

Министерство образования и науки Российской Федерации
**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ «ЛЭТИ»**

Факультет компьютерных технологий и информатики

Кафедра автоматике и процессов управления

А.Н. Ветров

ИНФОРМАТИКА

(первое издание)

Учебное издание
(учебник)

Рекомендовано в качестве методического пособия (учебника) для студентов,
обучающихся по техническим специальностям

Санкт-Петербург
2008 г.

УДК 004.01(03):004.2-9

Ветров А.Н. Информатика: Учебник. – СПб.: «СПбГЭТУ "ЛЭТИ"», 2005, 2008, деп. во «ВИНИТИ» «РАН», 2008, деп. в «РАО», 2008. – 331 с.

Изложены генезис, концептуальные основы и развитие информации и информатики, необходимые и достаточные для обеспечения начального изучения ее объекта и предмета (информационное общество, информационный кризис и революция, информатизация, информационные ресурсы, продукты и услуги на информационном рынке, информатика, информация и данные, сигналы и квантование), арифметические основы цифровых автоматов (информационное взаимодействие, представление и кодирование информации, системы счисления и правила выполнения арифметических операций, правила перевода целых чисел и правильных дробей из одной системы счисления в другую), логические основы цифровых автоматов (логическая функция, приоритет выполнения и способы представления логических операций, основные законы булевой алгебры логики и схемы комбинаторной логики с памятью состояния), генезис и тенденции развития архитектуры информационных систем (виды и особенности систем и технологий, поколения развития ЭВМ, классическая, неоклассическая и современная архитектуры ЭВМ), элементы и устройства архитектуры современных компьютеров (понятие и основные принципы, материнская плата, слоты и разъемы, центральный процессор, внутренняя и внешняя память, системный блок и блок питания, физическая и логическая основы накопителей информации на гибких, жестких, оптических и электронных дисках и картах памяти, периферийное оборудование ПЭВМ, дисплей, принтер, клавиатура и манипуляторы), программное обеспечение современных компьютеров (командный, WIMP, SILK интерфейсы взаимодействия с пользователем, локальные и сетевые, однопользовательские и многопользовательские операционные системы, операционные системы MS DOS и MS Windows, пакет прикладных программ, пакет офисных программ и пакет утилит).

Предназначен для учащихся учреждений среднего (общего) и специального образования, студентов учреждений высшего профессионального образования, слушателей курсов повышения квалификации и профессиональной переподготовки по техническим направлениям и специальностям, а также ориентирован на использование преподавателями, учеными, сотрудниками НИИ и интересующимися в области ИТ.

Ил. 114, табл. 32, список литературных источников из 26 наименований.

РЕЦЕНЗЕНТЫ:

- Филиппов П.В., д.т.н., профессор, капитан 1 ранга, начальник кафедры «Систем и средств автоматизации управления» «Военно-морского института радиоэлектроники им. А.С. Попова»;
- Хименко В.И., Заслуженный деятель науки РФ, д.т.н., профессор, первый проректор «Санкт-Петербургского государственного университета аэрокосмического приборостроения», заведующий кафедрой «Компьютерной математики и программирования»;
- Александров Д.П., д.т.н., профессор, профессор «Факультета военного обучения» «Балтийского государственного технического университета "ВОЕНМЕХ" им. Д.Ф. Устинова»

Оглавление

Введение	6
1. Генезис и концептуальные основы информации, информатики, информатизации в современном обществе	10
1.1. Информационное общество и процесс информатизации.....	12
1.2. Сущность и роль информационного кризиса в развитии информационных технологий	14
1.3. Информационные ресурсы, продукты и услуги на информационном рынке в информационном обществе	18
1.4. Правовые аспекты регулирования отношений на информационном рынке....	23
1.5. Генезис и развитие информатики	27
1.6. Предмет информатики	28
1.7. Структура информатики.....	31
1.8. Информация, ее представление и свойства	34
1.8.1. Понятие и характерные черты информации	35
1.8.2. Свойства информации.....	36
1.8.3. Классификация информации.....	37
1.8.4. Понятие и структура процесса информационного взаимодействия	38
1.9. Подходы к оценке количества информации.....	40
1.10. Основные виды сигнала как материального носителя информации	42
1.10.1. Квантование непрерывного (аналогового) сигнала	44
1.10.2. Восстановление аналогового сигнала по дискретному	46
2. Арифметические основы цифровых автоматов	47
2.1. Основные формы представления чисел в компьютере	49
2.2. Основные виды позиционных систем счисления	53
2.3. Правила перевода чисел из одной системы счисления в другую.....	56
2.3.1. Правила перевода целых чисел	58
2.3.2. Правила перевода правильных дробей.....	66
2.3.3. Правила перевода дробных чисел.....	77
2.4. Правила выполнения простейших арифметических операций	80
2.4.1. Правила сложения	81
2.4.2. Правила вычитания	89
2.4.3. Правила умножения	97
2.4.4. Правила деления	112
3. Логические основы цифровых автоматов	121
3.1. Логические переменные и функции алгебры логики	122
3.2. Способы задания и реализации логических функций	123
3.3. Основные логические операции	129
3.3.1. Операция инверсии НЕ, функция логического отрицания	131
3.3.2. Операция конъюнкции, функция логического умножения И.....	134
3.3.3. Операция дизъюнкции, функция логического сложения ИЛИ	138
3.3.4. Гибридные логические операции И-НЕ и ИЛИ-НЕ.....	142
3.4. Основные законы булевой алгебры.....	154
3.5. Схемы комбинационной логики, обеспечивающие хранение информации ..	169
3.5.1. Асинхронный RS-триггер.....	179
3.5.2. Синхронный RS-триггер.....	181
3.5.3. Тактируемый триггер.....	183

4.	Генезис и теоретические основы построения информационных вычислительных систем и компьютеров	189
4.1.	Понятие информационной системы	190
4.2.	Основные виды информационных систем.....	194
4.2.1.	Информационная технология получения данных.....	196
4.2.2.	Информационная технология обработки данных	197
4.2.3.	Информационная технология хранения данных	199
4.2.4.	Информационная технология передачи данных	201
4.3.	Зарождение архитектуры компьютеров.....	202
4.4.	Развитие архитектуры электронных вычислительных машин	203
4.5.	Основные принципы функционирования классических компьютеров	204
4.6.	Структура и принцип функционирования классической (принстонской) электронной вычислительной машины	205
4.7.	Элементы неоклассической архитектуры электронной вычислительной машины..	209
4.7.1.	Структура и принцип действия устройства управления	210
4.7.2.	Структура и принцип действия арифметико-логического устройства	213
4.7.3.	Внутренняя память	215
4.7.4.	Внешняя память.....	219
5.	Базовые элементы и устройства архитектуры современных персональных электронных вычислительных машин	220
5.1.	Классификация современных компьютеров	221
5.2.	Основные принципы построения архитектуры современных компьютеров	223
5.3.	Структура персонального компьютера.....	224
5.4.	Особенности расположения внутренних и внешних элементов и устройств современных компьютеров	225
5.5.	Внешние элементы персональной электронной вычислительной машины..	226
5.5.1.	Системный блок и блок питания	227
5.5.2.	Видеодисплей.....	231
5.5.3.	Клавиатура	233
5.5.4.	Манипулятор типа мышь, трекбол и тачпад.....	235
5.6.	Внутренние элементы структуры персонального компьютера	236
5.6.1.	Системная плата компьютера.....	237
5.6.2.	Центральный процессор	238
5.6.3.	Оперативное запоминающее устройство	239
5.6.4.	Внешнее запоминающее устройство	241
5.7.	Накопители информации.....	242
5.7.1.	Накопители информации на перфокартах	243
5.7.2.	Накопители информации на перфолентах	244
5.7.3.	Накопители информации на магнитных лентах.....	245
5.7.4.	Накопители информации на гибких магнитных дисках.....	246
5.7.5.	Накопители информации на жестких магнитных дисках	248
5.8.	Лазерные накопители информации на оптических дисках.....	252
5.8.1.	Лазерные накопители информации на оптических дисках CD-ROM.....	253
5.8.2.	Лазерные накопители информации на оптических дисках DVD-ROM.....	254
5.9.	Электронные накопители информации	255
5.9.1.	Внутренний электронный накопитель информации.....	256
5.9.2.	Внешний и носимый электронный накопитель информации	258

5.10.	Периферийное оборудование компьютера	260
5.10.1.	Устройства ввода информации	261
5.10.2.	Устройства вывода информации.....	262
5.10.3.	Устройства сетевого взаимодействия и передачи данных	263
5.11.	Устройства печати информации на разные носители	272
5.11.1.	Матричные принтеры.....	273
5.11.2.	Струйные принтеры	274
5.11.3.	Лазерные принтеры	276
5.12.	Сканеры.....	279
5.12.1.	Планшетный сканер	280
5.12.2.	Ручной сканер	281
5.12.3.	Сканер для сканирования штрих-кодов	282
6.	Программное обеспечение современных компьютеров	283
6.1.	Современные интерфейсы взаимодействия	286
6.2.	Системное программное обеспечение	287
6.2.1.	Локальные однопользовательские операционные системы.....	288
6.2.2.	Сетевые и многопользовательские операционные системы.....	289
6.2.3.	Оболочки	292
6.2.4.	Программное обеспечение для промышленных контроллеров.....	293
6.3.	Прикладное программное обеспечение	296
6.3.1.	Пакеты прикладных программ офисного назначения	297
6.3.2.	Прикладные программы для автоматизации бухгалтерского учета	300
6.3.3.	Программное обеспечение для защиты информации и информационных систем	320
6.4.	Пакеты утилит и сервисные программы.....	322
6.4.1.	Пакеты диагностических утилит	323
6.4.2.	Пакеты сервисных программ	326
	Заключение.....	328
	Перечень литературных источников.....	330

Введение

«Информация – это информация,
а не материя и не энергия»

Н. Винер

Информатика как фундаментальная наука и прикладная дисциплина выступает относительно новым предметом изучения среди начинающих пользователей и квалифицированных специалистов, а ее концепты употребляются в лексиконе субъектов информационной индустрии, которые обеспечивают поддержку процесса создания, распределения и использования информационных ресурсов, продуктов и услуг. Несмотря на повсеместное употребление научного термина «Информатика» и предмета изучения «Теория информации», сегодня их содержание остается не раскрытым до конца в силу непрерывности процесса информатизации и появления новаций в области новых информационных и коммуникационных технологий.

Интуитивно понятно, что они связаны с понятием «информация», а также с организацией процессов ее обработки посредством использования компьютеров, вычислительных комплексов и сетей. Исторически достоверным является факт происхождения слова ИНФОРМАТИКА в русском языке: оно образовано двумя латинскими словами – INFORMAció (ИНФОРМАция) и automaTIGUE (автомаТИКА).

Научный термин «Информатика» рассматривается дифференциально по отношению к предметной области и проблемной среде: как фундаментальная научная дисциплина, как прикладная отрасль производства, как совокупность информационных технологий для создания средств автоматизации для «преобразования» информации.

Зарождение предмета «Информатика» связывают с появлением феномена «Информация», которая существует до и вне зависимости от ее потребителя, а также с созданием средств накопления, обработки и передачи информации выраженной в форме данных, которые в свою очередь выступают производными от информации.

Вследствие расширения сферы использования средств вычислительной техники и повышения влияния «информационного кризиса» (проявляется в виде «информационной лавины» или «информационного голода»), который время от времени переживает человечество, актуализируется необходимость изучения теории информации каждым представителем информационного общества, а дисциплина включена в программы и курсы среднего (общего), специального и высшего образования.

Данное учебное пособие раскрывает основные понятия теории информации, внешние и внутренние свойства информации, основные виды сигналов как материальных носителей информации и особенности их взаимного преобразования, арифметические и логические основы цифровых автоматов, специфику разработки информационных вычислительных систем, классическую архитектуру ЭВМ и принципы функционирования ее компонентов, архитектуру современных цифровых автоматов и их аппаратное, программное и алгоритмическое обеспечение.

Представленное читателю методическое пособие (учебник) включает введение, шесть глав в основной части, заключение и список литературных источников.

В первой главе «Генезис и концептуальные основы информации, информатики, информатизации в современном обществе» определяются несколько понятий: информационное общество и его характерные черты, информационный рынок и его основные сегменты, сущность информационных ресурсов, продуктов и услуг информационной индустрии, а также правовые аспекты государственного регулирования отношений возникающих между субъектами на информационном рынке.

Выделяется понятие информатики и ее структура, а также понятие и характерные черты информации, ее представление и свойства, классификация, понятие и структура процесса информационного взаимодействия источников и потребителей, подходы к оценке количества информации, основные виды сигналов как материальных носителей информации, квантование непрерывного аналогового сигнала.

Поскольку обработка информации предполагает использование цифровых автоматов, выступающих дискретными по принципу своего функционирования, то возникает необходимость представления информации на стандартном и расширенном алфавитах кода, посредством которых представляются данные при кодировании сигнала для дальнейшей обработки посредством вычислительных процедур. Возникает необходимость использования систем счисления с разными основаниями, которые применимы в современной информатике, наряду с правилами арифметического счета и взаимного перевода чисел из одной системы счисления в другую. В настоящий момент понятие информационного общества соответствует в большей мере ряду стран: США, Канада, Япония, Англия, страны Западной Европы, Россия.

Во второй главе «Арифметические основы цифровых автоматов» представлены основные формы представления (кодирования) чисел в ЭВМ (компьютере) и виды позиционных систем счисления, правила перевода чисел из одной системы счисления в другую, а также правила выполнения простейших арифметических операций.

В третьей главе «Логические основы цифровых автоматов» рассматриваются логические переменные и функции алгебры логики, способы задания и реализации логических функций (аналитический, табличный, функциональная схема логического устройства, временная диаграмма состояний достигнутых на выходах логического устройства в зависимости от комбинаций значений уровней сигналов соответствующих логической единице или логическому нулю на его входах, электрическая принципиальная схема аппаратной реализации логической функции или устройства), основные и гибридные логические операции (инверсия НЕ, конъюнкция И, дизъюнкция ИЛИ, конъюнкция инверсная по выходу И-НЕ, дизъюнкция инверсная по выходу ИЛИ-НЕ), а также основные законы алгебры логики как правила эквивалентного преобразования сложного логического выражения к простому, схемы комбинаторной логики с памятью логических состояний для хранения информации (асинхронный RS-триггер, синхронный RS-триггер, тактируемый T-триггер).

В четвертой главе «Генезис и теоретические основы построения информационно-вычислительных систем и компьютеров» приводятся: понятие, основные виды информационных систем и технологий получения, обработки, хранения и передачи данных, сведения о зарождении и развитии классической архитектуры компьютеров, основные принципы функционирования классических электронно-вычислительных машин, структура и принцип функционирования классической (принстонской) архитектуры компьютера, элементы неоклассической архитектуры электронно-вычислительной машины, структура и принцип действия устройств классической и неоклассической архитектуры компьютера (устройство управления, арифметико-логическое устройство, память), классификация основных видов современных цифровых автоматов.

В пятой части «Базовые элементы и устройства архитектуры современных персональных электронно-вычислительных машин» представляются: основные принципы построения архитектуры и принципы функционирования современных компьютеров, внешние элементы современного компьютера (системный блок и блок питания, монитор, клавиатура, манипуляторы типа мышь, трекбол, тачпад), внутренние элементы структуры электронно-вычислительной машины (системная плата, центральный процессор, оперативное запоминающее устройство, внешнее запоминающее устройство), накопители информации на гибких и жестких магнитных дисках, лазерные накопители информации на оптических дисках CD-ROM и DVD-ROM, внутренние и внешние электронные накопители информации, периферийное оборудование компьютера: устройства ввода и вывода информации, устройства сетевого взаимодействия и передачи данных, устройства печати информации на разного рода носители (матричные, струйные и лазерные принтеры).

В шестой части «Программное обеспечение современных компьютеров» рассматривается системное программное обеспечение (локальные и сетевые, однопользовательские и многопользовательские операционные системы, оболочки, программное обеспечение для промышленных контроллеров), прикладное программное обеспечение (пакеты прикладных программ офисного назначения, прикладные программы для автоматизации бухгалтерского учета, программное обеспечение для защиты информации и информационных систем), пакеты утилит и сервисные программы (пакеты диагностических утилит и пакеты сервисных программ).

В заключении приводятся некоторые выводы, которые характеризуют особенности использования современных средств автоматизации для анализа и повышения эффективности информационного взаимодействия между разными источниками и потребителями разнородной информации, а также с целью сокращения временных и транзакционных издержек при производстве и обращения информационных ресурсов, продуктов и услуг на информационном рынке, созданных на основе ключевых инноваций в области информационных и коммуникационных технологий.

Перечень литературных источников включает публикации по предмету «Информатика», а также смежным предметным областям и сферам исследования.

Сегодня можно выделить два типа пользователей современных компьютеров:

- специалисты в области информационных технологий и теории информации;
- начинающие пользователи низкой квалификации, которые используют ЭВМ.

Специалистам и начинающим пользователям рекомендуется вначале ознакомиться с содержанием методического пособия (учебника), а затем внимательно изучить последовательно введение, перечень сокращений и условных обозначений, шесть частей основной части, заключение и перечень литературных источников.

Если специалистам в области информационных технологий и теории информации покажется знакомым содержание данного печатного труда, то, тем не менее, необходимо обратить особое внимание на некоторые части, которые представляют исключительный теоретический и практический интерес для любого читателя:

- глава «Арифметические основы цифровых автоматов» содержит полезные сведения об алгоритмах преобразования дробных чисел из одной системы счисления в другую и об осуществлении простейших арифметических операций над числами в разных системах счисления для реализации вычислительных процедур при разработке алгоритмического и программного обеспечения;
- глава «Логические основы цифровых автоматов» представляет интерес для специалистов, которые разрабатывают функциональные схемы логических устройств и электрические принципиальные схемы их аппаратной реализации, а также имеются таблицы истинности и временные диаграммы результирующих состояний достигнутых на выходах относительно бинарных комбинаций поданных на входы конъюнктора, дизъюнктора, инвертора, асинхронного RS-триггера, синхронного RS-триггера и тактируемого T-триггера;
- глава «Генезис и теоретические основы построения информационно-вычислительных систем и компьютеров» содержит некоторые примеры использования средств автоматизации в разных прикладных сферах деятельности;
- глава «Базовые элементы и устройства архитектуры современных персональных электронно-вычислительных машин» представляет особый интерес, поскольку содержит описание внешних и внутренних элементов современной архитектуры ЭВМ, а также особенности их технической и аппаратной реализации;
- глава «Программное обеспечение современных компьютеров» содержит классификацию современных операционных систем и прикладного программного обеспечения, а также пакетов сервисных программ и утилит.

Автором предполагается, что читатель знаком с концептуальными основами теории информации и современными информационными технологиями, а изложение материала в методическом пособии (учебнике) ориентировано на широкий круг потребителей различных образовательных услуг: школьники 10-11 классов в учреждениях среднего (общего) профессионального образования, студенты всех курсов в учреждениях высшего профессионального образования и их преподаватели, а также ученые и исследователи, которые заняты в данной предметной области.

1. Генезис и концептуальные основы информации, информатики, информатизации в современном обществе

Генезис и развитие теории информации связывают с началом проведения исследований закономерностей процесса информационного взаимодействия в различных сферах деятельности человека: фундаментальная и прикладная наука, техника и технология, бизнес, политика, религия, быденная практика и прочие.

Информационный обмен заключается в симплексной (однонаправленной) или дуплексной (двунаправленной) передаче набора информационных сообщений между источником и потребителем информации, которая передается по каналу связи с использованием определенного физического принципа или протокола обмена в виде совокупности сигналов разного типа: механических, электрических, электромагнитных, оптических (в видимом и невидимом диапазоне спектра излучения).

Собственно информация существует до и вне зависимости от ее потребителя, генерируется различными способами, но от источников двух основных типов:

- естественным источником – человек, животное, объект неживой природы;
- искусственным источником – техническое средство, устройство или организм, которые произведены человеком посредством использования ручной, промышленной (мануфактурной) или автоматизированной технологии сборки продукции, а также единичного, серийного и массового производства.

Информация передается в виде сигналов двух основных типов в разнородных информационных системах с использованием разных принципов функционирования:

- непрерывные – используются в аналоговых системах, которые разрабатываются с использованием электронных вакуумных ламп, резисторов и конденсаторов, а также описываются аналитической функцией, непрерывной во времени (за исключением наличия разрывов первого и второго рода), что также зависит от математических особенностей функции, в частности: поведения ее производной, области ее определения и множества ее значений;
- дискретные сигналы – характерны для систем, которые описываются математическими функциями, отражающими множество мгновенных значений (измерений) сигнала во времени (решетчатая и смещенная решетчатая функция), причем на промежутках между точками измерения в пределах области определения функции нельзя однозначно выявить значение уровня сигнала:
 - импульсные системы – конструируются с использованием механических сопряжений и передач, а также электрических принципиальных схем и элементной базы, включающей классические элементы (электронные вакуумные лампы, резисторы, конденсаторы) и полупроводниковые элементы (диоды, транзисторы, тиристоры, интегральные логические микросхемы);
 - цифровые системы – базируются на технологиях многослойного печатного монтажа, а также современных полупроводниковых приборах (малые и большие интегральные микросхемы на основе полупроводниковых кристаллов).

В ходе процесса становления современной цивилизации можно выделить несколько фаз и исторических периодов, которые характеризуются появлением письменности, книгопечатания, изобретением мануфактурного производства, электричества, радио и телевидения, Интернета, компьютеров и нейрокомпьютеров.

Очевидно, что при переходе общества в постиндустриальную фазу своего непрерывного развития наблюдается интенсификация роста разнородных потоков информации, которые циркулируют в различных сферах социальной активности, повышается частота удвоений объема информации как агрегата знаний по спектру предметных областей: гуманитарные, технические, медицинские и прочие науки.

На рубеже конца XIX – начала XX веков происходит очередное удвоение совокупного агрегата знаний, что, в свою очередь, актуализирует проблему очередного информационного кризиса и соответственно очередной информационной революции, проявляющихся в виде естественных процессов имеющих обоснование:

- информационный взрыв (лавина) – возникает переполнение каналов передачи данных информацией разного рода и назначения, что обуславливает проблему получения, классификации, систематизации, обработки, хранения, адресного распределения информации, а также инициирует создание, внедрение и сопровождение средств автоматизации, которые повышают эффективность процесса информационного обмена в разных предметных областях;
- информационный голод – обусловлен двумя взаимосвязанными факторами:
 - во-первых,- переполнение среды потребления информации инициирует сложность получения, систематизации и классификации информации;
 - во-вторых,- недостаток определенных документов, массивов данных, информационных хранилищ и средств автоматизации хранения, поиска, передачи информации по отношению к группам потребителей позволяет говорить о невозможности своевременного получения достоверной и актуальной информации, отражающей состояние объекта, процесса или явления.

