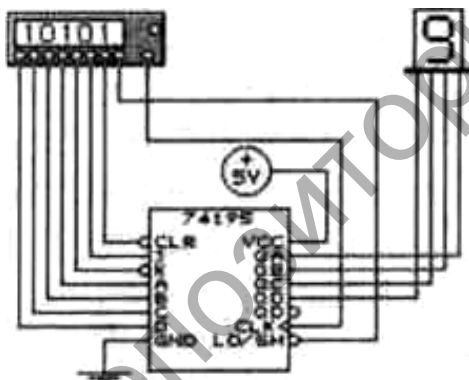


Рисунок 7.9 — Схема включения регистра 74195 в режиме сдвига

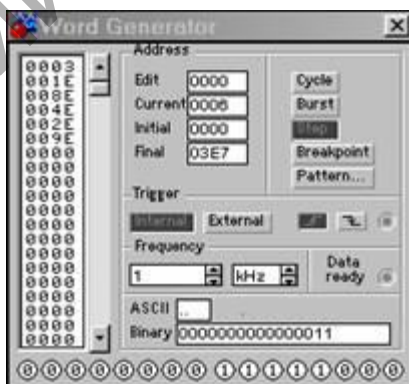


Рисунок 7.10 — Лицевая панель генератора слова с режимами работы в схеме рисунок 7.9

Напряжение низкого уровня на входе **CLR** также означает запрет на действие тактового импульса **CLK**. Для правильного сброса данных необходимо выбрать момент, когда на входе **CLR** присутствует напряжение низкого уровня.

Задания

1. Проведите моделирование регистра 74133 по схеме на рисунке 7.6. При моделировании необходимо выбрать с помощью генератора слова двоичные комбинации, которые позволяют проверить все режимы его работы. Целесообразно также составить так называемую таблицу состояния, напоминающую таблицу истинности.

2. Проведите моделирование регистра 74195 в режиме приема данных (см. рис. 7.8).

3. Для приведенной на рисунке 7.9 схемы исследуйте следующие режимы сдвига: «1» — сдвиг и установка по первому каскаду ($JK = 11$); «2» — сдвиг и сброс по первому каскаду ($JK = 00$); «3» — сдвиг и переключение первого каскада ($JK = 10$); «4» — сдвиг и хранение в первом каскаде ($JK = 01$). При этом, как указывается выше, **CLR** — «1», **LD/SH** — «1», состояние входа **A** безразлично.

Контрольные вопросы

1. Что представляет собой счетчики, какого типа они бывают?
2. Составьте схему, приведите таблицу состояний и временные диаграммы для двоичного суммирующего асинхронного счетчика с модулем счета $K = 4$.
3. Составьте схему, приведите таблицу состояний и временные диаграммы для двоичного вычитающего асинхронного счетчика с модулем счета $K = 4$.
4. Каким образом создаются счетчики с коэффициентом счета, не кратным 2?
5. Опишите принцип построения двоично-десятичного счетчика.
6. Что представляет собой программируемый счетчик (74160)?
7. Что представляет собой регистр, какие функции он может выполнять?
8. Назовите типы регистров и их возможные области применения.
9. Каким образом осуществляется хранение данных в регистрах?
10. Приведите схему изменения направления сдвига в регистре.

ЛАБОРАТОРНАЯ РАБОТА 8

ОПЕРАТИВНОЕ ЗАПОМИНАЮЩЕЕ УСТРОЙСТВО. ПОСТОЯННОЕ ЗАПОМИНАЮЩЕЕ УСТРОЙСТВО

Цель: изучить принципы работы и области применения оперативных запоминающих устройств и постоянных запоминающих устройств.

Часть 1

Оперативные запоминающие устройства

Теоретические сведения

Оперативные запоминающие устройства являются неотъемлемой частью микропроцессорных систем различного назначения. Они делятся на два класса: статические и динамические. В первых запоминание информации производится на триггерах, а в последних — на конденсаторах емкостью порядка 0,5 пФ. Длительность хранения информации в статических ОЗУ не ограничена, тогда как в динамических, она ограничена временем саморазряда конденсатора, что требует специальных средств регенерации и дополнительных затрат времени на этот процесс.