Средства автоматизации в рамках современных информационных технологий в информационном обществе представляются набором основных компонентов:

- аппаратное обеспечение (hardware) – образовано техническими средствами и устройствами ввода, вывода, обработки информации, которые используются в рамках определенного поколения развития архитектуры компьютера;
- программное обеспечение (software) – включает системные (операционные системы, оболочки), сервисные (диагностические утилиты), прикладные (пакет программ офисного, издательского, конструкторского назначения) и прочие программы (микропрограммы для промышленных контроллеров, регуляторов, системных плат, видеокарт, сетевых адаптеров, модемов, сканеров);
- алгоритмическое обеспечение (brainware) – представлено алгоритмами, математическими методами и процедурами, которые реализуют эффективную обработку информации при получении ее от источника и передаче потребителю.

1.1. Информационное общество и процесс информатизации

Информатизация различных предметных областей и сфер деятельности обеспечивает рационализацию процесса создания оптимальных условий для удовлетворения информационных потребностей большинства членов информационного общества, а также реализует поддержку разработки, внедрения, использования и сопровождения средств автоматизации различного назначения и уровня сложности.

Определение

Информационным (постиндустриальным) обществом называют такой вид общества, в котором приоритет отдается информационной форме взаимодействия на основе достижений и инноваций в области информационных технологий, а большинство социальных субъектов занято созданием, распределением и использованием информационных ресурсов, продуктов и услуг для повышения эффективности процессов в производственной и непромышленной сферах деятельности.

Определение

Под **информатизацией общества** понимают непрерывный организованный социально-экономический и научно-технический процесс создания, распределения и использования информационных ресурсов, продуктов и услуг для удовлетворения информационных потребностей всех членов информационного общества, повышения эффективности информационного взаимодействия в различных сферах деятельности специалистов на базе средств автоматизации и инноваций в области информационных и коммуникационных технологий, что позволяет реализовать на государственном уровне демократические права открытого доступа физических (ФЛ) и юридических лиц (ЮЛ) к информации на пути к синтезу единой информационной среды.

В частности, академиком А.П. Ершовым предлагается использовать термин «**инфосфера**», исходя из которого предполагается совместное использование большинством членов постиндустриального общества информационных ресурсов, продуктов и услуг в единой информационной среде потребления информации.

Таким образом, **целью информатизации** также является создание новых информационных технологий, средств автоматизации и информационного общества, в котором большинство социальных субъектов (граждан определенного государства) занято поиском, хранением, обработкой и распределением информации.

Информатизация выступает комплексной научной задачей, обуславливающей выработку политики и стратегии государственными органами, а также инициирует рассмотрение и появление новых направлений в области фундаментальной и прикладной науки, а также способствует повышению квалификации и практических навыков каждого члена информационного (постиндустриального) общества.

На современном этапе развития цивилизации переход общества в постиндустриальную фазу своего развития способствует становлению фундаментальных основ информатики и разработке новых информационных технологий и средств автоматизации (аппаратного, программного и алгоритмического обеспечения).

Характерными чертами информационного общества выступают следующие:

- решена проблема информационного кризиса, который проявляется в среде потребления информации в виде информационной лавины или голода;
- информационные технологии и ресурсы приобретают существенный приоритет по отношению к технологиям материального производства и ресурсам;
- информация неисчерпаема при ее потреблении и адресном распределении по отношению к возобновляемым и не возобновляемым материальным ресурсам;
- распределение информационных ресурсов, продуктов и услуг осуществляется на информационном рынке, обеспечивающем процесс ценообразования;
- в основу информационного общества закладывается автоматизированная генерация, хранение, обработка, передача и использование информационных массивов посредством информационных технологий и средств автоматизации;
- информационные технологии приобретают глобальный характер, охватывая все сферы социальной активности и продуктивной деятельности общества;
- формируется информационное единство всей человеческой цивилизации;
- с помощью аппаратного, программного и алгоритмического обеспечения реализуется открытый доступ каждого человека к информационным ресурсам, которые созданы и используются в процессе становления цивилизации;
- при разработке средств автоматизации и информационных технологий учитываются гуманистические принципы и защита окружающей среды.

Помимо перечисленных ранее положительных результатов процесса информатизации общества, возможно возникновение позитивных и негативных тенденций и факторов, которые его неизбежно сопровождают в различных областях и сферах:

- позитивные тенденции внедрения и применения информационных технологий:
 - интернационализация и развитие транснациональных корпораций способствует распространению средств автоматизации и обучению людей;
 - преобладающее влияние приобретают средства массовой информации;
 - появляются информационные системы поиска и качественного отбора достоверной информации в рамках нескольких информационных ресурсов;
- негативные факторы, сопутствующие процессу информатизации следующие:
 - возникает необходимость разработки политики и стратегии процесса информатизации на государственном уровне консолидировано с научно-исследовательскими институтами и образовательными учреждениями;
 - информационные технологии могут разрушить частную жизнь человека;
 - современное информационное общество обуславливает появление информационной культуры, которая накладывает дополнительные требования к уровню образования, но некоторые люди испытывают сложности социальной адаптации, вследствие отсутствия свободного времени и низкого уровня базового образования в области информационных технологий;
 - информационный рынок обуславливает повышение квалификации субъектов.

1.2. Сущность и роль информационного кризиса в развитии информационных технологий

Экспоненциальное увеличение совокупного агрегата знаний по различным предметным областям и сферам деятельности общества обусловлено интенсификацией непрерывного роста разнородных потоков информации, что, в свою очередь, инициирует возникновение **информационного кризиса** или **информационного взрыва** не только в информационной индустрии, но и на информационном рынке.

Информационный кризис или взрыв имеет ряд негативных проявлений:

- появляются противоречия между ограниченными физиологическими, психологическими и лингвистическими возможностями человека при восприятии, обработке и понимании содержания информации, а также существенным увеличением количества потоков данных, документов, массивов и хранилищ первичной и систематизированной информации по разным отраслям наук;
- накопленный агрегат знаний по совокупности предметных областей меняется вначале очень медленно и частота удвоения количества циркулирующей информации минимальна, но уже с 1900 г. он удваивается каждые 50 лет, к 1950 г. удвоение происходит каждые 10 лет, а к 1970 г. – каждые 5 лет, с 1990 г. – оно происходит практически ежегодно, при этом четко начинает наблюдаться экспоненциальная закономерность процесса накопления знаний;
- возникает необходимость повторного исследования закономерностей информационных процессов в различных предметных областях и сферах деятельности членов информационного общества с целью модернизации аппаратного, программного и алгоритмического обеспечения для повышения уровня автоматизации процессов создания, распределения и использования информационных ресурсов, продуктов и услуг, что позволяет быстро усовершенствовать архитектуру существующих и новых информационных систем;
- существует большое количество бесполезной и избыточной информации, вследствие появления недостоверных источников, сложностей с систематизацией, поиском, хранением и передачей данных по каналам связи, которая затрудняет восприятие полезной и актуальной для потребителя информации;
- возникают определенные экономические, политические, социальные, организационные барьеры, препятствующие распространению информации;
- повышаются требования к точности, достоверности, актуальности, доступности информации содержащейся на бумажных и электронных носителях;
- внутренние свойства информации (неоднородность, неаддитивность, несоответствие формы и содержания) существенно усложняют процесс создания средств автоматизации независимо от предметной области их использования;
- возникает проблема разграничения доступа к информации в зависимости от ее назначения, а также категории ее промежуточного и конечного потребителя;
- ограничение открытого доступа к информации обусловленное секретностью.

Информационный кризис является системным явлением и ставит задачу разработки новых информационных технологий и средств автоматизации (аппаратного, программного и алгоритмического обеспечения) для повышения эффективности информационного взаимодействия субъектов в сфере создания, распределения и использования информации разного рода и назначения на определенных носителях.

В истории формирования цивилизации произошло несколько информационных кризисов, которые выступали предпосылками информационных революций.

Определение

Революция – кардинальное качественное или количественное преобразование общественных отношений в определенной сфере деятельности общества, которое обуславливает пересмотр классических принципов, методов, технологий и прочего, а также инициирует создание, внедрение и использование инновационных подходов.

Определение

Под **информационной революцией** понимают эволюционно обусловленные качественные и количественные преобразования технологий обработки информации и общественных отношений, обусловленные кардинальными изменениями в сфере создания, распределения и использования информационных ресурсов, продуктов и услуг, которые обеспечивают прирост продуктивности деятельности человека.

Следствием глубоких кардинальных преобразований в сфере получения, обработки и передачи информации разного рода является приобретение обществом:

- новых методов и технологий поиска, обработки и передачи информации;
- модернизированной структуры информационных потоков для оптимизации внедрения и использования аппаратного, программного и алгоритмического обеспечения в разных сферах деятельности информационного общества;
- новых средств автоматизации процессов создания, распределения и использования информационных ресурсов, продуктов и услуг в разных областях;
- усовершенствованных технологий материального производства с целью внедрения энергосберегающих ресурсов и сырьевой базы, а также снижения себестоимости продукции, которая производится посредством использования автоматизированной гибкой реконфигурируемой сборочной линии (конвейера);
- возможностей внедрения информационных технологий, что позволяет устранить внутренние свойства информации (неоднородность, неаддитивность, некоммутативность, несоответствие формы и содержания структур данных);
- средства автоматизации обработки и передачи информации позволяют повысить потребительскую привлекательность информации (ценность информации по отношению к потребителю и предметной области) на информационном рынке за счет улучшения внешних свойств информации (точность и надежность, полнота и глубина, достоверность и актуальность).

Комплексно информационный кризис обуславливает появление нового качества, позволяющего существенно повысить эффективность создания, распределения и использования информационных ресурсов, продуктов и услуг в разных сферах.

Предпосылки каждой информационной революции в сфере потребления информации чрезвычайно сложны, поскольку носят субъективную основу при их формулировке и интерпретации. Возникновение предпосылок и развитие революции существенно влияет на появление определенных информационных технологий.

В ходе исторического процесса выделяют ряд **информационных революций**, предпосылки которых возникали в разные исторические эпохи и периоды времени:

- **первая (III – IV в. д.н.э.)** связана с процессом образования локальных цивилизаций, изобретением письменности и алфавита национальных языков, что позволило передавать накопленные знания из поколения в поколение;
- **вторая (середина XVI в.)** вызвана изобретением технологии печати книг и технических средств для обеспечения книгопечатания, которое радикально изменило индустриальное общество, а также культуру и деятельность людей;
- **третья (конец XIX в.)** обусловлена изобретением электричества, благодаря которому появились телеграф, телефон, радио, которые позволяют оперативно получать, передавать и искать информацию в любом объеме;
- **четвертая (70-е гг. XX в.)** связана с изобретением полупроводниковых элементов, микропроцессорной технологии и появлением компьютера, а также изобретением технологии печатного монтажа при реализации электрических принципиальных схем, что позволило усовершенствовать аппаратную реализацию элементов информационных систем разного вида;
- **пятая (90-е гг. XX в.)** обусловлена появлением и усовершенствованием технологии многослойного печатного монтажа, полупроводникового лазера для реализации электрических принципиальных схем нового поколения, микросхем малой, средней и большой степени интеграции, что позволило обеспечить возможность аппаратной реализации элементов комбинаторной логики (простейшие логические операции, память, коммутатор), микропроцессора (сверхбольшая интегральная микросхема), а также модернизировать топологию и принципы функционирования локальных и глобальных вычислительных сетей (коммуникационные технологии), систем передачи данных и информационных систем, этот период в целом характеризуется тремя фундаментальными инновациями в сфере средств обработки информации:
 - переход от механических и электрических технических средств автоматизации сбора, обработки и передачи данных к электронным компонентам;
 - минимизация габаритных размеров информационных систем, их узлов, устройств, приборов, датчиков, контроллеров, локальных регуляторов;
 - создание программных средств на языках высокого уровня и интегрированных сред для разработки аппаратного, программного и алгоритмического обеспечения ЭВМ, промышленных контроллеров, информационно-поисковых систем, Веб-серверов, систем управления и мониторинга состояния сложных динамических объектов, процессов и явлений.

Для создания целостного представления о двух последних периодах необходимо познакомиться с приведенной ниже справкой о смене поколений электронных вычислительных машин (ЭВМ), а также сопоставить эти сведения с этапами эволюции и модернизации элементной базы, которая используется инженерами при аппаратной реализации цифровых автоматов и разных информационных систем.

Для анализа динамики эволюции аппаратного, программного и алгоритмического обеспечения ЭВМ выделяют несколько поколений в процессе их развития (табл. 1.1).

Таблица 1.1

Данные о смене поколений электронных вычислительных машин

№ п.п.	Наименование поколения	Период времени	Габаритные размеры и энергопотребление	Аппаратное обеспечение и быстродействие, надежность	Программное обеспечение информационных систем
1.	первое	начало 50-х гг.	очень большие; очень большое	электронные лампы, реле, конденсаторы; низкое; низкая	программированием в машинных кодах (Assembler)
2.	второе	конец 50-х гг.	большие; большое	полупроводниковые приборы; низкое; низкая	алгоритмические и языки описания данных (Algol, Cobol)
3.	третье	начало 60-х гг.	средние; среднее	интегральные микросхемы, многослойный печатный монтаж; среднее; средняя	локальные вычислительные сети для совместного использования ресурсов
4.	четвертое	середина 70-х гг.	малые; малое	большие интегральные микросхемы (микропроцессоры)	массовый выпуск персональных ЭВМ, многопроцессорные системы
5.	пятое	середина 90-х гг.	очень малые; очень малое	микросхемы сверхбольшой степени интеграции	нейрокомпьютеры, глобальные сети, распределенная обработка данных

Генезис и развитие синергетики обусловили разработку нейросетевых технологий (Г. Хакен), а также начало проведения исследований открытых информационных систем. Начали стремительно развиваться коммуникационные технологии.

Последняя информационная революция выдвигает на первый план новую отрасль – **информационную индустрию**, связанную с производством технических средств, методов и технологий для производства средств автоматизации. Важнейшими составляющими информационной индустрии становятся все виды информационных технологий и телекоммуникаций. Современная информационная технология опирается на достижения в области компьютерной техники и средств связи.

1.3. Информационные ресурсы, продукты и услуги на информационном рынке в информационном обществе

В информационном обществе акцент внимания и значимости смещается с традиционных видов ресурсов на информационный ресурс, который, хотя всегда существовал, не рассматривался ни как экономическая, ни как иная категория: никто специально о нем не говорил и тем более не вводил никаких определений. Одним из ключевых понятий при информатизации общества стало понятие *информационного ресурса*, толкование и обсуждение которого велось с того момента, когда начали говорить о переходе к информационному обществу. Этому вопросу посвящено довольно много публикаций, в которых отразились разные мнения и определения, а также разные научные школы, рассматривающие эти понятия. С принятием Федерального закона «Об информации, информатизации и защите информации» большая часть неопределенности была снята. Руководствуясь не научной стороной этого вопроса, а скорее прагматической позицией потребителя информации, целесообразно воспользоваться тем определением, которое приведено в этом законе. Тем более необходимо учитывать тот факт, что юридическое толкование во всех случаях является для потребителя информации опорой при защите его прав.

Определение

Информационные ресурсы – сочетание отдельных документов и отдельных массивов документов, которые сосредоточены на носителях информации и используются в информационных системах различного назначения: библиотеках, архивах, фондах, банках данных, территориально распределенных информационных системах.

Надо понимать, что документы и массивы информации, о которых говорится в этом законе, не существуют сами по себе. В них в разных формах представлены знания, которыми обладали люди, создавшие их. Таким образом, информационные ресурсы – это структурированные данные, которые подготовлены людьми для совместного использования в обществе и сохраненные на разных носителях информации.

Информационные ресурсы постиндустриального общества – структурированные данные и основа для поддержки формирования знаний в ходе обучающих процедур, отчуждены от людей, которые их создавали, накапливали, обобщали и т.п.

Структурированная информация и данные материализуются в виде документов (файлов), баз данных (БД), БЗ с информацией предоставленной экспертом, алгоритмов, программ для ЭВМ, а также произведений искусства, литературы, науки и техники.

Развитие мировых информационных ресурсов и цифровых автоматов позволило:

- превратить оказание информационных услуг в глобальную общественную деятельность посредством использования информационных технологий;
- сформировать мировой и внутригосударственный рынок информационных услуг;
- образовать информационные ресурсы и хранилища данных на уровне регионов и государств, к которым возможен открытый и недорогой доступ;
- повысить оперативность принимаемых решений в фирмах, банках, биржах, промышленности, торговле за счет своевременного использования данных.

Информационные ресурсы выступают основой для создания информационных продуктов и оказания информационных услуг на информационном рынке (рис. 1.1).

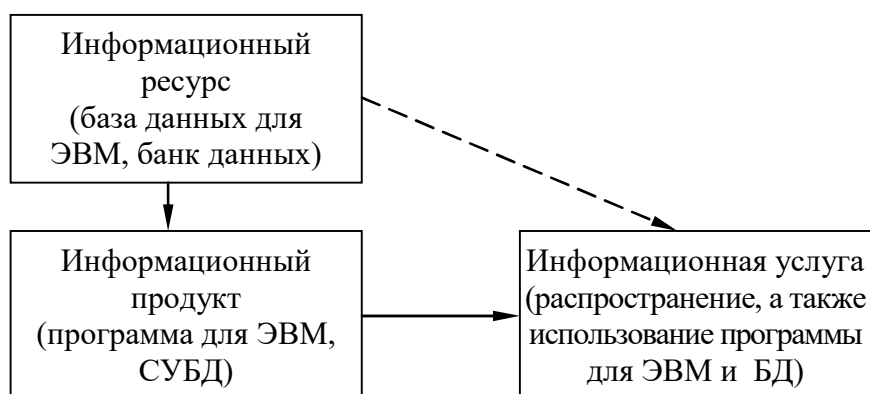


Рисунок 1.1. Оказание информационной услуги

на основе информационного ресурса и информационного продукта

Определенный информационный продукт разрабатывается согласно определенной маркетинговой стратегии, отражает имеющиеся технологии и концептуальную модель производителя, который ориентируется на один или несколько сегментов информационного рынка, а также на определенную группу потребителей и на их потребительские предпочтения. Свойства и уровень качества продукта определяются производителем на основе анализа конкурентной среды и контингента потребителей.

Информационный продукт выступает конечным результатом интеллектуальной деятельности человека или сотрудников организации, распространяется в виде документа, программы для ЭВМ или БД на носителе информации определенного типа: магнитный диск, оптический диск, электронный диск, файловый сервер в сети.

Определение

Информационный продукт – совокупность данных или код программы для ЭВМ, которые сформированы производителем для распространения в вещественной или невещественной форме на носителе информации определенного типа.

Информационный продукт и информационный ресурс выступают основой для оказания информационной услуги ее потребителю, поэтому они распространяются аналогичными способами по сравнению с традиционными продуктами, которые получены посредством использования технологии материального производства.

Определение (общее)

Услуга – результат непроизводственной деятельности предприятия (ЮЛ) или человека (ФЛ), направленный на удовлетворение потребностей определенного круга потребителей ЮЛ или ФЛ, оказывается на возмездной или безвозмездной основе.

Определение (частное)

Информационная услуга – получение и предоставление в распоряжение конечного потребителя или пользователя информационных ресурсов и продуктов.

Предоставлением информационных услуг занимаются франчайзинговые сети, которые занимаются распространением информационных продуктов (1С, Парус) и обеспечивают возможность доступа к информационным ресурсам (Гарант).

Информационные услуги бывают разных видов и назначения (рис. 1.2).

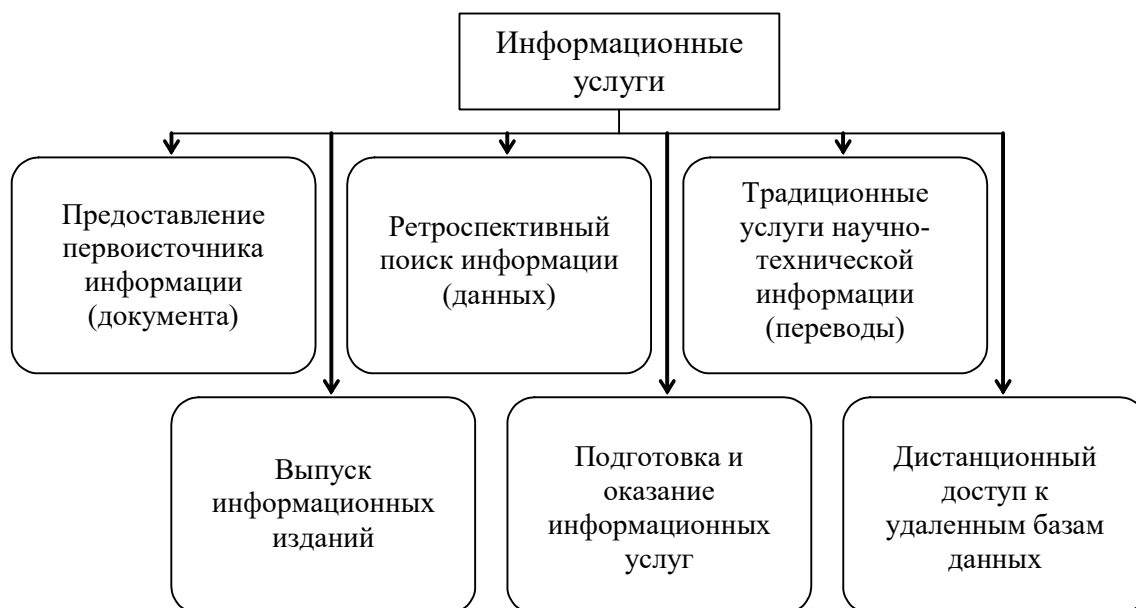


Рисунок 1.2. Виды информационных услуг

Реализация и распространение информационных ресурсов, продуктов и услуг между производителями и потребителями происходит на информационном рынке.

Информационный рынок выступает регулятором цены спроса и цены предложения, что позволяет установить равновесную цену на продукцию информационной индустрии, которая удовлетворяет одновременно производителя и потребителя.

Определение

Информационный рынок или рынок информационных ресурсов, продуктов и услуг – система экономических и правовых отношений, направленных на организацию торговли продуктами интеллектуального труда на возмездной основе.

Информационный рынок характеризуется определенной номенклатурой информационных ресурсов, продуктов и услуг, а также условиями и механизмами их предоставления, в частности ценообразованием, обеспечивающим формирование свободной рыночной цены, которая удовлетворяет производителя и потребителя.

На информационном рынке в качестве предмета купли или продажи, а также обмена выступают информационные системы, информационные технологии, лицензии, патенты, товарные знаки, ноу-хау, инженерно-технические услуги, информация различного рода и прочие виды информационных ресурсов, которые имеют материально-вещественное выражение или форму: документ, файл, БД, программа для ЭВМ.

Основным источником информации и актуальных данных для информационного обслуживания в современном постиндустриальном обществе являются БД, банки данных, хранилища информации, которые интегрируют в своей основе поставщиков и потребителей информационных услуг, связи и отношения между ними, порядок и условия продажи и покупки информационных ресурсов, продуктов и услуг.

Инновации и достижения в области информационных и коммуникационных технологий публикуются в открытой печати, а также распространяются посредством прочих средств массовой информации: радиовещание, телевидение, сеть Интернет.

В зависимости от вида обращаемых информационных ресурсов, продуктов и услуг на информационном рынке выделяют ряд основных сегментов или секторов:

1. Деловая и финансовая информация – включает несколько секторов:

- биржевая и финансовая информация – котировки ценных бумаг, валютные курсы, учетные ставки в банках, котировки фондового рынка, инвестиции и инновации, а их поставщиками являются специальные службы биржевой и финансовой информации, брокерские компании, биржи, банки;
- статистическая информация – ряды, тенденции, динамики, прогнозные модели и оценки по политической, экономической, социальной, демографической, научной, образовательной областям, а их создателями являются государственные службы, финансовые компании, консалтинговые фирмы;
- коммерческая информация – данные по компаниям, фирмам, корпорациям, направлениям работы и выпускаемой ими продукции, отпускным ценам;
- аналитические данные о финансовом состоянии фирм – перечень сделок, руководителей, деловые новости и изменения в области экономики и бизнеса, а их источниками являются специальные информационные службы;
- бухгалтерская информация – документы, которые регламентируют методику, правила и формы ведения учета финансово-хозяйственных операций.

2. Информация для квалифицированных специалистов – состоит из секторов:

- научно-техническая информация – документация, библиографическая, рефераты, монографии и обзоры, справочники по естественным, техническим, медицинским, общественным наукам, которые дифференцированы по отраслям производства и разным сферам человеческой деятельности;
- профессиональная и специальная информация – данные и информация, предназначенная для политиков, юристов, врачей-практиков, фармацевтов, преподавателей, инженеров, геологов, метеорологов, военных и т.д.;
- открытый доступ к первоисточникам – организация возможности просмотра источников информации через сеть библиотек, выставочные центры, образовательные учреждения, научные центры и службы доступа;
- данные для реализации источников – поиск информационных ресурсов на информационном рынке, предоставление прайс-листов и мест продажи.

3. Повседневная информация – состоит из нескольких секторов рынка:

- новости и литература – информация служб новостей, рекламных агентств, прессы, электронные журналы, справочники и энциклопедии;
- транспортная информация – расписание движения транспорта, заказ и резервирование билетов, бронирование мест в гостиницах, заказ товаров и услуг, список территориальных банковских и медицинских учреждений;
- рекламная информация – информация о реализуемых организациями сферы сбыта товарах, оказываемых работах и услугах, а также их цены;
- развлекательная информация – игры, музыка, телетекст, аудиотека.

- 4. Образовательные услуги** – охватывает все формы и ступени образования:
- дошкольное образование – перечень яслей-садов, детских садов и их особенности: профиль, специализация, а также их контактная информация;
 - школьное – информация об учреждениях среднего (общего) образования;
 - специальное – данные о техникумах и профессиональных училищах;
 - высшее образование – информация о высших образовательных учреждениях, научно-исследовательских институтах, научных центрах и библиотеках;
 - повышение квалификации и профессиональная переподготовка – перечень научных и образовательных учреждений, их специализация, образовательные программы, курсы обучения с указанием специальности и специализации.
- 5. Научная и образовательная печатная продукция** – включает секторы:
- научно-публицистические журналы, газеты, книги разного назначения;
 - учебники, методические пособия, лабораторные практикумы, конспекты;
 - научные монографии, аналитические обзоры, сборники докладов и статьи.
- 6. Аппаратное, программное и алгоритмическое обеспечение** – содержит:
- развивающие, обучающие и контролирующие компьютерные программы, технологии и методики обучения, средства автоматизации процесса обучения;
 - программные комплексы для начинающих и профессиональных пользователей ЭВМ – системное программное обеспечение (ПО), прикладное программное обеспечение, диагностические программы и утилиты, программы для автоматизации документооборота, информационных процессов и деятельности персонала в определенной предметной области, программы для автоматизации процесса математической обработки апостериорных данных статистическими методами, программные средства поддержки конструирования, моделирования, аналитически-численных расчетов;
 - аппаратное обеспечение и технические средства – платы расширения, ЭВМ, телекоммуникационное оборудование, оргтехника, датчики и измерительная аппаратура, а также расходные материалы и комплектующие;
 - внедрение и сопровождение информационных систем и технологий – анализ подразделений организации в целях выявления информационных потоков, разработка концептуальных моделей и структур программных компонентов в основе архитектуры информационных систем, создание и сопровождение БД, банков данных и хранилищ информации организации.
- 7. Консультационные услуги и сопровождение информационных систем:**
- консультирование по вопросам структуры продуктов информационной индустрии – закупка средств автоматизации, выбор программного обеспечения для поддержки профессиональной деятельности, особенности создания архитектуры информационных систем, подбор информационной технологии для организации уставной деятельности организации;
 - подготовка источников информации – создание тематических ресурсов и БД.

1.4. Правовые аспекты регулирования отношений на информационном рынке

Развитие рыночных отношений в информационном обществе поставило вопрос о необходимости выработки ряда мер для защиты информации как объекта интеллектуальной собственности, а также следующих из этого авторских и смежных прав.

В РФ принят ряд законов и постановлений, которые регулируют правоотношения между разными субъектами и объектами авторского права и смежных прав:

- Федеральный закон от 20.02.1995 г. №24-ФЗ «Об информации, информатизации и защите информации» (принят Государственной Думой РФ 25 января 1995 г., в ред. от 10.01.2003 г. №15-ФЗ, который позже утратил свою силу);
- Федеральный закон от 27.07.2006 г. №149-ФЗ «Об информации, информационных технологиях и о защите информации» (принят Государственной Думой Федерального Собрания РФ 08.07.2006 г.);
- Закон РФ от 09.07.1993 г. №5351-1 (ред. от 20.07.2004 г.) «Об авторском праве и смежных правах» (документ утрачивает силу с 1 января 2008 г. в связи с принятием Федерального закона от 18.12.2006 г. №231-ФЗ);
- Федеральный закон РФ от 18.12.2006 г. №231-ФЗ «О введении в действие части четвертой Гражданского кодекса РФ» (принят Государственной Думой РФ 24 ноября 2006 г., одобрен Советом Федерации РФ 8 декабря 2006 г.);
- Закон РФ от 23.09.1992 г. №3520-1 «О товарных знаках, знаках обслуживания и наименованиях мест происхождения товаров» (документ утратил силу с 1 января 2008 г. в связи с принятием Федерального закона от 18.12.2006 г. №231-ФЗ);
- Закон РФ от 23.09.1992 г. №3523-1 «О правовой охране программ для ЭВМ и БД» (в ред. Федеральных законов РФ от 24.12.2002 г. №177-ФЗ, от 02.11.2004 г. №127-ФЗ, от 02.02.2006 г. №19-ФЗ);
- Гражданский кодекс РФ: Раздел VII «Права на результаты интеллектуальной деятельности и средства индивидуализации», Глава 74 «Право на топологии интегральных микросхем»: ст. 1449 «Права на топологию интегральной микросхемы»; ст. 1450 «Автор топологии интегральной микросхемы»; ст. 1451 «Соавторы топологии интегральной микросхемы»; ст. 1452 «Гос. регистрация топологии интегральной микросхемы»; ст. 1453 «Право авторства на топологию интегральной микросхемы»; ст. 1454 «Исключительное право на топологию»; ст. 1455 «Знак охраны топологии интегральной микросхемы»; ст. 1456 «Действия, не являющиеся нарушением исключительного права на топологию»; ст. 1457 «Срок действия исключительного права на топологию»; ст. 1458 «Договор об отчуждении исключительного права на топологию»; ст. 1459 «Лицензионный договор о предоставлении права использования топологии интегральной микросхемы»; ст. 1460 «Форма и государственная регистрация договора об отчуждении исключительного права на топологию и лицензионного договора»; ст. 1461 «Служебная топология»; ст. 1462 «Топология, созданная при выполнении работ по договору»; ст. 1463 «Топология, созданная по заказу»; ст. 1464 «Топология, созданная при выполнении работ по государственному или муниципальному контракту».

Рассмотрим основные положения Федерального закона от 27 июля 2006 г. №149-ФЗ «Об информации, информационных технологиях и защите информации» (принят Государственной Думой РФ 8 июля 2006 г., одобрен Советом Федерации РФ 14 июля 2006 г.), который является базовым юридическим документом, открывающим путь к принятию дополнительных нормативных законодательных актов для успешного развития информационного общества в перспективе. С его помощью удалось частично решить вопросы правового регулирования на информационном рынке и ряд проблем: защиты прав и свобод личности от угроз и ущерба, связанного с искажением, порчей, а также уничтожением «персональной» информации.

Представленный Федеральный закон состоит из 18 различных статей:

- сфера действия Федерального закона;
- основные понятия, используемые в Федеральном законе;
- принципы правового регулирования отношений в сфере информации, информационных технологий и защиты информации;
- законодательство Российской Федерации об информации, информационных технологиях и о защите информации;
- информация как объект правовых отношений;
- обладатель информации;
- общедоступная информация;
- право на доступ к информации;
- ограничение доступа к информации;
- распространение информации или предоставление информации;
- документирование информации;
- гос. регулирование в сфере применения информационных технологий;
- информационные системы;
- государственные информационные системы;
- использование информационно-телекоммуникационных сетей;
- защита информации;
- ответственность за правонарушения в сфере информации, информационных технологий и защиты информации;
- о признании утратившими силу отдельных законодательных актов (положений законодательных актов) Российской Федерации, в частности:
 - Закон РФ от 20 февраля 1995 г. №24-ФЗ «Об информации, информатизации и защите информации» (Собрание законодательства РФ, 1995, №8, ст. 609);
 - Закон РФ от 4 июля 1996 г. №85-ФЗ «Об участии в международном информационном обмене» (Собрание законодательства РФ, 1996, №28, ст. 3347);
 - ст. 16 Закона РФ от 10 января 2003 г. №15-ФЗ «О внесении изменений и дополнений в некоторые законодательные акты РФ в связи с принятием Федерального закона "О лицензировании отдельных видов деятельности"» (Собрание законодательства Российской Федерации, 2003, №2, ст. 167);

- ст. 21 Федерального закона от 30 июня 2003 г. №86-ФЗ «О внесении изменений и дополнений в некоторые законодательные акты Российской Федерации, признании утратившими силу отдельных законодательных актов Российской Федерации, предоставлении отдельных гарантий сотрудникам органов внутренних дел, органов по контролю за оборотом наркотических средств и психотропных веществ и упраздняемых федеральных органов налоговой полиции в связи с осуществлением мер по совершенствованию государственного управления» (Собрание законодательства Российской Федерации, 2003, №27, ст. 2700);
- ст. 39 Федерального закона от 29 июня 2004 г. №58-ФЗ «О внесении изменений в некоторые законодательные акты Российской Федерации и признании утратившими силу некоторых законодательных актов Российской Федерации в связи с осуществлением мер по совершенствованию государственного управления» (Собрание законодательства Российской Федерации, 2004, №27, ст. 2711).

Закон регламентирует основы разработки и реализации государственной политики в области информации, информационных и коммуникационных технологий, защиты информации, а также авторских прав и возникающих смежных прав.

Информатизация разных сфер деятельности общества рассматривается в рамках государственных целевых программ развития науки, техники и технологий, обороны, медицины, образования, а также отражается в государственной научно-технической и промышленной политике развития РФ в долгосрочной перспективе.

Закон создает условия для включения России в международную информационную среду использования информационных ресурсов, продуктов и услуг, а также:

- предотвращает бесхозяйственное отношение к информационным ресурсам, средствам автоматизации, информационной безопасности в обществе;
- закрепляет основу правоотношений ЮЛ и ФЛ на информацию как объект интеллектуальной собственности, как на объект распространения и использования;
- обеспечивает основу для разработки нормативных актов, постановлений правительства, целевых программ развития единой информационной среды для научно-исследовательских и образовательных учреждений, которые обеспечивают комплексное решение проблемы организации информационных ресурсов, создают подходы к информатизации и внедрению средств автоматизации.

В рамках правового поля действующего законодательства «информация» рассматривается как информационный продукт, который сохранен на машинном носителе в вещественной форме, подлежит распределению и использованию в среде ее потребления определенным или неопределенным кругом лиц, а также выступает объектом интеллектуальной собственности между субъектами авторских (автор, соавторы) и возникающих смежных прав (заявитель, патентообладатель, лицензиар как лицо выдающее определенную лицензию, лицензиат как номинальный держатель полной или неполной лицензии на право использования программы для ЭВМ и БД).

В Законе РФ от 27.07.2006 г. №149-ФЗ «Об информации, информационных технологиях и защите информации» используются следующие основные понятия:

- **информация** – сведения (сообщения, данные) независимо от формы их представления;
- **информатизация** – процессы, методы поиска, сбора, хранения, обработки, предоставления, распространения информации и способы осуществления таких процессов и методов;
- **информационная система** – совокупность содержащейся в БД информации и обеспечивающих ее обработку информационных технологий и технических средств;
- **информационно-телекоммуникационная сеть** – технологическая система, предназначенная для передачи по линиям связи информации, доступ к которой осуществляется с использованием средств вычислительной техники;
- **обладатель информации** – лицо, самостоятельно создавшее информацию либо получившее на основании закона или договора право разрешать или ограничивать доступ к информации, определяемой по каким-либо признакам;
- **доступ к информации** – возможность получения информации и ее использования;
- **конфиденциальность информации** – обязательное для выполнения лицом, получившим доступ к определенной информации, требование не передавать такую информацию третьим лицам без согласия ее обладателя;
- **предоставление информации** – действия, направленные на получение информации определенным кругом лиц или передачу информации определенному кругу лиц;
- **распространение информации** – действия, направленные на получение информации неопределенным кругом лиц или передачу информации неопределенному кругу лиц;
- **электронное сообщение** – информация, переданная или полученная пользователем информационно-телекоммуникационной сети;
- **документированная информация** – зафиксированная на материальном носителе путем документирования информация с реквизитами, позволяющими определить такую информацию или в установленных законодательством Российской Федерации случаях ее материальный носитель;
- **оператор информационной системы** – гражданин или юридическое лицо, осуществляющие деятельность по эксплуатации информационной системы, в том числе по обработке информации, содержащейся в ее базах данных.

Ратификация представленного закона РФ и обеспечение выполнения его положений гарантируют, что государство получит значительную экономию финансовых средств и необходимые условия для устойчивого развития экономики и построения демократического общества в Российской Федерации на пороге XXI века.

1.5. Генезис и развитие информатики

Термин **информатика** возник в 60-х гг. во Франции для именовании научной области, которая занимается автоматизированной обработкой информации посредством ЭВМ. Французский термин *informatique* (информатика) образован путем слияния слов **informatio** (информация) и **automatique** (автоматика) и означает «информационная автоматика или автоматизированная переработка информации».

В англоязычных странах этому термину соответствует синоним “**computer science**” – наука о компьютерной технике, а позже возникает теория информации.

Выделение информатики как самостоятельной области человеческой деятельности в первую очередь связано с развитием компьютерной техники. Причем основная заслуга в этом принадлежит основам алгоритмизации и программирования, микропроцессорной технике, появление которых в середине 70-х гг. послужило началом четвертого этапа эволюции аппаратного обеспечения в основе архитектуры ЭВМ. На этом этапе развития элементной базы архитектуры вычислительных машин становятся платы многослойного печатного монтажа, микросхемы и микропроцессорная техника, а область, связанная с созданием и использованием ЭВМ получила мощный импульс в своем развитии: появляются предпосылки ЭВМ с распределенной архитектурой.

Термин «**информатика**» приобретает новое дыхание и используется не только для отображения достижений в области компьютерной техники и информационных технологий, но связывается с процессами получения, передачи и обработки информации.

В нашей стране подобная трактовка термина «**информатика**» утвердилась с момента принятия решения в 1983 г. на сессии годовичного собрания Академии наук СССР об организации нового отделения информатики, вычислительной техники и автоматизации. Информатика трактовалась как «комплексная научная и инженерная дисциплина, изучающая все аспекты разработки, проектирования, создания, оценки, функционирования систем переработки информации основанных на ЭВМ, их применение и степень влияния на различные области социальной практики».

Информатика в таком понимании нацелена на разработку общеметодологических принципов построения функциональных и концептуальных моделей информационных потоков и распределенных информационных систем. Фундаментальные и прикладные методы информатики применяются повсеместно, но при существовании возможности описания объекта, процесса или явления с помощью моделирования, которое позволяет построить модель – намеренно структурно или функционально обедненный эквивалент объекта, отражающий динамику его функционирования в определенной локальности для выявления новых закономерностей.

При рассмотрении предмета информатики принято выделять теоретическую и прикладную, фундаментальную, отраслевую и межотраслевую теорию информации.

Существует множество определений информатики, что связано с многогранностью ее функций, задач, подходов, средств и методов. Однозначной интерпретации определения «информатика» в опубликованных литературных источниках нет.

1.6. Предмет информатики

Информатика охватывает области, связанные с проектированием, реализацией, использованием и материально-техническим обеспечением (обслуживанием) систем обработки информации, включая машины, оборудование, организационное и технологическое обеспечение, а также средства автоматизации разного назначения.

Очевидно, что предмет «информатика» (рис. 1.3) и «структура информатики», по мнению академика РАН А.А. Дородницына (рис. 1.4) позволяют говорить о ней как о междисциплинарной теории информационного взаимодействия, которая может возникнуть, когда самостоятельным объектом наблюдения (исследования, изучения) оказываются особенности (свойства) самого процесса передачи информации (сведений о динамике развития той или иной ситуации, объекта, процесса или явления), содержание конкретных этапов и фаз обмена информацией. В соответствии с этим структура информатики различна в зависимости от вкладываемого содержания.

Концептуально «информатика» как предмет изучения имеет большое количество определений в современной научно-технической и образовательной литературе. Это происходит оттого, что данная область научных знаний относительно новая и динамичная, а имеющийся терминологический аппарат постоянно совершенствуется.

В результате анализа существующих определений выделим некоторые из них.

Информатика – наука, изучающая закономерности информационных процессов.

Информатика – наука о методах и средствах обработки информации и способах решения различных задач посредством использования ЭВМ в разных областях.

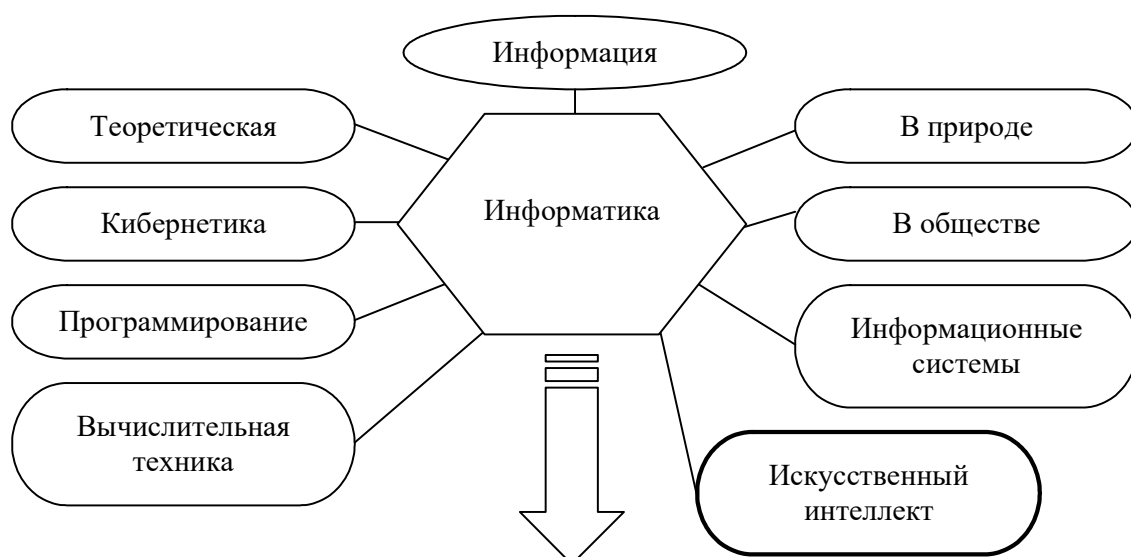
Информатика – это межотраслевая научная дисциплина, которая изучает подходы к информатизации производственных и непроизводственных сфер деятельности, методы и средства автоматизации на основе новых информационных технологий (аппаратное, программное и алгоритмическое обеспечение), а также инфраструктуру информационного рынка, производство средств сбора и обработки информации, структуру информационных процессов различного рода и особенности технологического использования информации и данных в определенных предметных областях.

Анализ существующих и представленных выше определений позволил выделить их сущность и сформулировать то определение, которое приведено ниже.

Определение

Информатика – это междисциплинарная наука и область человеческой деятельности, которая связана с процессами преобразования информации с помощью цифровых автоматов (ЭВМ) и прочих средств вычислительной техники, рассматривается специалистами как: фундаментальная наука, отрасль производства и прикладная дисциплина, а также совокупность определенных средств, используемых для поддержки процесса преобразования информации выраженной в форме данных.

Также существует большое количество определений в зарубежных литературных научно-публицистических источниках (газетах, журналах, брошюрах, монографиях, книгах, конспектах лекций, научных отчетах), которые по сущности и содержанию дублируют представленные выше определения предмета «информатика».