Конструктивно любое ОЗУ состоит из двух блоков: матрицы запоминающих элементов и дешифратора адреса. По технологическим соображениям матрица чаще всего имеет двухкоординатную дешифрацию адреса — по строкам и столбцам. На рисунке 8.1 показана матрица 16-битного статического ОЗУ. Матрица состоит из 16 ячеек памяти mem_i , схема которой приведена на рисунке 8.2, а настройка генератора слова на рисунке 8.3. Каждая ячейка памяти адресуется по входам **X, Y** путем выбора дешифраторами адресных линий по строкам **Ax0...Ax3** и по столбцам **Ay0...Ay3** (см. рис. 8.1) и подачи по выбранным линиям сигнала логической единицы. При этом в выбранной ячейке памяти срабатывает двухходовый элемент И (**U1**), подготавливая цепи чтения-записи информации на входных **DI0...DI3** или выходных **DO0...DO3** разрядных шинах. Разрешающим сигналом для выдачи адреса является **CS** (*Chip Select* — выбор кристалла), который подается на вход разрешения счетчика адреса (**Addr_cnt**) или такой же вход дешифраторов, подключенных к выходу счетчика.

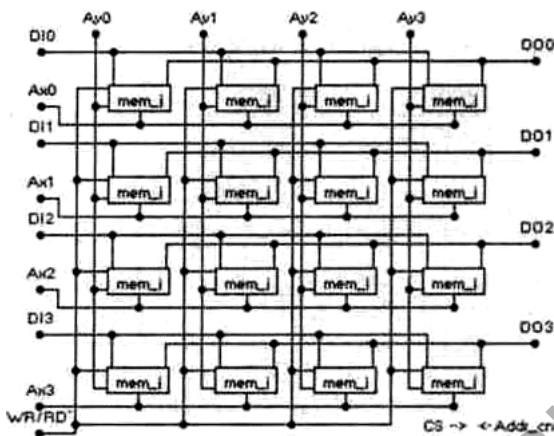


Рисунок 8.1 — Матрица 16-битного ОЗУ

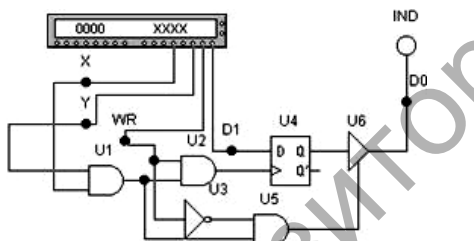


Рисунок 8.2 — Схема ячейки памяти mem_i

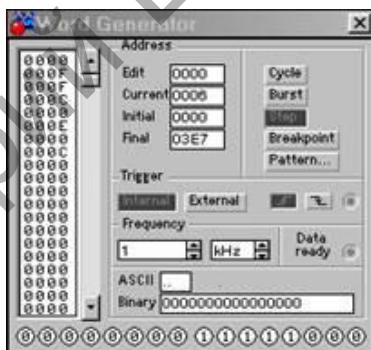


Рисунок 8.3 — Лицевая панель генератора слова с установками для схемы на рисунке 8.2

Современные запоминающие устройства статического типа отличаются высоким быстродействием и в микропроцессорных системах используются ограниченно из-за сравнительно высокой стоимости. В таких системах они используются только в качестве кэш-памяти (от англ. *Cache* — запас) — быстродействующей буферной памяти между процессором и основной памятью, служащей для частичной компенсации разницы в скорости процессора и основной памяти, в которую заносятся наиболее часто используемые данные. Когда процессор первый

раз обращается к ячейке памяти, ее содержимое параллельно копируется в кэш, и в случае повторного обращения может быть с гораздо большей скоростью из нее извлечено. При записи в память информация попадает в кэш и одновременно копируется в память (схема **Write Through** — прямая или сквозная запись) или копируется через некоторое время (схема **Write Back** — обратная запись). При обратной записи, называемой так же буферизованной сквозной записью, информация копируется в память в первом же свободном такте, а при отложенной (**Delayed Write**) — когда для помещения в кэш нового значения не оказывается свободной области. При этом в основное ОЗУ вытесняются сравнительно редко используемые данные. Вторая схема более эффективна, но и более сложна за счет необходимости поддержания соответствия содержимого кэша и основной памяти.

В микропроцессорных системах в качестве ОЗУ чаще всего используются динамические ОЗУ с запоминающим конденсатором, которые отличаются большим многообразием.

В динамической памяти ячейки выполнены на основе областей с накоплением заряда, занимающих гораздо меньшую площадь, чем триггеры, и практически не потребляющих энергию при хранении информации. При записи бита в такую ячейку в ней формируется электрический заряд, который сохраняется в течение нескольких миллисекунд. Для постоянного сохранения заряда ячейки необходимо регенерировать (перезаписывать) ее содержимое. Ячейки микросхем динамической памяти также организованы в виде прямоугольной матрицы. При обращении к микросхеме на ее входы вначале подается адрес строки матрицы, сопровождаемый сигналом **RAS** (Row Address Strobe — строб адреса строки), затем, через некоторое время, — адрес столбца, сопровождаемый сигналом **CAS** (Column Address Strobe — строб адреса столбца). При каждом обращении к отдельной ячейке регенерируются все ячейки выбранной строки, поэтому для полной регенерации матрицы достаточно перебрать адреса строк. Ячейки динамической памяти имеют сравнительно малое быстродействие (десятки—сотни наносекунд), но большую удельную плотность (порядка нескольких мегабайт на корпус) и потребляют меньше энергии.

Обычные ОЗУ часто называют асинхронными, так как установка адреса и подача управляющих сигналов могут выполняться в произвольные моменты времени, необходимо только соблюдение временных соотношений между этими сигналами. В них включены охранные интервалы, необходимые для установления сигналов. Существуют также синхронные виды памяти, получившие внешний синхросигнал, к им-

пульсам которого жестко привязаны моменты подачи адресов и обмена данными. Они позволяют более полно использовать внутреннюю конвейеризацию и блочный доступ.

FPM DRAM (Fast Page Mode DRAM — динамическая память с быстрым страничным доступом) — память со страничным доступом, отличающаяся от обычной динамической памяти тем, что после выбора строки матрицы и удержания сигнала **RAS** допускает многократную установку адреса столбца, стробируемого сигналом **CAS**, а также быструю регенерацию по схеме «**CAS** прежде **RAS**». Первое позволяет ускорить блочные передачи, когда весь блок данных или его часть находятся внутри одной строки матрицы, называемой в этой системе страницей, а второе — снизить затраты времени на регенерацию памяти.

EDO (Extend Data Out — расширенное время удержания данных на выходе) фактически представляет собой обычные микросхемы **FPM**, на выходе которых установлены регистры-защелки данных. При страничном обмене такие микросхемы работают в режиме конвейера: удерживают на выходах данных содержимое последней выбранной ячейки, в то время как на их входы уже подается адрес следующей выбираемой ячейки.

BEDO (Burst EDO — EDO с блочным доступом) — память на основе **EDO**, работающая не одиночными, а пакетными циклами чтения/записи. При наличии памяти **BEDO** отпадает необходимость постоянной подачи последовательных адресов на входы микросхем с соблюдением необходимых временных задержек, достаточно стробировать переход к очередному слову отдельным сигналом.

SDRAM (Synchronous DRAM — синхронная динамическая память) — память с синхронным доступом, работающая быстрее обычной асинхронной (**FPM/EDO/BEDO**). Кроме синхронного доступа, **SDRAM** использует внутреннее разделение массива памяти на два независимых банка, что позволяет совмещать выборку из одного банка с установкой адреса в другом.