Объектами исследования информатики служат методы, средства и технологии, используемые для сбора, систематизации, хранения, поиска и распространения (передачи) информации по спектру предметных областей, а также особенности ее использования различными категориями потребителей. Сегодня образовалась новая планетарная структура – «инфосфера» (Ершов А.П.), от которой в значительной степени зависят темпы развития науки и технологии, а также эффективность всей системы информационного обеспечения и коммуникаций в обществе. Информатика в том или ином виде оптимизирует деятельность общественных единиц

Важнейшие идеи и методы информатики

Систематизация и индексирование документов и объектов с соответствующими характеристиками: ключевые слова, дескрипторы и другие параметры

Разработка информационных ресурсов, баз данных, организация хранилищ данных и глубокое структурирование массивов в информационно-поисковых системах

Организация средств репрезентации данных и формирования знаний, систем автоматического перевода носителей информации, оптимизация транзакционного обмена

Полностью индивидуально-ориентированный интерактивный диалоговый гибкий и открытый доступ к информационным ресурсам, продуктам и услугам

Ранжирование документов по степени актуальности и релевантности критериям отбора, интеллектуализация информационного поиска и обмена данными между потребителями

Использование многомерного шкалирования, кластерного анализа и других статистических методов для классификации информационных ресурсов

Выявление закономерностей функционирования информационных систем, а также проверка фундаментальных и прикладных основ теории информации

Рисунок 1.3. Объект, важнейшие идеи и методы современной информатики как междисциплинарной теории информационного взаимодействия

Генезис и развитие информатики позволяют говорить о нечеткости интерпретации понятий «информатика» и «кибернетика», поскольку они практически не отличались на первых этапах становления теории информации как научной дисциплины, но сегодня существует множество попыток объяснения их существенных сходств и различий, которые обусловлены дроблением современной теории информации, инициированным минимизацией объекта и предмета исследования.

Следует отметить, что основная концепция, заложенная Н. Винером в кибернетику, связана с разработкой положений кибернетики, программирования и основ теории управления сложными динамическими объектами и системами в разных областях социальной активности и продуктивной деятельности. Кибернетика существует независимо от наличия или отсутствия цифровых автоматов и ЭВМ.

Кибернетика – это наука об общих принципах управления в разных системах: технических, биологических, социальных, экономических, политических и прочих.

Информатика занимается изучением процессов поиска новой и преобразования накопленной информации более широко, но практически не решает задачи мониторинга и управления разными классами объектов, в отличие от кибернетики. Сейчас информатика представляется как фундаментальная многопрофильная дисциплина по отношению к кибернетике. Информатика обеспечивает поддержку решения разных проблем, которые связаны с использованием компьютерной техники, в частности внедрением средств автоматизации процессами и производствами. Разработкой алгоритмического и аппаратного обеспечения для автоматизации сборочных конвейеров и линий занимается соответственно кибернетика и робототехника.

На ранней стадии развития теории информации между кибернетикой и робототехникой провести четкую границу мог только квалифицированный специалист, поскольку объект и предмет исследования выступает смежным, а также отсутствовали технологии, которые обеспечивали бы возможность реализации аппаратного обеспечения для робототехнических комплексов. Обе науки образованы в результате дробления информатики на пути становления современной теории информации.

Информатика появилась благодаря развитию алгоритмического (алгебра логики и функциональные схемы логических устройств) и программного (средства и языки программирования) обеспечения информационных систем, а также вычислительной техники при появлении возможности аппаратной реализации их элементов.

Кибернетика и информатика – это дисциплины, которые имеют общие сходства, а их различия проявляются в процессе становления теории информации:

- в информатике – отдается приоритет свойствам информации, изучению закономерностей информационного обмена в различных предметных областях, а также разработке алгоритмов, программ и устройств для обработки данных;
- в кибернетике – выделяется разработка концепций и подходов к построению моделей сложных объектов с использованием аппарата алгебры логики, генетических и эвристических алгоритмов, имитирующих поведение человека.

1.7. Структура информатики

Информатика охватывает несколько основных направлений ее рассмотрения:

- фундаментальная наука – формирование ядра науки: теория и практика;
- отрасль производства – создание технологий и средств обработки данных;
- прикладная дисциплина – изучение закономерностей информационных процессов в разных предметных областях и сферах деятельности общества;
- средства автоматизации процессов, производств, исследований, разработок.

Поскольку теория информации выступает междисциплинарной наукой, то ее структура дифференцируется в зависимости от рассматриваемого содержания.



Рисунок 1.4. Структура дисциплины «информатика»

Средства автоматизации процессов в разных сферах деятельности общества:

- **аппаратные и технические средства (hardware)** – компьютеры и периферийные устройства разного назначения (мониторы, клавиатуры, модемы, принтеры и плоттеры), линии связи, аппаратура передачи данных, т.е. набор технических средств и расходных материалов, обеспечивающих передачу и преобразование информации под управлением информационных систем и ПЭВМ;
- **программные средства (software)** – локальные и сетевые операционные системы (ОС), интегрированные среды программирования на языках высокого уровня, системы проектирования, пакеты для аналитически-численных расчетов и статистического анализа, пакеты прикладных программ (текстовые и графические редакторы, финансовые и бухгалтерские программы, издательские системы, комплексы автоматизации документооборота и принятия решений);
- **математические модели, методы и алгоритмы (brainware)** – алгоритмы и процедуры, блок-схемы алгоритмов, вычислительные процессоры, которые построены по принципу централизованного, локального и супервизорного управления, а также с использованием параллельной и распределенной архитектуры.

Информатика как **фундаментальная наука** обеспечивает теоретические и экспериментальные исследования процессов обработки информации в разных отраслях и сферах деятельности человека, что позволяет формировать и уточнять фундаментальное ядро этой науки, а также разрабатывать соответствующие теории, модели, методы и алгоритмы, которые затем применяются для повышения эффективности функционирования информационных систем и внедрения средств автоматизации (аппаратного, программного и алгоритмического обеспечения).

Информатика как **отрасль производства** реализует практическое использование теоретически подтвержденных результатов исследований информатики как фундаментальной науки, что позволяет исследовать информационное взаимодействие в различных отраслях производственной деятельности общества, а также разрабатывать и внедрять унифицированные и специализированные средства автоматизации для повышения эффективности функционирования производственного цикла (единичное, серийное, массовое производство) в рамках использования технологий материального производства и информационных технологий: производство программных продуктов (Microsoft, Lotus, Borland), технических средств (IBM, Apple, Intel, Hewlett Packard) и технологий для оптимизации получения, обработки и передачи данных.

Информатика как **прикладная дисциплина** решает междисциплинарные задачи анализа и повышения эффективности функционирования информационных систем, закономерностей протекания информационных процессов в определенных предметных областях, методологий создания информационных систем и технологий, разработкой математического, программного и алгоритмического обеспечения ЭВМ, а также подготовкой и повышением квалификации специалистов в области получения, обработки и передачи информации, которая выражена в виде данных.

Аппаратное обеспечение классифицируется в зависимости от габаритных размеров, быстродействия, энергопотребления, частотных и вольтамперных характеристик в процессе функционирования в составе ЭВМ или информационной системы.

Программное обеспечение реализует автоматизацию выполнения ограниченного или унифицированного набора задач системного и прикладного характера, а применение определенного программного продукта специфично: в зависимости от концепции, набора используемых технологий и методов программирования в основе его программной реализации, которые заложены тем или иным производителем.

Алгоритмическое обеспечение позволяет представить алгоритмическую структуру разного вида и назначения в наглядной для последующей интерпретации форме: в частности, посредством использования набора условно-графических обозначений, которые используются разработчиками в основе блок-схемы алгоритма.

Любая вычислительная система нацелена на решение очень широкого круга задач по преобразованию информации и обработке данных, а для смены вида и набора решаемых задач посредством ЭВМ пользователю необходимо и достаточно:

- приобрести на информационном рынке аппаратное обеспечение, которое соответствует системным требованиям производителя алгоритмов и программ;
- собрать для решения набора задач пользователя требуемую конфигурацию ЭВМ, которая содержит аппаратные компоненты (системную плату, процессор, оперативную память, накопители информации на гибких, жестких и электронных дисках, клавиатуру, манипулятор типа мышь, тачпад, трекбол, джойстик) и платы расширения (видео-адаптер, аудио-карта, сетевая карта), периферийное оборудование (принтер, сканер, плоттер) соответствующие современной архитектуре компьютеров, предложенной ведущими производителями;
- сменить набор алгоритмического обеспечения (линейные, разветвленные и циклические алгоритмические структуры), а также его программную реализацию (системные, прикладные, диагностические программы и утилиты для анализа отказоустойчивости и эффективности функционирования аппаратуры).

Выполняя свою основную функцию, информатика решает ряд важных задач:

- исследует информационные процессы в технических и социальных системах;
- разрабатывает информационную технику и создает новые технологии обработки информации на основе результатов, которые получены в ходе исследования информационных процессов в различных предметных областях;
- решает научные и инженерные проблемы создания, внедрения и обеспечения эффективного использования компьютерной техники и технологий во всех сферах профессиональной деятельности, а также для домашнего использования.

В рамках информатики как прикладной дисциплины изучаются: понятие информации, ее свойства, измерение информации, использование в управлении; способы кодирования информации; понятие и составные части информационных процессов; организация технических устройств преобразования информации, в частности компьютера; структура и методология проектирования программного обеспечения.

1.8. Информация, ее представление и свойства

Согласно множеству определений информатики, ее функциям и задачам одним из ключевых понятий информатики является информация как объект преобразования.

Принято считать, что информация существует до и вне зависимости от наличия наблюдателя или технического средства ее регистрации (датчик), а также она передается посредством информационных сообщений от источника к приемнику.

Информация характеризуется набором свойств, которые дифференцируются на:

- внутренние свойства информации – присущи информации независимо от предметной области, в которой она используется и вида ее представления:
 - неоднородность – информация подлежит представлению в форме данных разного типа, что обуславливает необходимость разработки структур данных (инфологическая схема БД), которые выступают специфическими по отношению к определенной предметной области и сфере использования, поэтому для унификации хранения информации необходимо усовершенствовать существующие структуры данных и разрабатывать новые инфологические схемы БД повышающие эффективность доступа к данным;
 - неаддитивность – обусловлена отсутствием взаимной совместимости структур данных в основе информационных хранилищ и БД, невозможностью осуществлять прямое сохранение и копирование информации от первоисточника в случайное хранилище, а также из одного хранилища в другое без предварительной (первичной) обработки информации и представления ее в форме данных адекватно заданной инфологической схеме БД;
 - несоответствие формы представления и содержания информации – информация существует в виде сигналов согласно выбранному принципу передачи в канале связи и подлежит представлению в форме данных независимо от ее содержания, а содержание инвариантно ее представлению;
- внешние свойства информации – проявляются только в сфере ее потребления.

Внутренние и внешние свойства информации способствуют развитию новых информационных технологий и средств автоматизации, которые повышают эффективность процесса информационного обмена и потребительскую ценность (полезность) информации по отношению к предметной области ее использования.

Существует большое количество критериев классификации информации, которые унифицируют и дифференцируют рассмотрение ее сущности разными специалистами, которые заняты исследованием процесса информационного обмена.

Выделена структура процесса информационного взаимодействия, который включает набор этапов преобразования информации из сигнальной формы в структурированные определенным образом данные, а также представлена совокупность фаз информационного обмена сообщениями, обеспечивающими поддержку процесса передачи информации: регистрация сигнала, кодирование, передача кодовых последовательностей в виде сигналов, прием сигналов из канала связи, декодирование кодовых комбинаций и интерпретация данных конечным потребителем.

1.8.1. Понятие и характерные черты информации

На сегодняшний день строгое определение понятия «информация» отсутствует, но в российских и зарубежных литературных источниках многими ведущими учеными дается трактовка сущности информации в разных секторах этой науки.

Определение (латинско-русский словарь)

Информация (от латинского слова *informatio*) – разъяснение, представление, осведомление, обмен сигналами между субъектами естественного и искусственного происхождения, некоторые сведения, совокупность данных и знаний об объекте.

Определение (словарь С.И. Ожегова)

Информация – сведения об окружающем мире и протекающих в нем процессах и явлениях, воспринимаемые человеком или регистрируемые специальными техническими устройствами: датчиками и аналого-цифровыми преобразователями.

Определения (общие)

Информация – это сведения, снимающие неопределенность системы об объекте.

Информация – это то, что поступает в головной мозг человека из многих источников, во многих формах и при этом обеспечивает формирование структуры знаний.

Информация – это единственный и универсальный способ передачи данных, знаний, эмоциональных переживаний и волевых усилий между людьми в среде обитания.

Информация – это сущность, которая регистрируется и сохраняется при вычислимом изоморфизме определенного объекта исследования в имеющейся среде.

Определение

Информация – это сведения об окружающем мире и о всех объектах, процессах, явлениях, событиях, которые являются объектом преобразования при получении, хранении, передаче данных с целью последующего используются для выработки рекомендаций, управляющих воздействий, принятия решений или обучения.

Характерными чертами информации являются следующие:

- материальным носителем информации выступает механический, электрический, электромагнитный или оптический сигнал, который передается в различной среде посредством использования канала передачи данных на основе определенного принципа, алгоритма или протокола информационного обмена между определенными источниками и приемниками (конечными потребителями);
- стратегический ресурс информационной индустрии в информационном обществе;
- способствует появлению и развитию новых информационных технологий;
- обуславливает появление новых специальностей в разных сферах общества;
- является ресурсом неисчерпаемым при потреблении и использовании;
- придает дополнительную ценность другим ресурсам, в частности: материальным, информационным, социальным, финансовым, технологическим;
- информация может накапливаться на носителях разного рода: гибкие и жесткие магнитные диски, оптические и электронные диски и карты памяти;
- информация передается посредством аппаратуры передачи данных по каналу связи на базе достижений в области новых коммуникационных технологий.

1.8.2. Свойства информации

Предмет информатики как междисциплинарной теории информационного взаимодействия ориентирует заинтересованных специалистов на рассмотрение основных свойств информации как самостоятельного объекта изучения (рис. 1.5).

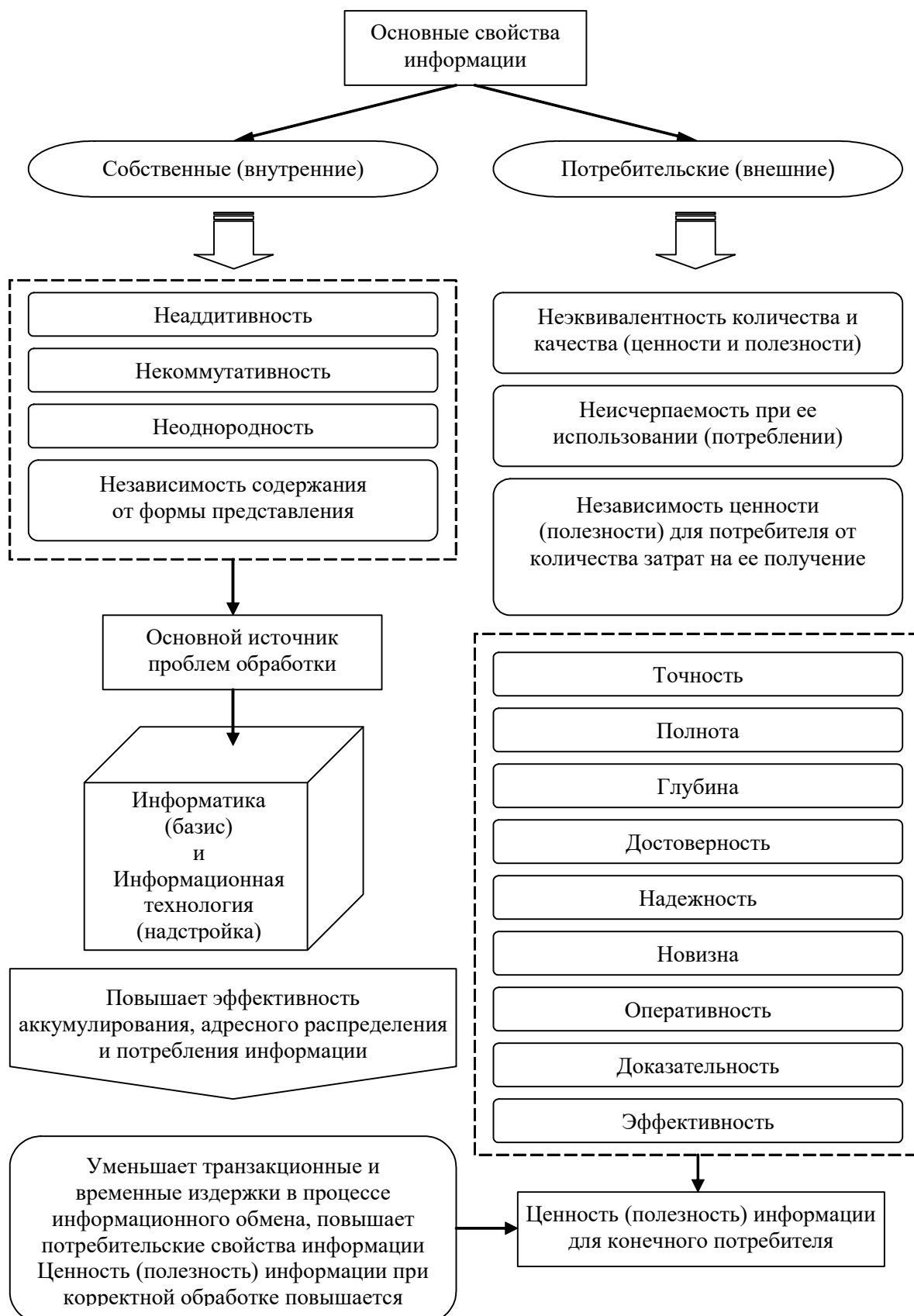


Рисунок 1.5. Основные свойства информации как объекта преобразования

1.8.3. Классификация информации

Классификацию информации выполняют по ряду оснований и критериев, набор которых не является строгим и может существенно варьировать, поскольку носит субъективную основу – мнения и точки зрения разных специалистов не совпадают:

1. По времени возникновения:

- **априорная** – заранее известна потребителю до получения очередного сигнала или информационного сообщения от источника информации;
- **апостериорная** – становится известной потребителю после получения сигнала, снимает неопределенность информационной системы после получения и обработки информационного сообщения от источника информации.

2. По стабильности (в частности во времени):

- **переменная (динамическая)** – отражает изменяющиеся (во времени) фактические характеристики (ключевые параметры) источника информации, которые отражают состояние определенного объекта исследования;
- **постоянная (статическая)** – условно-постоянная, многократно используемая совокупность структурированных данных и значений набора параметров, которые не изменяются в течение длительного периода времени, поскольку объект наблюдения не изменяет своего состояния.

3. По степени открытости:

- **открытая** – общедоступная и повсеместно используемая информация;
- **закрытая** – использование информации возможно только с разрешения определенных субъектов: ФЛ и ЮЛ или государственных органов.

4. По способу использования:

- **основная** – информационные фрагменты с основной частью данных;
- **вспомогательная** – дополнительные или необязательные сведения;
- **избыточная** – дублирует эффективные данные по каким-либо причинам.

5. По избыточности:

- **с естественной избыточностью** – данные, которые поступают от источника информации содержат систематически повторяющиеся сведения, которые характеризуют состояние объекта наблюдения и не увеличивают общую информативность накопленных апостериорных данных;
- **с искусственной избыточностью** – данные содержат расширенные информационные элементы, которые добавлены с целью достижения помехоустойчивости при передаче по каналу связи за счет ряда дополнительных свойств: обнаруживающие, исправляющие (корректирующие).

6. По области практического использования:

- **социальная** – обеспечивает удовлетворение социальных потребностей;
- **коммерческая** – финансовая (биржевая и банковская) информация;
- **научно-техническая** – реферативные журналы и аналитические обзоры.

1.8.4. Понятие и структура процесса информационного взаимодействия

При работе с информацией выделяют следующие базовые понятия: источник, потребитель (получатель), сигнал и канал информационного взаимодействия (обмена).

Определение

Источник информации – объект живой или неживой природы, либо техническое средство в определенной среде, генерирующий набор сигналов характеризующих параметры состояния, которые преобразуются в данные для целей дальнейшей обработки в процессе осуществления этапов информационного обмена.

Определение

Потребитель информации – элемент внешней среды естественного или искусственного происхождения, который использует информацию представленную в форме данных для дальнейшего преобразования и использования в предметной области.

Определение

Сигнал – это материальный носитель информации в рамках определенного физического принципа передачи волны или пучка (механический, электрический, электромагнитный, оптический), который обеспечивает перенос ее от источника к потребителю по каналу информационного обмена посредством технических средств.

Определение

Процесс информационного взаимодействия представляет собой совокупность этапов преобразования информации и фаз передачи данных от источника к конечному потребителю посредством использования канала информационного обмена (связи).

Любой процесс информационного взаимодействия является процессом передачи информационных сообщений, который состоит из отдельных этапов преобразования информации и фаз передачи данных: регистрация, подготовка (кодирование), передача, прием, расшифровка (декодирование), обработка и сохранение.

Важными характеристиками любой фазы обработки информации выступают:

- время преобразования информации при непосредственной передаче;
- актуальность и корректность при получении данных по каналу связи.

Время преобразования информации определяет потенциальное быстродействие информационной системы, а возникающие временные задержки приводят к старению информации и потере ее актуальности, что снижает ее ценность и может привести к катастрофическим последствиям, в частности: мониторинг, обеспечение управляемости и наблюдаемости реакцией термоядерного синтеза в реакторе.

Верность (корректность) преобразования информации определяет степень доверия к информации при принятии решений, а, следовательно, и эффективность функционирования любой информационно-вычислительной системы в целом.

Время и корректность преобразования информации определяются техническими характеристиками средств автоматизации, которые выполняют преобразование, а также внутренними и внешними свойствами непосредственно информации, в частности: несоответствие формы и содержания, достоверность, полнота, точность.

Наличие первичной и привнесенной избыточности в информации и данных снижает быстродействие информационной системы и обуславливает необходимость увеличения объема памяти запоминающих устройств, поэтому одной из важнейших задач является устранение первичной избыточности в информации и данных.

Искусственно привнесенная избыточность согласно определенному принципу или закономерности является весьма полезным средством борьбы с ошибками, которые неизбежно возникают вследствие зашумленности канала информационного обмена при передаче данных на различных этапах преобразования информации.

Основная задача при реализации фаз информационного обмена – обеспечение максимально возможного быстродействия при заданной верности информации.

Отвлекаясь от конкретной физической и технической реализации фазы преобразования информации решение этой задачи можно выполнить в два этапа:

- устранение исходной (вредной) избыточности в информации для уменьшения времени преобразования данных и объемов запоминающих устройств;
- внесение в данные специально-организованной избыточности обеспечивает заданный уровень помехоустойчивости для борьбы с ошибками, которые возникают при приеме, преобразовании (обработке) и передаче информации.

Реализация этих этапов связана с выбором методов кодирования информации.

Структуру процесса информационного взаимодействия отражает рис. 1.6.

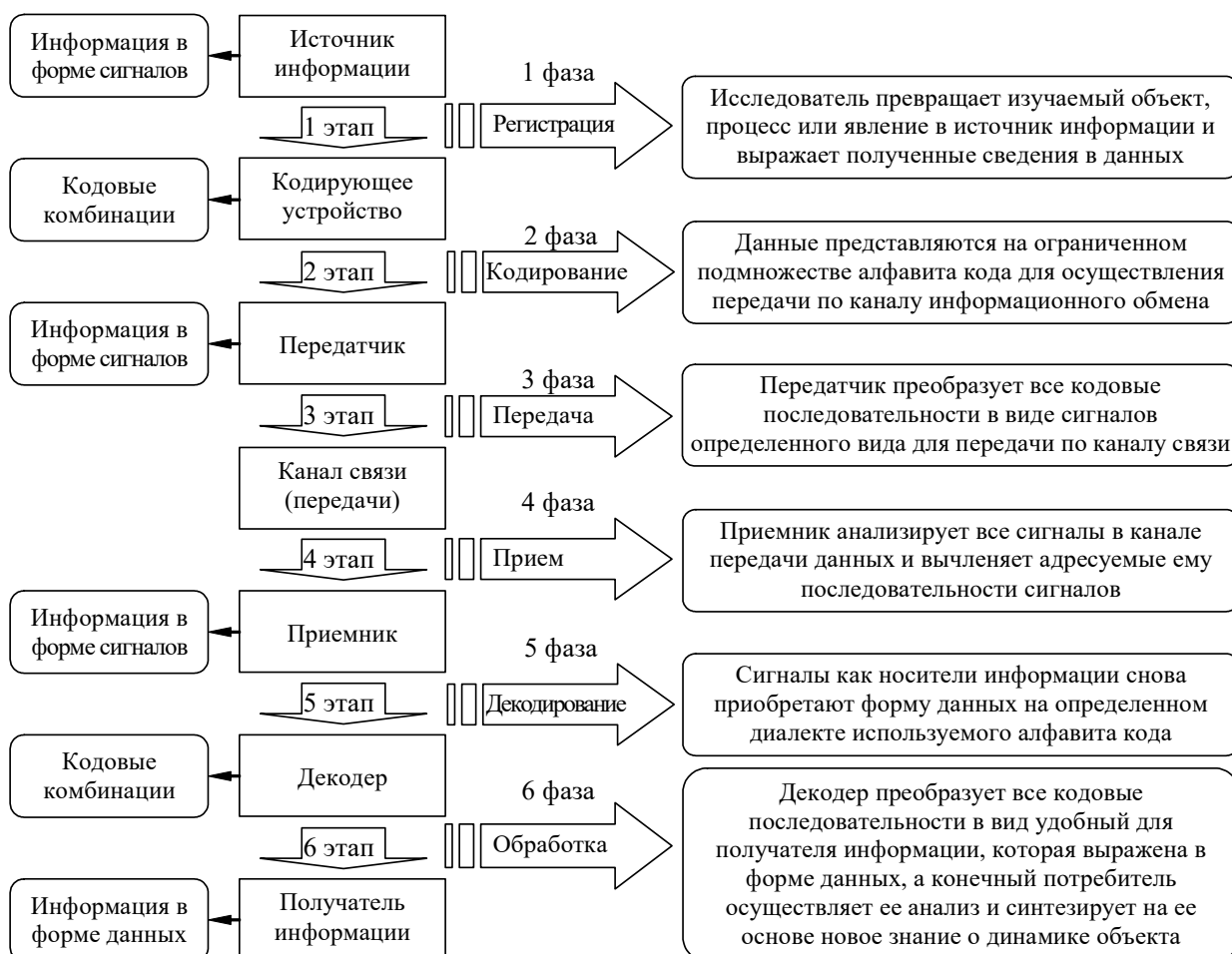


Рисунок 1.6. Обобщенная структура процесса информационного взаимодействия

1.9. Подходы к оценке количества информации

На данном этапе развития науки и технологии существует широкий спектр критериев классификации характеристик информации, которые варьируют относительно разных предметных областей и сфер использования, но все они базируются на основных подходах к феномену «информация»: количественных и качественных.

Существует несколько подходов к анализу количества информации и данных, среди которых выделим (рис. 1.7): структурный, семантический, статистический.

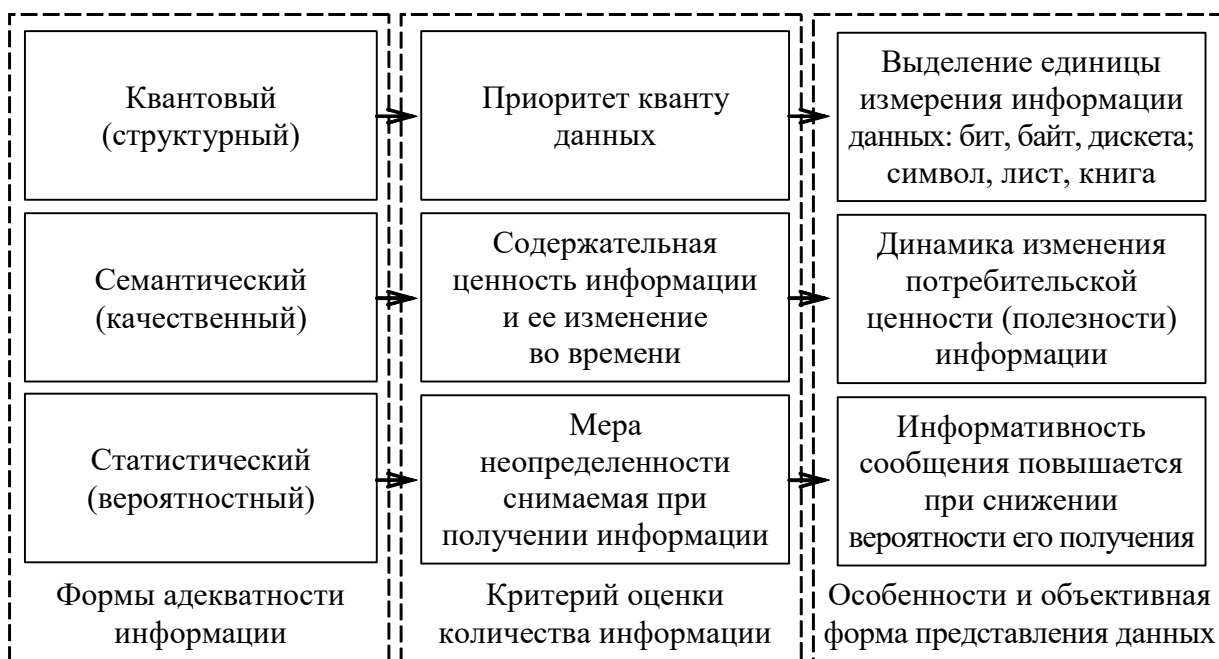


Рисунок 1.7. Подходы к оценке количества информации

При **структурном подходе** за единицу количества информации принимается некоторый квант данных и количество информации оценивается простым подсчетом. В вычислительной технике за единицу количества информации принимается байт как двоичная восьмиразрядная битовая кодовая последовательность. При хранении и передаче данных их объем измеряется количеством байт, которые содержатся, в частности, в файле на машинном носителе информации или передаются по линии (каналу) связи между источником (отправителем) и приемником (получателем).

В **семантической теории** учитывается в основном содержательная ценность информации (данных), рассматривается ее изменение во времени (старение информации) и влияние этих изменений на эффективность потребления (адекватность, достоверность и полнота описания структуры данных для организации управления в системе).

Статистическая теория оценивает информацию с точки зрения меры неопределенности, снимаемой при получении данных, она не затрагивает смыслового содержания информации, а основывается на вероятностных свойствах информационных элементов. Немаловажным фактом является **корреляционная связь между информативностью (количеством информации) некоторого события (сообщения) и вероятностью его возникновения**: «Чем реже событие (информационное сообщение), тем большую важность оно имеет для получателя, то есть содержит большую информативность по отношению к более частым сообщениям (событиям)».

Можно показать и математически строго доказать корректность указанного выше утверждения, как это сделали в свое время К. Шеннон, Н. Винер, Р. Хартли.

Пусть имеется множество X , содержащее M сообщений X_j ($j=1, 2, \dots, M$), которые возникают с вероятностями $p(X_j)$, при этом сумма всех вероятностей равна 1 ($\sum_j p(X_j) = 1$) ($0 < p(X_j) \leq 1$). Тогда, количество информации (информативность) I , которое содержится в определенном сообщении X_j , вычисляется по формуле: $I(X_j) = -\log_2 p(X_j)$ и интерпретируется посредством графической зависимости (рис.1.8).

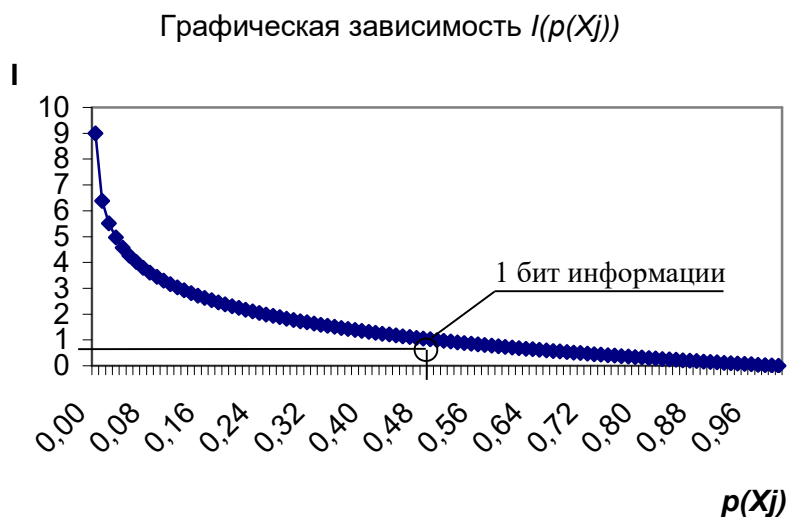


Рисунок 1.8. Связь между информативностью (количеством информации I) некоторого события (сообщения X_j) и вероятностью его возникновения $p(X_j)$, учитывая подстановку их значений: $I(X_j) = \{0, 1, 2, 3\}$, $p(X_j) = \{1/8, 1/4, 1/2, 1\}$

На основании представленной графической интерпретации, отражающей зависимость между информативностью (количеством информации I) некоторого события (сообщения) X_j и вероятностью его возникновения $p(X_j)$, можно сформулировать следующее определение 1 бита как наименьшей логически неделимой единицы измерения информации (данных), используемой в вычислительных системах.

Определение

За **один бит информации** принимают информативность (количество информации) такого информационного сообщения, вероятность появления которого равна 0,5.

Пример 1.1. Для интерпретации данного закона необходимо взять два листа бумаги одинакового формата, нанести на каждый из них отдельно цифры 0 – ноль и 1 – единица, а затем попросить человека вытянуть один из них произвольным образом.

Вероятность вытягивания листа бумаги с цифрой 0 или 1 равна 0,5, а информативность информационного сообщения содержащегося на каждом из них равна 1.

Существуют расширенные единицы измерения количества информации.

Определение

Одним байтом информации (данных) называется восемь бистабильных ячеек (бит).

Определение

1 Килобайт = 1024 байта, 1 Мбайт = 1024 Кбайта = 1048576 байт = 8388608 бит.

1.10. Основные виды сигнала как материального носителя информации

Материальным носителем информации является элементарный сигнал, который передается по каналу связи между источником и потребителем информации.

Для целей обработки информации в цифровом автомате (ЭВМ) сигнал представляется в виде кодовой комбинации и приобретает форму данных, которые располагаются в оперативной памяти и внешнем запоминающем устройстве (рис. 1.9).

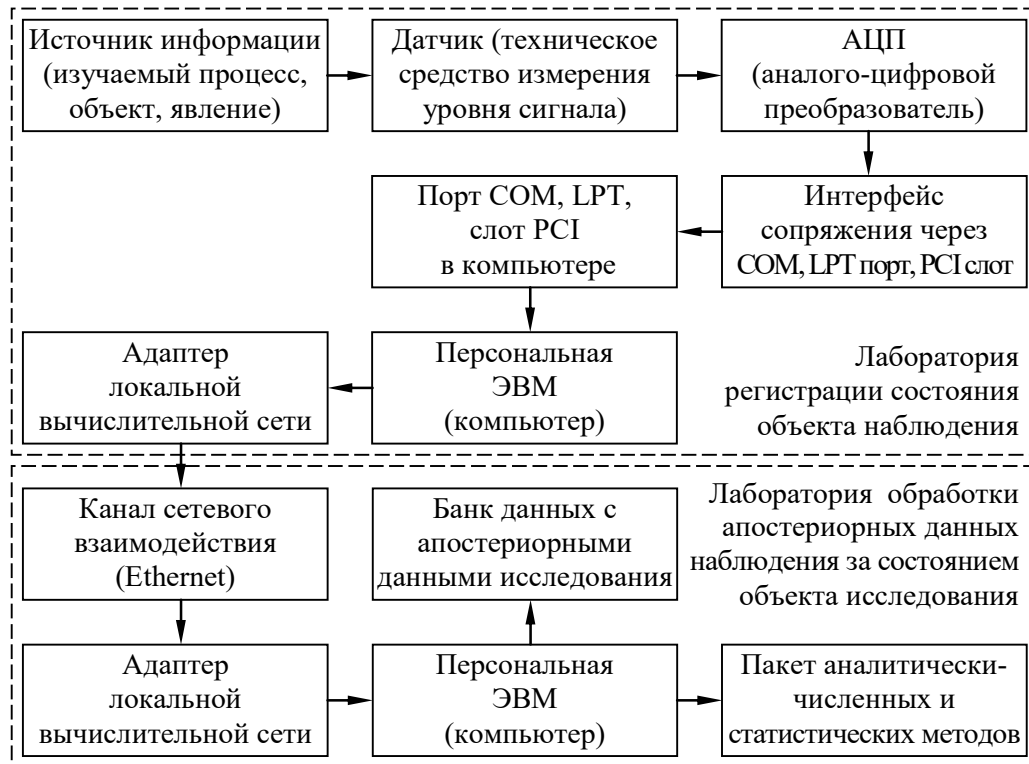


Рисунок 1.9. Особенности преобразования сигналов в данные на машинном носителе
Сигнал выступает материальным носителем информации в канале передачи данных:

- непрерывный сигнал – применяется в аналоговых информационных системах;
- дискретный сигнал – используется в импульсных и цифровых системах.

Определение

Непрерывный (аналоговый) сигнал аналитически задается функцией, которая непрерывна во времени, определяется на заданном интервале времени (ограниченном или бесконечномерном) и используется в аналоговых системах.

Примечание: на функцию накладывается ограничение – непрерывная дифференцируемость по крайней мере один раз с целью исключения разрывов первого и второго рода.

Определение

Дискретный сигнал состоит из счетного конечномерного множества информационных элементов (измерений), которые определены в дискретные моменты времени и используется в импульсных или цифровых системах обработки данных.

Дискретный сигнал лучше поддается преобразованиям в ЭВМ, поэтому имеет преимущества перед аналоговым, но на интервале между дискретными моментами времени измерения уровня сигнала нельзя однозначно определить значение его уровня.

Информация может существовать в различных формах и предназначена для разных целей при ее получении, обработке, передаче и адресном распределении:

- информация в форме сигналов – генерируется источником информации и измеряется посредством использования технических средств (датчиков) для ее обработки, а также продуцируется цифро-аналоговым преобразователем и подается на входы исполнительных устройств (регуляторов уровня, роботов);
- информация в форме данных – формируется аналого-цифровым преобразователем на основе измерения значений уровня сигнала в дискретные моменты времени для ее обработки посредством использования компьютера, а также генерируется цифровым автоматом (электронной вычислительной машиной) для последующей подачи кодовых последовательностей на вход цифро-аналогового преобразователя с целью регулирования и управления объектом.

На рис. 1.10 представлена схема, которая отражает различные формы существования информации в контуре мониторинга и управления определенного объекта.

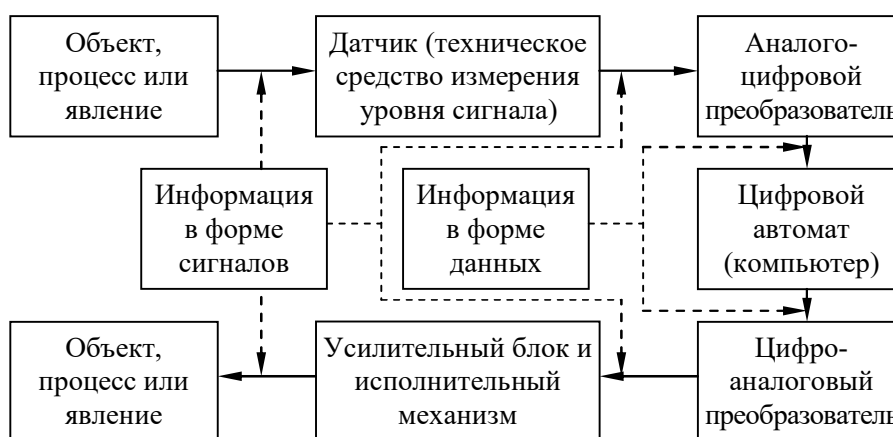


Рисунок 1.10. Объективные формы существования информации в контуре измерения и управления

Таким образом, взаимнооднозначное преобразование информации в форме сигналов и данных осуществляется посредством использования преобразователей.

Определение

Аналого-цифровым преобразователем называют техническое средство, которое является дискретным по принципу своего функционирования, обеспечивающее быстрое соотнесение динамически изменяющегося уровня непрерывного сигнала соответствующего определенной физической величине на его входе с последовательностью номинальных значений в виде кодовых комбинаций соответствующих измерениям исходного сигнала в дискретные моменты времени.

Определение

Цифро-аналоговым преобразователем называют техническое средство, которое является дискретным по принципу своего функционирования, обеспечивающее быстрое соотнесение последовательности номинальных значений измерений исходного сигнала в дискретные моменты времени на входе с множеством значений уровня сигнала соответствующего определенной физической величине.

1.10.1. Квантование непрерывного (аналогового) сигнала

Процедура квантования непрерывного сигнала позволяет получить его дискретный аналог для последующего преобразования посредством компьютера.

Определение

Для преобразования непрерывного аналогового сигнала в дискретный аналог используется процедура, которая называется **квантованием (дискретизацией)**.

Различают два вида квантования (дискретизации) – по времени и по уровню.

Определение

Квантование по времени – замена непрерывной по времени и по уровню функции $x(t)$ (рис. 1.11, а) некоторым множеством дискретных значений сигнала $x(t_i)$ – решетчатая функция, которая соответствует ряду мгновенных значений уровня сигнала измеренным в дискретные моменты времени (рис. 1.11, б), где $i=\{0,1,2,3,4\}$.

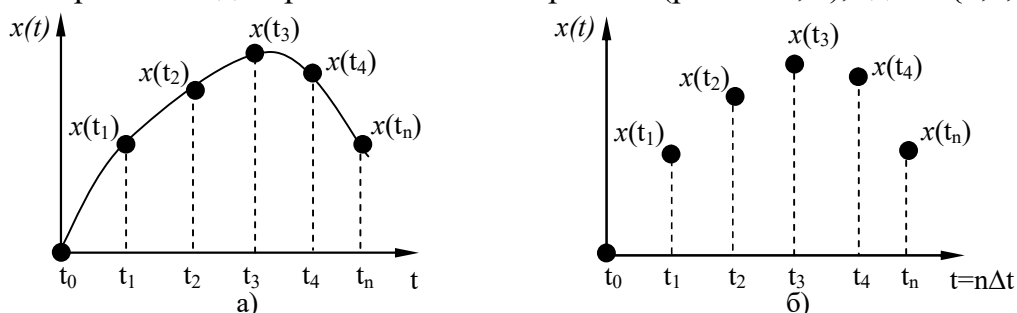


Рисунок 1.11. Квантование по времени:

- а) аналоговый сигнал $x(t)$ в дискретные моменты времени t_i до квантования;
- б) дискретный (по времени) сигнал $x(t_i)$ – результат квантования

Дискретизация связана с потерей данных, а дискретный сигнал (рис. 1.11, б) не отражает поведение исходного сигнала в промежуточные моменты времени, например, на интервале t_3 и t_4 . Дискретизация обуславливает появление некоторой погрешности ϵ – ошибка дискретизации, которая зависит от шага квантования $\Delta t = t_i - t_{i-1}$: при малых значениях шага квантования число точек замера большое, невязка между аналоговым и дискретным сигналами минимальна, а при увеличении шага дискретизации наоборот – существенно возрастает погрешность или ошибка дискретизации. Ошибка дискретизации $\epsilon(t)$ на интервале $[t_{i-1}, t_i]$ в определенный момент времени t_i определяется по формуле: $\epsilon(t) = x(t_{i-1}, i) - v(t_{i-1}, i)$, где $v(t)$ – функция восстановления, которая по множеству дискретных значений на интервалах восстанавливает аналог $x(t)$.

Виды дискретизации различаются по регулярности осуществления отсчетов:

- **равномерная дискретизация** – период квантования постоянный ($\Delta t = \text{const}$);
- **неравномерная дискретизация** – период дискретизации меняется ($\Delta t = \text{var}$), причем этот вид дискретизации, в свою очередь, делится на два подвида:
 - **адаптивная** – Δt изменяется автоматически в зависимости от текущего уровня измеренного сигнала: шаг дискретизации увеличивается при уменьшении ошибки квантования $\epsilon(t_{i-1}, i)$ и уменьшается в противном случае;
 - **программируемая** – период Δt изменяется оператором или в соответствии с заранее заданным алгоритмом в различные моменты времени t_i .

Определение

Квантование по уровню – преобразование последовательности мгновенных значений уровня сигнала $x(t_i)$ измеренных в дискретные моменты времени t_i в набор дискретных уровней сигнала x_k , определенных в дискретные моменты времени t_i , а результате: конечномерное множество значений уровня сигнала $x(t_i)$ в диапазоне от x_{\min} до x_{\max} преобразуется в множество дискретных значений уровней квантования x_k рассматриваемого сигнала (графическая интерпретация на рис. 1.12, а и б).

Шаг квантования по уровню Δx определяется по формуле: $\Delta x = x_j - x_{j-1}$.