PB SRAM (Pipelined Burst SRAM — статическая память с блочным конвейерным доступом) — разновидность синхронных **SRAM** с внутренней конвейеризацией, за счет которой примерно вдвое повышается скорость обмена блоками данных.

Кроме основного ОЗУ, устройством памяти снабжается и устройство отображения информации — видеодисплейная система. Такая память называется видеопамью и располагается на плате видеоадаптера.

Видеопамью служит для хранения изображения. От ее объема зависит максимально возможное разрешение видеокарты — $A \times B \times C$,

где **A** — количество точек по горизонтали, **B** — по вертикали, **C** — количество возможных цветов каждой точки. Например, для разрешения $640 \times 480 \times 16$ достаточно иметь видеопамять 265 Кбайт, для $1024 \times 786 \times 65536$ — 2 Мбайт и т. д. Поскольку для хранения цветов отводится целое число разрядов, количество цветов всегда является целой степенью 2 (16 цветов — 4 разряда, 64 к. — 16 и т. д.)

В видеоадаптерах используются следующие типы видеопамати:

1) **FPM DRAM** (Fast Page Mode Dynamic RAM — динамическое ОЗУ с быстрым строчным доступом) — основной тип видеопамати;

2) **VRAM** (Video RAM — видео ОЗУ) — двухпортовая **DRAM** с поддержкой одновременного доступа со стороны видеоадаптера и центрального процессора компьютера. Она позволяет совмещать во времени вывод изображения на экран и его обработку в видеопамати, что сокращает задержки и увеличивает скорость работы;

3) **EDO DRAM** (Extended Data Out DRAM — динамическое ОЗУ с расширенным временем удержания данных на выходе) — память с элементами конвейеризации, позволяющей несколько ускорить обмен блоками данных с видеопаматью;

4) **SGRAM** (Synchronous Graphics RAM — синхронное графическое ОЗУ) — вариант **DRAM** с синхронным доступом, когда все управляющие сигналы изменяются одновременно с системным тактовым синхросигналом, что позволяет уменьшить временные задержки;

5) **MDRAM** (Multibank DRAM — многобанковое ОЗУ) — вариант **DRAM**, организованный в виде множества независимых банков объемом по 32 Кбайт каждый, работающих в конвейерном режиме.

Увеличение скорости обращения видеопроцессора к видеопамати, кроме повышения пропускной способности адаптера, позволяет повысить максимальную частоту регенерации изображения, что снижает утомляемость глаз пользователя.

Микросхемы памяти имеют четыре основные характеристики — тип, объем, структуру и время доступа. Тип обозначает статическую или динамическую память, объем показывает общую емкость памяти, а структура — количество ячеек памяти и разрядность каждой ячейки. Время доступа характеризует скорость работы микросхемы и обычно указывается в наносекундах после тире в конце наименования.

ИМС памяти реализуется в корпусах следующих типов:

1) **SIP** (Single In line Package — корпус с одним рядом выводов) — микросхема с одним рядом выводов, устанавливаемая вертикально;

2) **SIMM** (Single In line Memory Module — модуль памяти с одним рядом контактов) — модуль памяти, вставляемый в зажимный разъем. Он применяется в платах старого типа, принтерах и прочих устройствах, имеет контакты с двух сторон модуля, но все они соединены между собой, образуя ряд контактов. На **SIMM** в настоящее время устанавливаются преимущественно микросхемы **FPM/EDO/BEDO**;

3) **DIMM** (Dual In line Memory Module — модуль с двумя рядами контактов) — модуль памяти, похожий на **SIMM**, но с раздельными контактами, за счет чего увеличивается разрядность или число банков памяти в модуле.

Задания

1. Проведите моделирование ячейки статической памяти, представленной на рисунке 8.2. Задачей моделирования является выбор двоичных комбинаций для сигналов на входе ячейки и регистрации результата на выходе ячейки с помощью индикатора **IND**.