Квантование по уровню – соотнесение каждого измеренного значения уровня сигнала $x(t_i)$ в дискретный момент времени t_i введенному уровню дискретизации x_j .

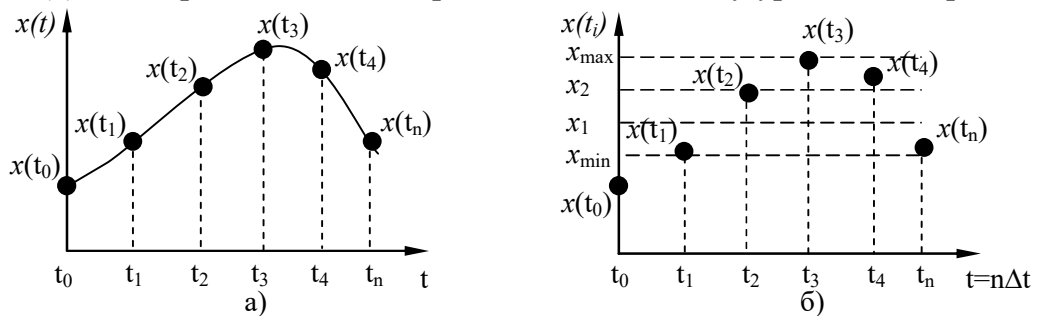


Рисунок 1.12. Иллюстрация к квантованию по уровню:

- а) аналоговые по уровню (но дискретные по времени) сигналы $x(t_i)$ до квантования;
б) квантованные по уровню сигналы $x(t_i)$.

Рис. 1.12, б отражает, что: значение $x(t_0)$ лежит за пределами интервала $[x_{\min}, x_{\max}]$ и не учитывается; значение уровня сигнала $x(t_1)$ соответствует уровню квантования x_{\min} ; $x(t_2) - x_2$; $x(t_3) - x_{\max}$; значение уровня сигнала $x(t_4)$ лежит между двумя уровнями квантования x_2 и x_{\max} (ближе к уровню x_2); значение уровня сигнала $x(t_n)$ находится между x_{\min} и x_1 приблизительно соответствует уровню квантования x_{\min} .

При квантовании по уровню не всегда значение $x(t_i)$ совпадает с уровнем квантования x_j (см. на рис. 1.12, б – $x(t_4)$, $x(t_n)$), тогда поступают следующим образом:

- критерий *avg* – значение $x(t_i)$ соотносят с ближайшим уровнем квантования x_j ;
- критерий *max* – значение $x(t_i)$ отождествляют в любом случае с ближайшим максимальным уровнем x_j , в частности: $x(t_0)$ не рассматривается (выходит за пределы интервала $[x_{\min}, x_{\max}]$); $x(t_1)$ с x_1 ; $x(t_4)$ с x_{\max} ; $x(t_n)$ с x_1 ;
- критерий *min* – значение уровня сигнала $x(t_i)$ в дискретный момент времени t_i соотносят с ближайшим меньшим уровнем квантования x_j , так: $x(t_0)$ не учитывается (выходит за интервал $[x_{\min}, x_{\max}]$); $x(t_1)$ с x_{\min} ; $x(t_2)$ с x_1 ; $x(t_3)$ с x_2 ; $x(t_4)$ с x_2 ; $x(t_n)$ с x_{\min} .

При квантовании по уровню возникает ошибка квантования $e(x_k)$: $e(x_k) = x(t_i) - x_k$.

Погрешность квантования по уровню тем меньше, чем меньше шаг квантования.

Виды квантования по уровню:

- **равномерное** – диапазон изменения сигнала разбивается на m одинаковых частей, тогда можно определить значение максимального уровня квантования $x_{\max} = x_k = k\Delta x$, где k – количество интервалов между уровнями квантования; Δx – значение периода дискретизации по уровню ($\Delta x = \text{const}$);
- **неравномерное** – диапазон изменения сигнала дробится на m разных частей.

1.10.2. Восстановление аналогового сигнала по дискретному

Иногда возникает задача мониторинга и непрерывной обработки данных, которые характеризуют динамическое состояние определенного объекта, процесса или явления с целью последующей выработки совокупности управляющих и корректирующих воздействий, что инициирует необходимость преобразования исходного дискретного сигнала в аналоговый: при этом множество измерений (значений) сигнала в дискретные моменты времени взаимно однозначно преобразуется в непрерывную совокупность уровней сигнала, каждый из которых представляется определенной физической величиной (ток и напряжение).

Определение

Кусочно-линейной аппроксимацией называют последовательное соединение прямыми линиями множества измерений сигнала в дискретные моменты времени выступающих геометрическим метом точек в n -мерной системе координат ($n=2, 3, \dots$).

Особенности преобразования совокупности измерений уровня сигнала в дискретные моменты времени зависят от исходного представления (рис. 1.13):

- если исходный сигнал представляет собой множество измерений уровня сигнала в дискретные моменты времени, то осуществляется процедура кусочно-линейной аппроксимации всех вершин решетчатой функции (а);
- если исходный сигнал представляет собой множество дискретных значений уровня сигнала измеренных в дискретные моменты времени, то осуществляется кусочно-линейная аппроксимация всех точек, которые расположены на пересечении значений дискретных уровней и дискретных моментов времени (б).

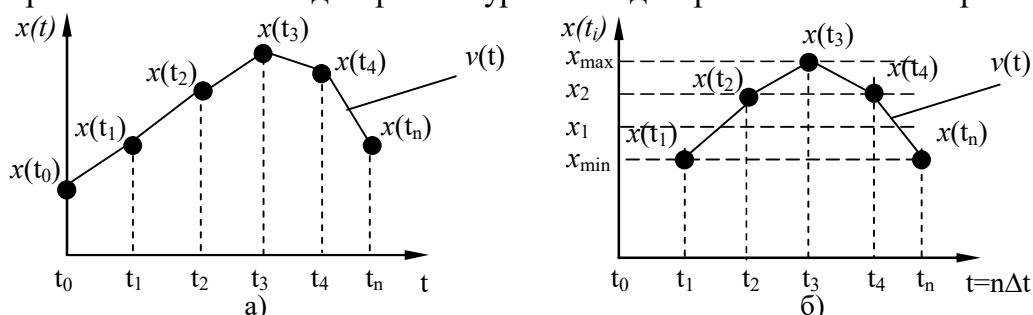


Рисунок 1.13. Кусочно-линейная аппроксимация дискретного сигнала:

а) преобразование дискретного во времени сигнала $x(t_i)$ в аналоговый;

б) преобразование дискретного по уровню сигнала $x(t_i)$ в аналоговый

Полученная в результате кусочно-линейной аппроксимации графическая зависимость непосредственно соответствует функции восстановления $v(t)$ (рис. 1.13).

Степень несоответствия между исходным непрерывным сигналом $x(t)$ и функцией восстановления $v(t)$ выражается разницей площадей и выражает ошибку квантования (рис. 1.14).

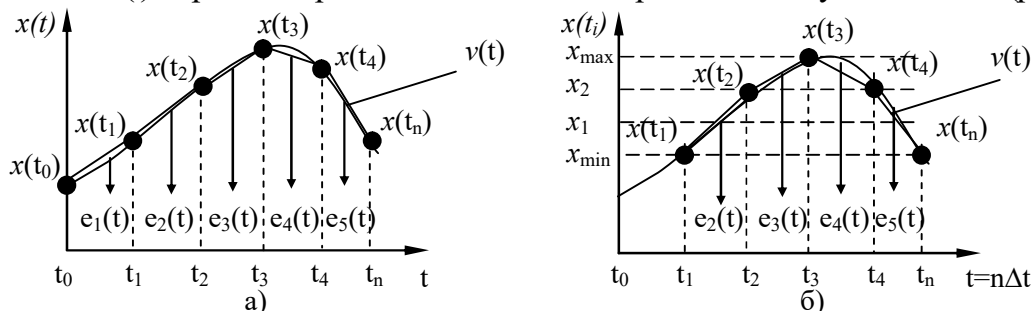


Рисунок 1.14. Ошибка дискретизации при квантовании по времени и по уровню

2. Арифметические основы цифровых автоматов

Для обеспечения возможности обработки дискретного сигнала как множества измерений уровня непрерывного сигнала $x(t)$ в дискретные моменты времени t_i необходимо произвести кодирование каждого отдельно взятого значения $x(t_i)$ или x_j .

Определение

Кодирование – представление значений измеренного уровня сигнала $x(t_i)$ и дискретных уровней x_j в виде кодовых комбинаций, которые пригодны для последующих аналитически-численных расчетов и моделирования с использованием ЭВМ.

Кодирование – совокупность правил и методов записи одного набора символов (**исходный алфавит**) в конечный набор символов (**кодовый алфавит**) посредством использования таблицы преобразования (**кодовая таблица** или **таблица кодов**).

Осуществляется отображение одного набора знаков в другой набор знаков.

Определение

Исходный набор знаков называется **исходным алфавитом**, а набор знаков, который используется для представления данных с целью обработки на компьютере именуется **кодовым алфавитом** или **алфавитом для кодирования** (рис. 2.1).

При этом кодированию подлежат как отдельные символы исходного алфавита, так и их комбинации. Иногда для построения **расширенного (криптографического) кода** используются как отдельные символы кодового алфавита, так и их комбинации.

Определение

Совокупность символов кодового алфавита, которые применяются для кодирования одного символа (или одной комбинации символов) исходного алфавита называется **кодовой комбинацией** или **кодом символа**. При этом кодовая комбинация может содержать один или несколько символов **кода** и **кодовой таблицы**.

Определение

Символ (комбинация символов) исходного алфавита, которому однозначно соответствует кодовая комбинация называется **исходным символом** при кодировании.

Взаимосвязь символов или комбинаций символов исходного алфавита с их кодовыми комбинациями составляет **таблицу соответствия** или **таблицу кодов**.

Определение

Код – это совокупность знаков (символов) и система определенных правил, при помощи которых информация представляется (кодируется) в виде набора из таких символов для передачи, хранения и обработки посредством использования ЭВМ.

Определение

Обратная процедура получения исходных символов по кодам символов с использованием кодовой таблицы соответствия называется **декодированием** (рис. 2.1).

Для выполнения правильного декодирования код должен быть однозначным – одному исходному знаку должен соответствовать точно один код и наоборот.

В цифровом автомате (ЭВМ) процедуры кодирования и декодирования реализуются на микропрограммном уровне на основе созданного алгоритмического обеспечения.



Рисунок 2.1. Структура (де)кодирования информации для ее обработки на компьютере
В зависимости от целей кодирования, различают следующие его виды:

- **кодирование по образцу** – каждое значение уровня дискретного сигнала представляется знаком или набором знаков того алфавита, в который выполняется процедура кодирования, в частности: используется всякий раз для ввода информации в компьютер и для ее внутреннего представления;
- **криптографическое кодирование или шифрование** – используется в случае необходимости защиты информации от несанкционированного доступа;
- **эффективное или оптимальное кодирование** – с целью устранения первичной избыточности в данных путем снижения среднего числа символов кодового алфавита для представления набора исходных символов, что позволяет реализовать алгоритмы сжатия информации в ЭВМ (архиваторы);
- **помехоустойчивое кодирование** – для внесения дополнительных качеств в информационные элементы кода: обнаруживающие, корректирующие и исправляющие свойства, которые позволяют существенно повысить эффективность передачи данных по зашумленному каналу связи (обмена).

2.1. Основные формы представления чисел в компьютере

Теория информации включает теорию кодирования, которая охватывает большое количество дополнительных разделов, которые здесь не рассматриваются.

Существует большое количество способов представления (кодирования) информации в ЭВМ, которые основаны на различных подходах и критериях (рис. 2.2).

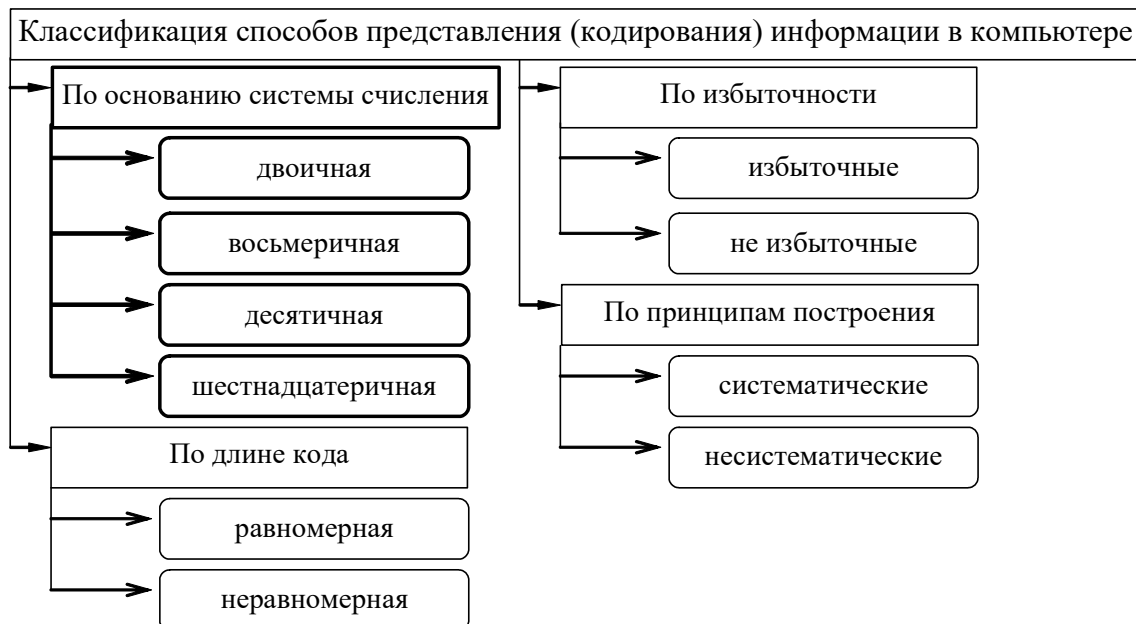


Рисунок 2.2. Классификация способов представления информации

При принудительном внесении избыточности в виде дополнительных информационных элементов осуществляется добавление расширенных свойств в данные:

- избыточные коды – обладают свойствами обнаружения, исправления (коррекции) ошибок при передаче данных по зашумленному каналу связи;
- не избыточные коды – не обладают дополнительными свойствами, поскольку не содержат избыточных информационных элементов в своей основе.

Принцип построения кодовой комбинации предполагает наличие закономерности:

- систематический код – итоговая кодовая последовательность формируется на основе заданной функциональной зависимости или закономерности;
- несистематические – кодовая последовательность формируется на основе определенного метода, правила, алгоритма без явной закономерности.

Анализ длины кодовой комбинации при представлении данных позволяет выделить:

- равномерную кодовую последовательность – количество информационных элементов (битов и разрядов) в кодовой комбинации (цифр в числе) постоянно;
- неравномерная – количество цифр во всех используемых числах переменна.

Предлагается рассмотреть представление чисел в ЭВМ посредством использования систем счисления с основаниями: 2 – двоичная (BINary), 8 – восьмеричная (OCTal), 10 – десятичная (DECimal), 16 – шестнадцатеричная (HEXadecimal).

Основным критерием классификации способов представления выступает тип числа:

- число без знака – выделяется явно набор правил для целой и дробной частей;
- со знаком – резервируется знаковый бит и применяются аналогичные правила.

Классически в ЭВМ применяют прямой, обратный и дополнительный коды чисел.

Определение

Прямой код числа (ПК) – соответствует записи числа в BIN системе счисления.

При использовании ПК число просто записывается в двоичной системе счисления, но для чисел со знаком вводится знаковый бит, который не изменяется:

- для положительных чисел – номинальное значение знакового бита равно 0;
- для отрицательных чисел – номинал знакового бита равен единице (1).

Определение

Обратный код числа (ОК) формируется посредством реализации побитовой инверсии прямого кода (ПК) исходного числа в двоичной системе счисления.

Определение

Инверсия – предполагает реализацию операции замены номинального значения каждого бита в последовательности цифр двоичного числа на противоположный.

Определение

Инкрементация – арифметическая операция приращения числа на единицу.

Определение

Дополнительным кодом числа (ДК) называют инкрементацию на единицу обратного кода (ОК) заданного числа. Аналитическое выражение: ДК=ОК+1.

Дополнительный код (ДК) числа введен в основу арифметико-логического устройства для приведения арифметической операции вычитания к операции сложения.

Сложение отрицательных чисел осуществляется в ДК:
$$\begin{array}{r} 5 \\ - \quad 3 \\ \hline 2 \end{array} \Leftrightarrow \begin{array}{r} 5 \\ + \quad -3 \\ \hline 2 \end{array} \text{ (табл. 2.1).}$$

Таблица 2.1

Таблица соответствия при записи чисел с использованием прямого, обратного и дополнительного кодов

Десятичное число	Прямой код (ПК)	Обратный код (ОК)	Дополнительный код (ДК)
-8	–	–	1000
-7	1111	1000	1001
-6	1110	1001	1010
-5	1101	1010	1011
-4	1100	1011	1110
-3	1011	1100	1101
-2	1010	1101	1110
-1	1001	1110	1111
0	0000	0000	0000
1	0001	0001	0001
2	0010	0010	0010
3	0011	0011	0011
4	0100	0100	0100
5	0101	0101	0101
6	0110	0110	0110
7	0111	0111	0111

Рассматриваются системы счисления с различным основанием, которые позволяют представлять числа и выполнять арифметические операции над ними.

Поэтому большинство существующих кодов основано на системах счисления, которые используют *позиционный принцип* записи числа посредством набора цифр, а значение каждой отдельной цифры зависит от ее положения (разряда) в числе.

Система счисления определяет способ представления информации и данных.

Определение

Система счисления – это способ записи чисел с помощью ограниченного множества символов, имеющих определенный смысл или номинальное значение.

Определение

Система счисления – это способ записи чисел с помощью ограниченного множества символов алфавита кода (цифр), которые имеют номинальное значение и располагаются в определенной позиции разрядной сетки с заданным весом.

Количество цифр ($N=n+m$, где n – количество разрядов в целой части числа, m – количество разрядов в дробной части числа) для записи числа и вес каждого разряда определяется основанием системы счисления (P): для целой части – P^i , для дробной части – P^{-j} .

Таким образом, системы счисления делятся на позиционные и непозиционные в зависимости от способа представления последовательности цифр числа (рис. 2.3).

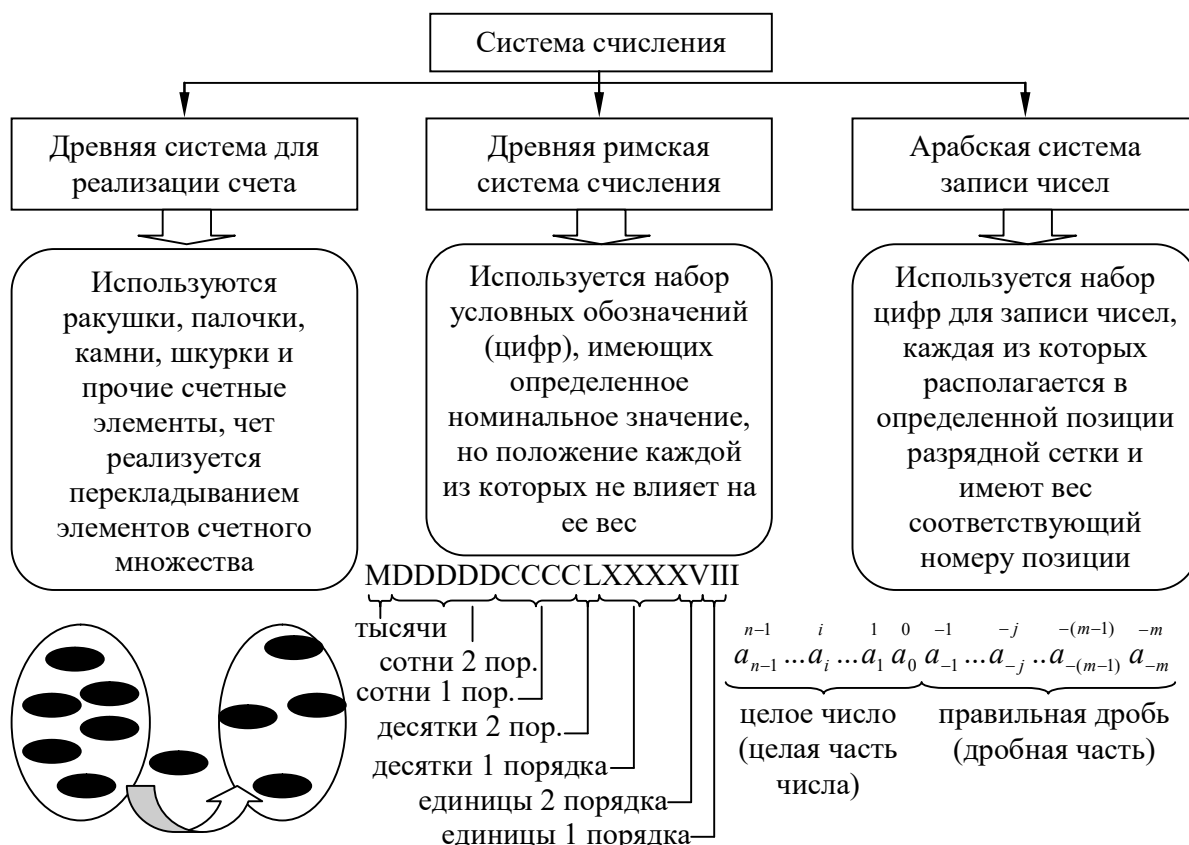


Рисунок 2.3. Характерные особенности позиционных и непозиционных систем счисления

В непозиционной системе счисления перестановка цифр оказывает влияние на номинальное значение числа, но отсутствует разрядная сетка и веса разрядов.

В позиционной системе счисления номинальное значение числа зависит от номинального значения каждой цифры и веса каждого разряда в разрядной сетке.

В римской системе счисления для записи произвольного числа используются следующие цифры, которые имеют определенное номинальное значение (номинал):

- I-III – единицы 1 порядка, при этом I имеет номинал равный 1;
- IV-VIII – единицы 2 порядка, где V имеет номинал равный 5;
- IX-XXXXVIII – десятки 1 пор., при этом X имеет номинал равный 10;
- IL-LXXXXVIII – десятки 2 пор., где L имеет номинал 50;
- IC-CCCCLXXXXVIII – сотни 1 порядка, при этом C имеет номинал 100;
- ID- DCCCCLXXXXVIII – сотни 2 порядка, где D имеет номинал 500;
- IM- M... – тысячи, при этом M имеет номинал равный 1000.