2. На базе схемы, представленной на рисунке 8.1, разработайте схему четырехразрядного ОЗУ с применением генератора слова. При этом в схеме на рисунке 8.1 используйте только четыре младших адреса (два по строкам и два по столбцам) и соответственно только две шины данных (две входные и две выходные). К выходным шинам подключите индикаторы.

Часть 2

Постоянное запоминающее устройство

Теоретические сведения

Постоянные запоминающие устройства (ПЗУ) делятся на четыре типа:

- 1) масочные, программируемые на заводе-изготовителе с применением специальных масок;
- 2) однократно программируемые потребителем путем пережиганием нихромовых или поликремневых перемычек;
- 3) многократно программируемые потребителем со стиранием записанной информации ультрафиолетовым излучением;

4) многократно программируемые потребителем с электрическим стиранием информации.

Рассмотрим ПЗУ второго типа, которое состоит из дешифратора $n \times 2^n$ и подключенных к его выходам схем ИЛИ с плавкими перемычками (рис. 8.4). Оно содержит дешифратор *pzu_dcd* (**A**, **B**, — кодовые входы, **E** — вход разрешения, активный сигнал высокого уровня), к выходам которых можно подключить четыре элемента 4ИЛИ с дополнительными устройствами. На рисунке 8.4 показаны два таких элемента, выполненных в виде отдельных подсхем *pzu_un1* и *pzu_un2*. Несмотря на то что они одинаковы, наращивание их на схеме путем копирования исключено из-за наличия пережигаемых перемычек. При наличии одноименных подсхем пережигание перемычки в одной подсхеме автоматически приведет к пережиганию такой же перемычки в другой. На схеме **D0**, **D1** — выходы младшего и первого разрядов.

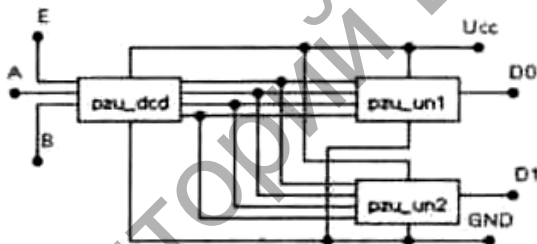


Рисунок 8.4 — Схема двухразрядного ПЗУ

Схема дешифратора *pzu_dcd* показана на рисунке 8.5. Дешифратор выполнен на трех элементах НЕ и четырех элементах 3ИЛИ—НЕ на транзисторах (рис. 8.6).

Ячейка ПЗУ работает следующим образом. В исходном состоянии транзисторы **T1...T4** и **T6** закрыты, и при подключенной к **T6** нагрузке на его выходе **D0** формируется сигнал логической единицы (около +5 В). При подаче на входы **A**, **B** дешифратора заданной кодовой комбинации, а на вход разрешения **E** — сигнала логической «1», один из транзисторов **T1...T4** откроется и на выходе **D0** сформируется сигнал логического «0».

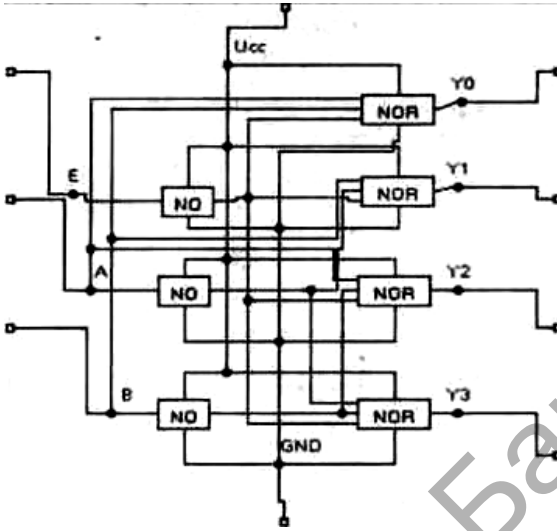


Рисунок 8.5 — Внутренняя структура подсхемы дешифратора.

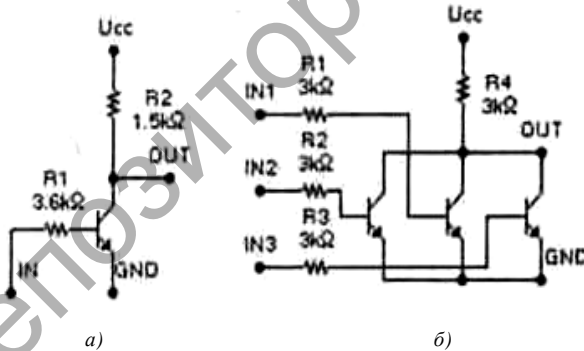


Рисунок 8.6 — Схемы элементов НЕ (а) и ИЛИ—НЕ (б)

Так, например, при $A = B = 1$ откроется транзистор T_4 и сигнал логической «1» с его эмиттера через переключку S_4 поступит на делитель на резисторах R_2, R_3 , транзистор T_6 откроется, и на его выходе сформируется сигнал логического «0» (рис. 8.7).

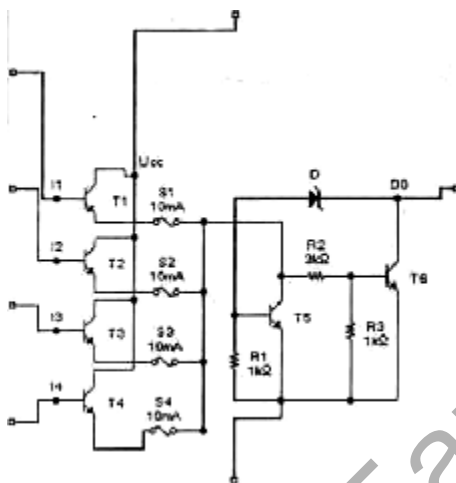


Рисунок 8.7 — Внутренняя структура ячейки памяти двухразрядного ПЗУ

Пережигание перемычек составляет суть программирования и осуществляется отдельно для каждого разряда (ячейки) следующим образом:

- на входы **A**, **B** (см. рис. 8.4) подается двоичная комбинация, соответствующая адресу пережигаемой перемычки в программируемом разряде (в ячейке `pzu_unx`, где `x` — номер ячейки);

- к выходу ячейки **Dx** через резистор нагрузки (его сопротивление для конкретных ИМС указывается в документации, для K155PE3 составляет около 300 Ом) подключается источник напряжения 12,5 В, в результате чего стабилитрон **D** пробивается и транзистор **T5** открывается;

- на вход разрешения **E** на короткое время подается сигнал логической «1», при этом через один из открытых транзисторов **T1...T2** и **T5** протекает ток, достаточный для пережигания соответствующей перемычки (длительность разрешающего сигнала на входе **E** в промышленных программаторах может увеличиваться после нескольких неудачных попыток программирования одной и той же ячейки);

- источник 12,5 В отключается. После раскрытия соответствующей подсхемы можно убедиться, что перемычка действительно разрушена.

Заключительным этапом программирования серийных микросхем ПЗУ в промышленных условиях является электротермотренировка, которая проводится чаще всего в течении 168 часов при повышенной температуре, после чего производится дополнительный контроль запи-

Постоянное запоминающее устройство с ультрафиолетовым стиранием используется в микропроцессорных системах для хранения управляющих программ, в частности, для размещения **BIOS** (Basic Input/Output System — основная система ввода/вывода, записанная в ПЗУ, отсюда ее полное название **ROM BIOS**), который представляет собой набор программ проверки и обслуживания аппаратуры компьютера и выполняет роль посредника между операционной системой и аппаратурой. **BIOS** получает управление при включении системной платы, тестирует саму плату и основные блоки компьютера — видеоадаптер, клавиатуру, контроллеры дисков и портов ввода/вывода, настраивает чипсет платы и загружает внешнюю операционную систему.