Таким образом, совокупность цифр имеющих определенный номинал (значение), которые стоят в определенном знакоместе (разряде) образует номинальное значение числа в той или иной системе счисления: *непозиционной или позиционной*.

Определение

Минимально необходимое количество цифр с различным номинальным значением, которое достаточно для записи произвольного числа в определенной *позиционной системе счисления* называют **основанием системы счисления (P)**.

При записи числа с основанием P значения цифр лежат в пределах от 0 до P-1.

В общем случае запись любого числа Z в позиционной системе счисления с основанием P будет представлять собой аналитическую запись следующего вида:

$$Z = \underbrace{a_{n-1}P^{n-1} + a_{n-2}P^{n-2} + \dots + a_1P^1 + a_0P^0}_{\text{разложение целой части числа}} + \underbrace{a_{-1}P^{-1} + a_{-2}P^{-2} + \dots + a_{-m}P^{-m}}_{\text{разложение правильной дроби}},$$

$$Z_{\text{целое}} = \sum_{i=0}^{n-1} a_i P^i$$

$$Z_{\text{дробное}} = \sum_{j=-m}^{-1} a_j P^j$$

$$Z = Z_{\text{целое}} + Z_{\text{правильная дробь}}$$

где первые n разрядов образуют целую часть, а последующие за нулевым m разрядов образуют дробную часть числа Z; i – порядковый номер разряда в целой части числа; j – порядковый номер разряда в дробной части числа; a_i – номинал цифры стоящей в i-м разряде целой части числа; a_j – номинал цифры стоящей в j-м разряде дробной части числа; P – основание системы счисления (количество цифр).

Для расчета десятичного эквивалента номинального значения числа, которое представлено в заданной системе счисления с определенным основанием необходимо:

- записать цифры исходного числа и пронумеровать все его разряды;
- выделить явно целую и дробную части: целое число и правильную дробь;
- применить формулу для расчета $Z_{\text{целое}}$ (для целой части числа);
- применить формулу для расчета $Z_{\text{правильная дробь}}$ (для дробной части числа);
- с целью получения результирующего значения числа в десятичной системе счисления необходимо сложить полученные результаты для целой и дробной частей: $Z = Z_{\text{целое}} + Z_{\text{правильная дробь}}$, где $Z_{\text{целое}}$ – разложение целой части представленного числа, $Z_{\text{правильная дробь}}$ – разложение правильной дроби.

2.2. Основные виды позиционных систем счисления

Систем счисления по заданному основанию (P) может быть бесконечно много. В современной информатике используют 4 позиционные системы счисления:

- **двоичная система счисления (основание $P=2$, указывается BINary, В или b после числа)** используется для кодирования дискретного сигнала, потребителем которого является вычислительная техника, что обусловлено требованиями аппаратной реализации функциональных схем логических устройств, развитием и использованием алгебры логики (применяется два устойчивых логических состояния: 0 – ложь или 0В; 1 – истина или 5В);
- **восьмеричная система счисления (основание $P=8$, указывается OCTal, О или o после младшего разряда числа)** используется в качестве вспомогательной и является производной от двоичной системы счисления;
- **шестнадцатеричная система счисления (основание $P=16$, указывается HEXadecimal, H или h после младшего разряда числа)** является производной от двоичной и используется для представления (кодирования) данных, потребителем которых является хорошо подготовленный пользователь – специалист в области информатики, при этом представляется содержание параграфов памяти и открытого файла в интегрированных оболочках для операционных систем, например: Norton Commander в MS DOS, Windows Commander в MS Windows (используемые символы для представления числа: десятичные цифры от 0 до 9, которые дополняются буквами латинского алфавита: A=10, B=11, C=12, D=13, E=14, F=15);
- **десятичная система счисления (основание $P=10$, указывается DECimal, D или d после числа, допустимо полное отсутствие обозначений)** используется для кодирования дискретного сигнала и представления данных, потребителем которых является конечный пользователь – неспециалист в области теории информации и информационных технологий (используемые знаки для представления числа – арабские цифры от 0 до 9).

При осуществлении процедуры автоинкрементации с приращением на единицу хорошо видно, что заполнение и переполнение разрядной сетки, а также перенос единицы из младшего разряда в старший происходит с различной скоростью: быстрее всего в двоичной системе счисления, медленнее всего в шестнадцатеричной.

Возникают существенные сложности при осуществлении простейших арифметических операций над числами, представленных в разных системах счисления:

- при осуществлении арифметических операций сложения и умножения перенос номинального значения (значащей единицы) из младшего разряда в старший затрудняется с ростом номинала основания системы счисления;
- при реализации операций вычитания и деления заем значащей единицы из старшего разряда упрощается с уменьшением основания системы счисления.

Самой простой для восприятия человека является двоичная система счисления.

Рассматриваемые системы счисления имеют некоторые взаимные отличия, которые определяют скорость заполнения и достижение состояния переполнения разрядной сетки при выполнении вычислений, а также обладают разной сложностью.

Двоичная система счисления (BIN, P=2) теоретически является самой простой для осуществления простейших арифметических операций над числами, поскольку проще всего отслуживается перенос в старший разряд при выполнении сложения и умножения, а также заем из старшего разряда при реализации вычитания и деления, но она обладает одним существенным практическим недостатком – наиболее быстро достигается состояние переполнения разрядной сетки, тем не менее, используется в силу ограничений аппаратной реализации логических устройств в основе компьютера.

Восьмеричная система счисления (ОСТ, P=8) выступает производной от двоичной, поскольку каждая цифра ОСТ числа представляется в виде двоичной комбинации из трех разрядов: триада двоичных разрядов или двоичная триада.

Шестнадцатеричная система счисления (HEX, P=16) выступает производной от двоичной системы счисления, поскольку каждая цифра HEX числа представляется в виде четырех двоичных разрядов: тетрада двоичных разрядов или двоичная тетрада, переполнение разрядной сетки происходит медленнее по отношению к BIN и HEX.

Взаимное однозначное соответствие между несколькими натуральными числами, которые представлены в четырех разных системах счисления приведено в табл. 2.2.

Таблица 2.2

Соответствие между числами, которые представлены в различных системах счисления

Номер комбинации по порядку	Десятичная система DEC (decimal)	Двоичная система BIN (binary)	Восьмеричная система ОСТ (octal)	Шестнадцатеричная система HEX (hexadecimal)
1	0	0000	000	00
2	1	0001	001	01
3	2	0010	002	02
4	3	0011	003	03
5	4	0100	004	04
6	5	0101	005	05
7	6	0110	006	06
8	7	0111	007	07
9	8	1000	010	08
10	9	1001	011	09
11	10	1010	012	0A
12	11	1011	013	0B
13	12	1100	014	0C
14	13	1101	015	0D
15	14	1110	016	0E
16	15	1111	017	0F
17	16	10000	020	10

Анализ представленной таблицы позволяет сделать ряд выводов:

1. При рассмотрении номинального значения основания системы счисления и скорости заполнения разрядной сетки, в которой содержатся цифры заданного числа:
 - в двоичной системе счисления с основанием равным двум используются две цифры (0 и 1), перенос из младшего разряда в старший происходит на 3, 5, 7, 9, 11, 13, 15, 17 итерации при автоинкрементации с шагом равным 1;
 - в восьмиричной системе счисления с основанием равным восьми используются восемь цифр (0-7), перенос из младшего разряда в старший происходит на 9, 17 итерации при автоинкрементации с шагом равным единице;
 - в десятичной системе счисления с основанием равным десяти используется десять цифр (0-9), а перенос из младшего разряда в старший происходит на 11 итерации при автоинкрементации с шагом равным единице;
 - в шестнадцатеричной системе счисления с основанием шестнадцать используется 16 цифр (0-15), перенос из младшего разряда в старший происходит на 17 итерации при автоинкрементации с шагом равным 1.
2. При рассмотрении особенностей записи чисел в различных системах счисления:
 - в двоичной системе счисления используют 2 цифры (0 – ноль, 1 – единица);
 - для записи максимальной по номиналу цифры числа, которое представлено в восьмиричной (шестнадцатеричной) системе счисления необходимо и достаточно 3 (4) двоичных разряда, поэтому для взаимного преобразования чисел необходимо 3 (4) бита информации: $7_{OCT}=111_{BIN}$ – двоичная триада из трех разрядов ($15_{HEX}=1111_{BIN}$ – двоичная тетрада из четырех бит);
3. При рассмотрении преобразования чисел из одной системы счисления в другую:
 - двоичная система счисления (BIN) является структурно простой и базовой;
 - из двоичной системы счисления (BIN) в восьмиричную систему счисления (OCT) и наоборот преобразование осуществляется посредством ввода двоичных триад, а каждая цифра восьмиричного числа соответствует двоичной триаде;
 - из двоичной (BIN) в шестнадцатеричную систему счисления (HEX) и наоборот преобразование осуществляется посредством ввода двоичных тетрад, а каждая цифра восьмиричного числа соответствует двоичной тетраде;
4. При рассмотрении осуществления разных арифметических операций над числами:
 - в двоичной системе счисления сложение и вычитание, умножение и деление осуществляется непосредственно по методам «в столбик» и «угол», а также допустимо использование специальных таблиц сложения и умножения;
 - в восьмиричной системе счисления сложение и вычитание, умножение и деление реализуется непосредственно посредством использования методов «в столбик» и «угол», а также рекомендуется использовать специальные таблицы сложения и умножения для снижения сложности вычислений;
 - в шестнадцатеричной системе счисления сложение и вычитание, умножение и деление реализуется только посредством использования таблиц.

2.3. Правила перевода чисел из одной системы счисления в другую

Правила перевода (рис. 2.4) имеют свою специфику в зависимости от формата представления числа: целое (целая часть числа) или правильная дробь (дробная часть).

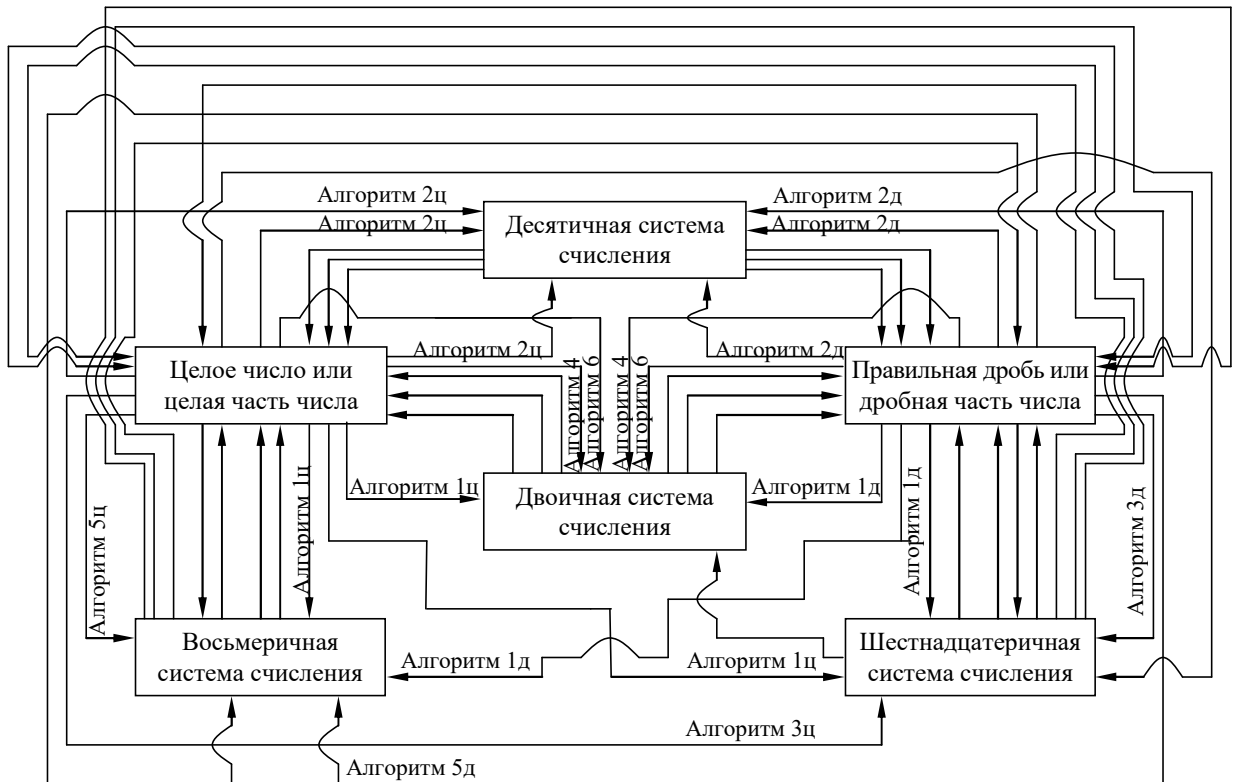


Рисунок 2.4. Топологическая схема взаимного преобразования чисел из одной системы счисления в другую

Для преобразования дробных чисел используется комбинация двух наборов правил:

- для перевода целой части числа (целого числа) – индекс «Алгоритм [цифра]ц»;
- для перевода дробной части заданного числа (правильной дроби) – необходимо использовать алгоритм с индексом «Алгоритм [цифра]д» (рис. 2.4, 2.5).

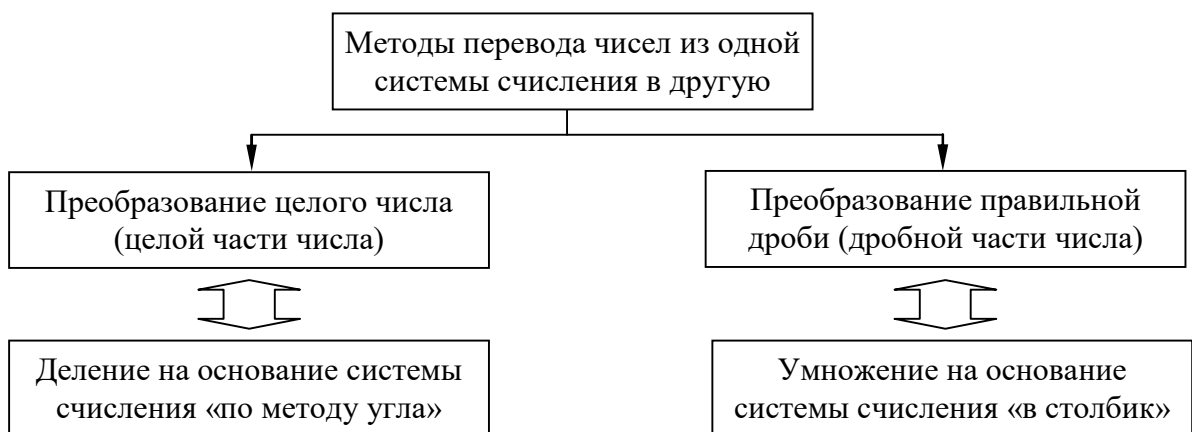


Рисунок 2.5. Методы преобразования целого числа и правильной дроби из десятичной системы счисления в произвольную

Процедура преобразования целых и дробных чисел из одной системы счисления в другую имеет существенные отличия, поскольку она существенно зависит от вида числа, которое необходимо преобразовать: целое число или правильная дробь.

Для преобразования целых чисел Z (целой части числа $Z_{\text{цел}}$) из десятичной системы счисления в двоичную ($P=2$), восьмеричную ($P=8$), шестнадцатеричную ($P=16$) и прочие необходимо использовать метод деления «углом», который включает:

- для операции деления исходное целое десятичное число рассматривается как делимое (Z), делителем выступает основание системы счисления (P), в которую осуществляется перевод с целью получения двоичного эквивалента исходного числа $P=2$, восьмеричного – $P=8$, шестнадцатеричного – $P=16$);
- на первой итерации после осуществления процедуры деления формируются соответственно: частное от деления (C_1) и остаток от деления (r_1);
- на второй и последующих итерациях осуществляется анализ возможности дальнейшего деления полученного на предыдущей итерации частного от деления (C_{i-1}) на основание системы счисления ($P=2, 8$ или 16) – проводится анализ соответствия заданному аналитическому условию $C_i \geq P$:
 - если *выполняется заданное аналитическое условие*, то деление по методу «угла» можно продолжать далее – частное от деления (C_i) делится на основание системы счисления (P); при этом фиксируется на каждой итерации результирующее частное от деления (C_{i+1}) и остаток от деления (r_{i+1});
 - если указанное аналитическое условие *не выполняется*, то процедура деления прекращается, а затем последнее частное от деления (C_k) и все полученные остатки от деления ($r_k, \dots, r_i, \dots, r_1$) записываются в обратном порядке, что позволяет образовать цифры результирующего числа, которое представлено в P -ичной системе счисления ($C_k r_k \dots r_i \dots r_2 r_1$).

Для преобразования правильных дробей из десятичной системы счисления в двоичную ($P=2$), восьмеричную ($P=8$), шестнадцатеричную ($P=16$) и прочие необходимо использовать метод умножения «в столбик», который предполагает ряд шагов:

- рассмотрение исходной правильной дроби (Z) как первого множителя, вторым множителем выступает номинал основания системы счисления (P), при этом очень важно заранее *выделить количество знаков после запятой* (m);
- на первой итерации после осуществления процедуры умножения формируются: результирующее произведение (N_1, M_1), в котором выделяется явно целая (N_1) и дробная (M_1) части, а затем целая часть фиксируется и отбрасывается, дробная часть подлежит дальнейшей процедуре умножения;
- правильная дробь (M_{i-1}), полученная на предыдущей итерации умножается на основание системы счисления (P), в которую осуществляется перевод, а в результирующем произведении (N_i, M_i): целая часть (N_i) фиксируется и отбрасывается, а дробная часть (M_i) подлежит дальнейшей процедуре умножения;
- процедура умножения продолжается до тех пор пока не будет достигнута:
 - требуемая точность преобразования, которая определяется количеством знаков после десятичной запятой в исходном заданном числе (m);
 - полученная дробная часть не будет соизмерима с алгебраическим нулем.

2.3.1. Правила перевода целых чисел

Алгоритм 1ц. Из десятичной системы счисления (DEC, $P=10$) в двоичную (BIN, $P=2$), восьмеричную (OCT, $P=8$) и шестнадцатеричную (HEX, $P=16$).

Для осуществления процедуры преобразования исходного целого числа (целой части числа) из десятичной системы счисления в P -ичную систему счисления необходимо воспользоваться рекуррентным алгоритмом 1ц предложенном на рис. 2.6.

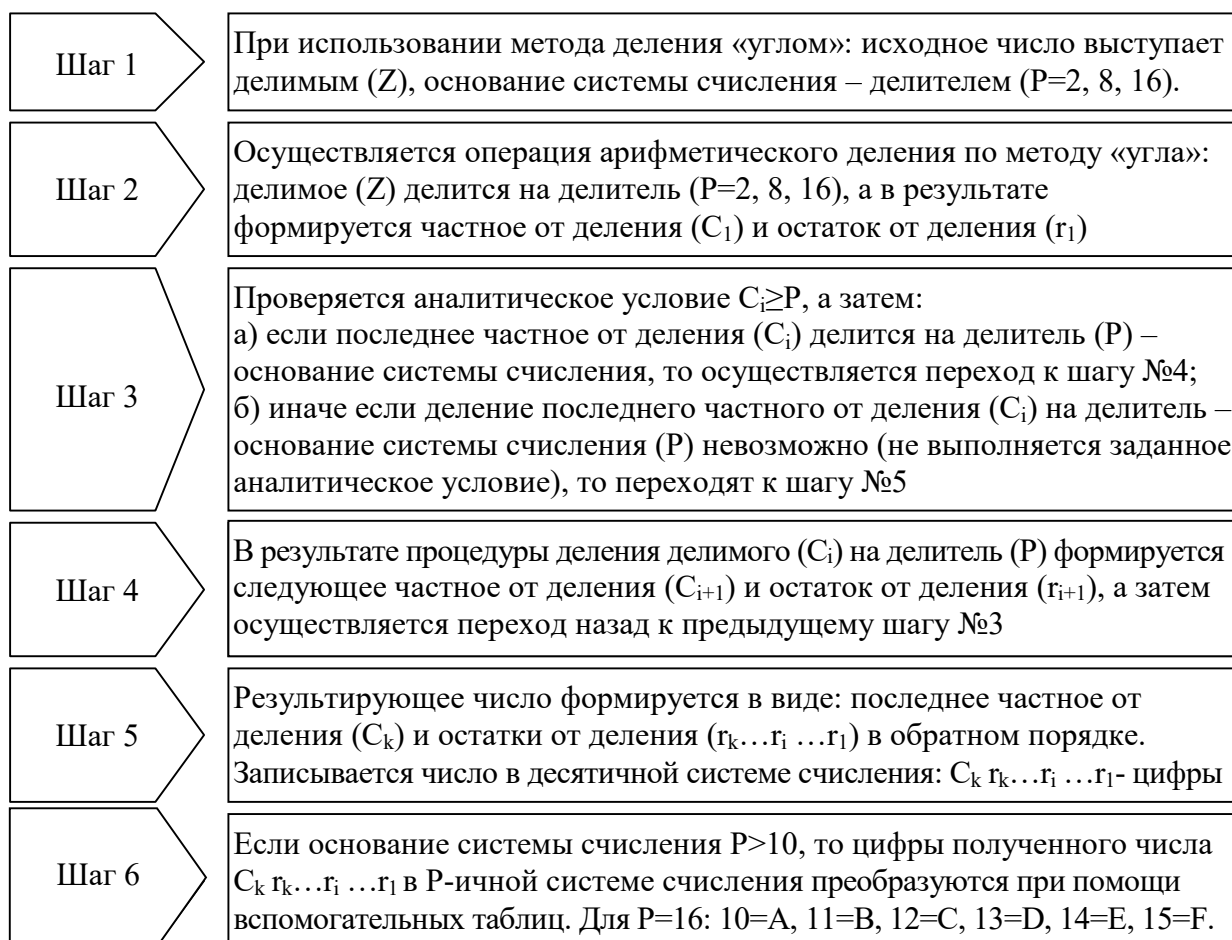


Рисунок 2.6. Алгоритм преобразования целой части десятичного числа в P -ичную систему счисления

Преобразование исходного целого числа или целой части числа из десятичной в P -ичную систему счисления осуществляется итеративно по шагам согласно алгоритму 1ц, а в результате операции деления методом «угол» формируется целое число.