Раньше **BIOS** помещался в однократно программируемые ПЗУ либо ПЗУ с ультрафиолетовым стиранием; сейчас же выпускаются платы с электрически программируемыми ПЗУ (**Flash ROM**), которые допускают перепрограммирование **BIOS** средствами самой платы. Это позволяет исправлять заводские ошибки в **BIOS**, изменять заводские установки по умолчанию, программировать собственные экранные заставки и т. д.

Задания

1. Используя схему на рисунке 8.8, проведите моделирование процесса программирования ПЗУ с пережигаемыми перемычками.
2. Составьте схему ПЗУ на базе двухразрядного ПЗУ на рисунке 8.4 и проведите моделирование процесса программирования одной из его ячеек памяти.

Контрольные вопросы

1. Какие существуют типы памяти?
2. Где в современных компьютерах используется память статического типа?
3. Чем отличается динамическая память от статической?
4. Какие типы памяти используются в современных компьютерах?
5. Что представляет собой видеопамять и как она связана с характеристиками отображаемой на дисплее информации?
6. Какие типы памяти используются в качестве видеопамати?
7. Какое конструктивное оформление имеют микросхемы памяти?
8. Какие существуют типы ПЗУ и где они используются?
9. Что такое **BIOS** для компьютера, видео контроллера, накопителя на жестком магнитном диске (винчестера) и других периферийных устройств?
10. Какие типы ПЗУ используются для хранения программ **BIOS**?

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. *Карлашук, В.* Электронная лаборатория на IBM PC. Программа Electronics Workbench и ее применение / В. Карлашук. — М. : Солон, 2003. — 736 с.
2. *Панфилов, Д. И.* Электротехника и электроника в экспериментах и упражнениях : Практикум на Electronics Workbench / Д. И. Панфилов, В. С. Иванов, И. Н. Чепурин. — М. : ДОДЭКА, 1999. — 304 с.
3. *Симонович, С. Р.* Информатика : Базовый курс / С. Р. Симонович. — СПб. : Питер, 2005. — 640 с.
4. *Таненбаум, Э.* Архитектура компьютера / Э. Таненбаум. — СПб. : Питер, 2005.
5. *Фигурнов, В. Э.* IBM PC для пользователя / В. Э. Фигурнов. — 7-е изд., испр. и доп. — М. : Финансы и статистика: НПО «Информатика и компьютеры», 1997. — 467 с.
6. *Финогенов, К. Ф.* Устройства и настройка ПК / К. Ф. Финогенов. — М. : Бином, Лаб. знаний, 2004.
7. *Хамахер, К.* Организация ЭВМ / К. Хамахер, Э. Вранешич, С. Заки. — СПб. : Питер, 2003.
8. *Цилькер, Б. Я.* Организация ЭВМ и систем / Б. Я. Цилькер, С. А. Орлов. — СПб. : Питер, 2006. — 688 с. : ил.

Учебное издание

**ОРГАНИЗАЦИЯ
И ФУНКЦИОНИРОВАНИЕ ЭВМ**

**Лабораторный практикум
для студентов II курса
специальности 1-40 01 02
Информационные системы и технологии**

Корректор *А. М. Войнова*
Технический редактор *Н. В. Иванова*

Ответственный за выпуск: *Е. Г. Хохол*

Подписано в печать 24.03.2011.
Формат 60 × 84 1/16. Бумага офсетная.
Гарнитура Таймс. Отпечатано на ризографе.
Усл. печ. л. 4,65. Уч-изд. л. 3,28.
Заказ 1. Тираж 70 экз.

ЛИ 02330/0552803 от 09.02.2010

Издатель и полиграфическое исполнение:
учреждение образования
«Барановичский государственный университет»,
225404, г. Барановичи, ул. Войкова, 21.

Репозиторий Баргу

Репозиторий Баргу