Все полученные остатки от деления и последнее частное от деления целесообразно обводить кружками или обозначать маркерами (рис. 2.7), чтобы было удобнее сформировать результирующее число в виде последовательности его цифр: [последнее частное от деления – C_k][последний остаток от деления – r_k][предпоследний остаток от деления – r_{k-1}]...[i -й остаток от деления – r_i]...[второй остаток от деления – r_2][первый остаток от деления – r_1]. Для формирования результирующего числа последнее частное от деления и остатки берутся в обратном порядке.

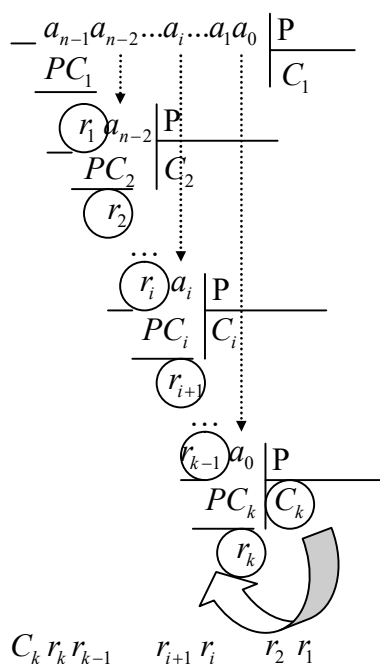


Рисунок 2.7. Метод деления «углом» при преобразовании целого числа из десятичной системы счисления в R -ичную систему счисления

Пример 2.1. Выполнить перевод числа 19 в двоичную систему счисления:

$$\begin{array}{r}
 \underline{19} \mid \underline{2} \\
 \underline{18} \quad \underline{9} \mid \underline{2} \\
 \mathbf{1} \quad \underline{8} \quad \underline{4} \mid \underline{2} \\
 \quad \mathbf{1} \quad \underline{4} \quad \underline{2} \mid \underline{2} \\
 \quad \quad \mathbf{0} \quad \underline{2} \quad \mathbf{1} \\
 \quad \quad \quad \mathbf{0}
 \end{array}$$

← последнее частное от деления (последующее деление 1 на 2 нацело не представляется возможным, $1 < 2$).
 Получен старший (четвертый) разряд результирующего двоичного числа.

1 0 0 1 1 – результирующее число или $19 = 10011_2$.

Пример 2.2. Выполнить перевод десятичного числа 19 в HEX:

$$\begin{array}{r}
 \underline{19} \mid \underline{16} \\
 \underline{16} \quad \mathbf{1} \\
 \mathbf{3}
 \end{array}$$

1 3 – результирующее число или $19 = 13_{16}$.

Пример 2.3. Выполнить перевод числа 123 в HEX:

$$\begin{array}{r}
 \underline{123} \mid \underline{16} \\
 \underline{112} \quad \mathbf{7} \\
 \mathbf{11}
 \end{array}$$

7 B – результирующее число.

Таким образом, $123 = 7B_{16}$.

Алгоритм 2ц. Из двоичной (P=2), восьмеричной (P=8) и шестнадцатеричной (P=16) систем счисления в десятичную (DECimal).

В исходном целом числе вводится нумерация всех разрядов разрядной сетки, в которых последовательно расположены номиналы каждой из его цифр (от 0 до n-1).

$$\underbrace{a_{n-1} \dots a_i \dots a_1 a_0}_{\text{целая часть числа}} \underbrace{a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m}}_{\text{дробная часть числа}}$$

В этом случае рассчитывается номинальное значение целого числа в десятичной системе счисления (DEC) по формуле разложения для целой части числа:

$$Z = \underbrace{a_{n-1}P^{n-1} + a_{n-2}P^{n-2} + \dots + a_1P^1 + a_0P^0}_{\text{разложение целой части числа}} + \underbrace{a_{-1}P^{-1} + a_{-2}P^{-2} + \dots + a_{-m}P^{-m}}_{\text{разложение правильной дроби}},$$

$$Z = Z^{\text{целое}}, \text{ где } Z^{\text{целое}} = \sum_{i=0}^{n-1} a_i P^i.$$

Последовательность преобразования исходного целого числа следующая:

- исходное целое число (Z) записывается поразрядно, а все разряды нумеруются последовательно от младшего (нулевой) до старшего (n-1);
- выделяется заданное основание системы счисления (P), в которой представлено исходное число (Z) – целое число или целая часть числа:
 - для двоичного числа (BIN) – P=2;
 - для восьмеричного (OCT) – P=8;
 - для шестнадцатеричного (HEX) – P=16;
- применяется формула разложения для целой части числа ($Z^{\text{целое}}$) для получения эквивалентного номинального значения исходного числа (Z) в десятичной системе счисления (DEC) независимо от первичной системы счисления;
- номинальное значение цифры числа (a_0), которая расположена в младшем разряде (нулевой разряд) умножается на номинал основания системы счисления (P=2, 8, 16) возведенный в показатель степени, который соответствует номеру разряда цифры числа (ноль), взятой в первом множителе (a_0);
- и так далее, номинальное значение цифры числа (a_i), которая расположена в i-м разряде умножается на номинал основания системы счисления (P) возведенный в показатель степени, который соответствует номеру разряда цифры числа (i), взятой в первом множителе (a_i) i-го произведения, и так далее;
- номинальное значение цифры числа (a_{n-1}), которая расположена в n-1-м разряде умножается на номинал основания системы счисления (P) возведенный в показатель степени, который соответствует номеру разряда цифры числа (n-1), взятой в первом множителе (a_{n-1}) последнего произведения.
- осуществляется перемножение всех множителей (первый множитель a_i и второй множитель P^i), а затем суммирование полученных результатов: от 0 до n-1;
- полученный результат суммирования указанных произведений соответствует номинальному значению целого числа (целой части числа) в DEC.

Пример 2.4. Выполнить перевод числа 10011_2 в десятичную систему счисления. Применим формулу разложения для целого числа (для целой части числа). Имеем: $10011_2 = 1*2^4 + 0*2^3 + 0*2^2 + 1*2^1 + 1*2^0 = 16 + 0 + 0 + 2 + 1 = 19$. Таким образом, $10011_2 = 19$.

Пример 2.5. Выполнить перевод числа 100111_2 в DEC систему счисления. Применим формулу разложения для целого числа (для целой части числа). Имеем: $100111_2 = 1*2^5 + 0*2^4 + 0*2^3 + 1*2^2 + 1*2^1 + 1*2^0 = 32 + 0 + 0 + 4 + 2 + 1 = 41$. Таким образом, $100111_2 = 41$.

Пример 2.6. Выполнить перевод числа 101101_2 в десятичную систему счисления. Применим формулу разложения для целого числа (для целой части числа). Имеем: $101101_2 = 1*2^5 + 0*2^4 + 1*2^3 + 1*2^2 + 0*2^1 + 1*2^0 = 32 + 0 + 8 + 4 + 0 + 1 = 45$. Таким образом, $101101_2 = 45$.

Пример 2.7. Выполнить перевод числа 123_8 в десятичную систему счисления. Применим формулу разложения для целого числа (для целой части числа). Имеем: $123_8 = 1*8^2 + 2*8^1 + 3*8^0 = 64 + 16 + 3 = 83$. Таким образом, $123_8 = 83$.

Пример 2.8. Выполнить перевод числа 1745_8 в десятичную систему счисления. Применим формулу разложения для целого числа (для целой части числа). Имеем: $1745_8 = 1*8^3 + 7*8^2 + 4*8^1 + 5*8^0 = 512 + 112 + 32 + 5 = 661$. Таким образом, $1745_8 = 661$.

Пример 2.9. Выполнить перевод числа 241_8 в десятичную систему счисления. Применим формулу разложения для целого числа (для целой части числа). Имеем: $241_8 = 2*8^2 + 4*8^1 + 1*8^0 = 128 + 32 + 1 = 161$. Таким образом, $241_8 = 161$.

Пример 2.10. Выполнить перевод числа 13_{16} в DEC систему счисления. Применим формулу разложения для целого числа (для целой части числа). Имеем: $13_{16} = 1*16^1 + 3*16^0 = 16 + 3 = 19$. Таким образом, $13_{16} = 19$.

Пример 2.11. Выполнить перевод числа 148_{16} в десятичную систему счисления. Применим формулу разложения для целого числа (для целой части числа). Имеем: $148_{16} = 1*16^2 + 4*16^1 + 8*16^0 = 56 + 64 + 8 = 128$. Таким образом, $148_{16} = 128$.

Пример 2.12. Выполнить перевод числа $A7B5_{16}$ в DEC систему счисления. Применим формулу разложения для целого числа (для целой части числа). Имеем: $A7B5_{16} = 10*16^3 + 7*16^2 + 11*16^1 + 5*16^0 = 462 + 176 + 5 = 643$. Таким образом, $A7B5_{16} = 643$.

Алгоритм 3ц. Из двоичной системы счисления в шестнадцатеричную:

Определение

Двоичной тетрадой называют группу информационных элементов, которая включает 4 бита информации расположенных в четырех двоичных разрядах.

Первый шаг

Исходное число (Z) разбивается на тетрады (группа из 4^x двоичных разрядов), начиная от младшего разряда двоичной комбинации до старшего разряда включительно.

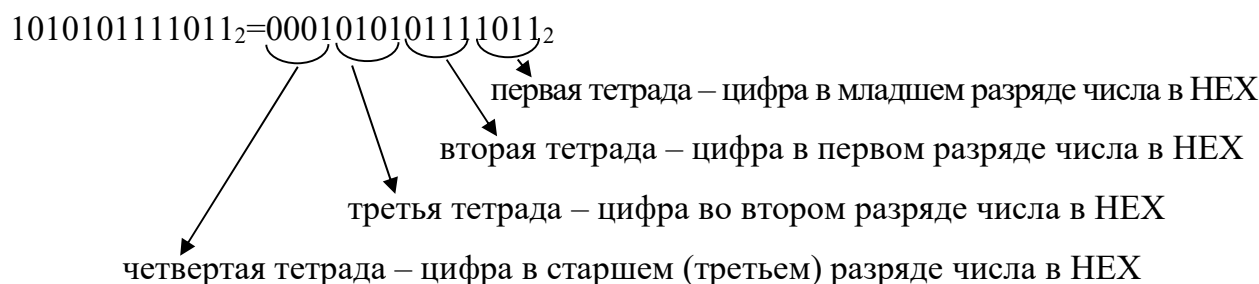
Если количество цифр (n) в имеющейся двоичной комбинации (числе) не кратно четырем, то осуществляется дополнение исходного числа незначащими нулями слева для достижения кратности четырем (дополнение до полной тетрады).

Второй шаг

Для числа представленного в системе счисления с основанием $P > 10$ необходимо использовать вспомогательные таблицы соответствия, в частности: если исходное число представлено в шестнадцатеричной системе счисления, то соответственно производится замена десятичных цифр $10=A$, $11=B$, $12=C$, $13=D$, $14=E$, $15=F$.

Пример 2.13. Выполнить перевод целого двоичного числа 1010101111011_2 в шестнадцатеричную систему счисления.

Поскольку в исходном двоичном числе количество цифр не кратно 4, дополняем его слева тремя незначащими нулями до достижения кратности 4 числа цифр.

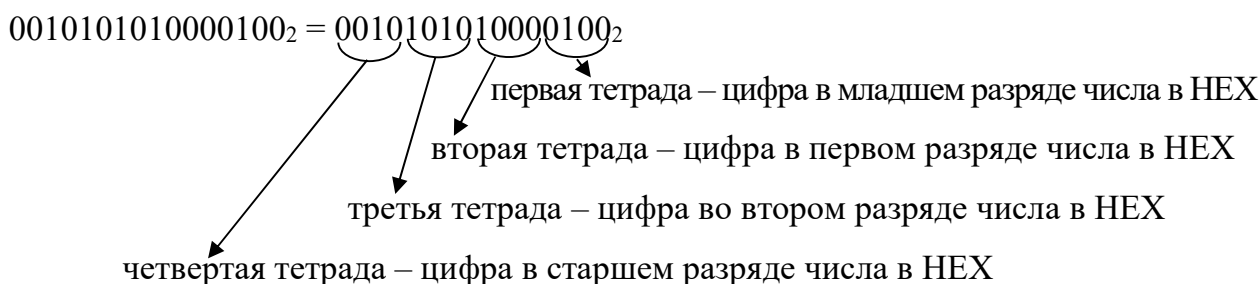


Вспользуемся табл. 2.2: $0001_2 = 1_{16}$, $0101_2 = 5_{16}$, $0111_2 = 7_{16}$, $1011_2 = B_{16}$.

Тогда $1010101111011_2 = 157B_{16}$.

Пример 2.14. Выполнить перевод числа 10101010000100_2 в шестнадцатеричную систему счисления.

Поскольку в исходном двоичном числе количество цифр не кратно четырем, дополняем его слева двумя незначащими нулями до достижения кратности четырем.



Вспользуемся табл. 2.2: $0010_2 = 2_{16}$, $1010_2 = A_{16}$, $1000_2 = 8_{16}$, $0100_2 = 4_{16}$.

Тогда $0010101010000100_2 = 2A84_{16}$.

Алгоритм 4ц. Из шестнадцатеричной системы счисления в двоичную:

Исходное шестнадцатеричное число (Z) разбивается на тетрады (группы из 4^x двоичных разрядов), начиная от младшего разряда двоичной комбинации до старшего.

Если количество цифр (n) в двоичной комбинации (числе) не кратно 4, то осуществляется дополнение числа незначащими нулями слева для достижения кратности 4.

Первый шаг

Каждая цифра исходного шестнадцатеричного числа заменяется справа налево тетрадой двоичных цифр (разрядов), имеющих определенное номинальное значение.

Для целей преобразования необходимо использовать вспомогательную таблицу соответствия между различными системами счисления (табл. 2.2).

Второй шаг

Если в полученной двоичной комбинации цифр числа образуются незначащие нули слева, то они отбрасываются, а конечный результат записывается без них.

Пример 2.15. Выполнить перевод числа 13742_{16} в BIN систему счисления.

По табл. 2.2. имеем: $1_{16} = 0001_2$, $3_{16} = 0011_2$, $7_{16} = 0111_2$, $4_{16} = 0100_2$, $2_{16} = 0010_2$, а после удаления незначащих нулей слева в результирующей двоичной комбинации ~~00010011011101000010~~_b.

Тогда получим $13742_{16} = 10011011101000010_2$.

Пример 2.16. Выполнить перевод числа 48397_{16} в BIN систему счисления.

По табл. 2.2. имеем: $4_{16} = 0100_2$, $8_{16} = 1000_2$, $3_{16} = 0011_2$, $9_{16} = 1001_2$, $7_{16} = 0111_2$, а после удаления незначащего нуля слева в результирующей двоичной комбинации ~~01001000001110010111~~_b.

Тогда получим $48397_{16} = 1001000001110010111_2$.

Пример 2.17. Выполнить перевод числа $ABEF_{16}$ в BIN систему счисления.

По табл. 2.2. имеем: $A_{16} = 1010_2$, $B_{16} = 1011_2$, $E_{16} = 1110_2$, $F_{16} = 1111_2$, а после объединения четырех двоичных тетрад в результирующей двоичной комбинации 101010111101111_2 .

Тогда получим $ABEF_{16} = 101010111101111_2$.

Пример 2.18. Выполнить перевод числа $1AD7_{16}$ в BIN систему счисления.

По табл. 2.2. имеем: $1_{16} = 0001_2$, $A_{16} = 1010_2$, $D_{16} = 1101_2$, $7_{16} = 0111_2$, а после удаления незначащих нулей слева в результирующей двоичной комбинации ~~0001101011010111~~_b.

Тогда получим $1AD7_{16} = 101011010111_2$.

Пример 2.19. Выполнить перевод числа $ACDF_{16}$ в BIN систему счисления.

По табл. 2.2. имеем: $A_{16} = 1010_2$, $C_{16} = 1100_2$, $D_{16} = 1101_2$, $F_{16} = 1111_2$, а после объединения четырех двоичных тетрад в результирующей двоичной комбинации 101011001101111_2 .

Тогда получим $ACDF_{16} = 101011001101111_2$.

Алгоритм 5ц. Из двоичной системы счисления в восьмеричную:

Определение

Двоичной триадой называют группу информационных элементов, которая включает 3 бита информации в содержащихся трех двоичных разрядах.

Первый шаг

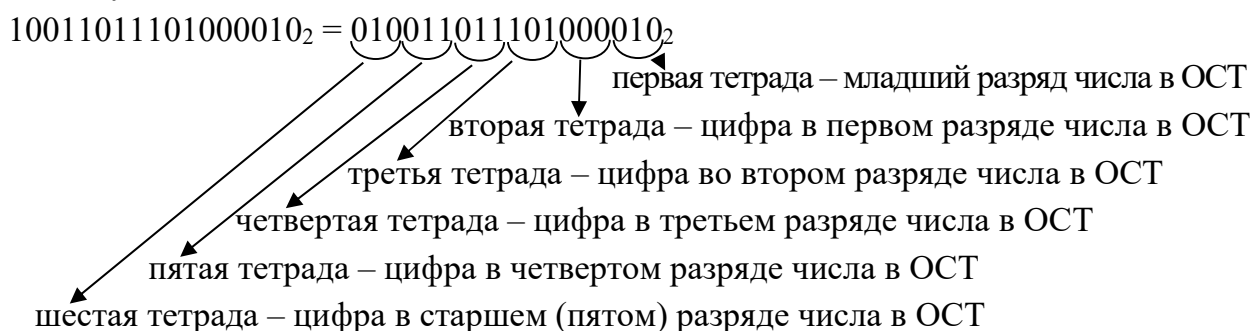
Исходное число (Z) разбивается на триады (группа из трех двоичных разрядов), начиная от младшего (нулевого) разряда двоичной комбинации до старшего ($n-1$ -го).

Если количество цифр (n) в двоичной комбинации (числе) не кратно 3, то осуществляется дополнение числа незначащими нулями слева для достижения кратности 3.

Второй шаг

Каждой триаде двоичных разрядов вводится в соответствие цифра числа, которое представлено в восьмеричной системе счисления, а в процессе преобразования используется табл. 2.2 для вспомогательных и справочных целей.

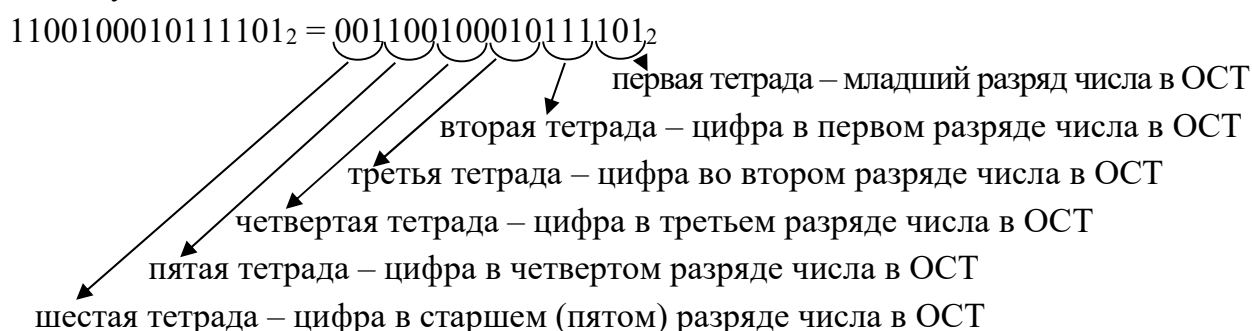
Пример 2.20. Выполнить перевод числа 10011011101000010_2 в восьмеричную систему счисления.



Воспользуемся табл. 2.2: $010_2 = 2_8$, $011_2 = 3_8$, $101_2 = 5_8$, $000_2 = 0_8$, $010_2 = 2_8$.

Тогда $10011011101000010_2 = 233502_8$.

Пример 2.21. Выполнить перевод числа 1100100010111101_2 в восьмеричную систему счисления.



Воспользуемся табл. 2.2: $001_2 = 1_8$, $100_2 = 4_8$, $100_2 = 4_8$, $010_2 = 2_8$, $111_2 = 7_8$, $101_2 = 5_8$.

Тогда $1100100010111101_2 = 144275_8$.

Алгоритм бц. Из восьмеричной системы счисления в двоичную:

Исходное целое число (Z) разбивается на триады (группы из 3^x двоичных разрядов), начиная от младшего (нулевого) разряда двоичной комбинации до старшего ($n-1$ -го).

Если количество цифр (n) в исходной двоичной комбинации (числе) не кратно трем, то осуществляется дополнение числа незначащими нулями слева для достижения кратности трем и реализации возможности дальнейшего преобразования.

Первый шаг

Каждая цифра исходного восьмеричного числа заменяется справа налево триадой двоичных цифр (разрядов), имеющих определенное номинальное значение.

Для целей преобразования необходимо использовать вспомогательную таблицу соответствия между различными системами счисления (табл. 2.2).

Второй шаг

Если в полученной двоичной комбинации (числе) образуются незначащие нули слева, то они отбрасываются, а конечный результат записывается без них.

Пример 2.22. Выполнить перевод числа 13742_8 в BIN систему счисления.

По табл. 2.2. имеем: $1_8 = 001_2$, $3_8 = 011_2$, $7_8 = 111_2$, $4_8 = 100_2$, $2_8 = 010_2$, а после удаления незначащих нулей слева в результирующей двоичной комбинации ~~001011111100010b~~.

Тогда получим $13742_{16} = 1011111100010_2$.

Пример 2.23. Выполнить перевод числа 345432_8 в BIN систему счисления.

По табл. 2.2. имеем: $3_8 = 011_2$, $4_8 = 100_2$, $5_8 = 101_2$, $4_8 = 100_2$, $3_8 = 011_2$, $2_8 = 010_2$, а после удаления незначащего нуля слева в результирующей двоичной комбинации ~~011100101100011010b~~.

Тогда получим $345432_{16} = 11100101100011010_2$.

Пример 2.24. Выполнить перевод числа 37467_8 в BIN систему счисления.

По табл. 2.2. имеем: $3_8 = 011_2$, $7_8 = 111_2$, $4_8 = 100_2$, $6_8 = 110_2$, $7_8 = 111_2$, а после удаления незначащего нуля слева в результирующей двоичной комбинации ~~011111100110111b~~.

Тогда получим $37467_{16} = 11111100110111_2$.

Пример 2.25. Выполнить перевод числа 27543_8 в BIN систему счисления.

По табл. 2.2. имеем: $2_8 = 010_2$, $7_8 = 111_2$, $5_8 = 101_2$, $4_8 = 100_2$, $3_8 = 011_2$, а после удаления незначащего нуля слева в результирующей двоичной комбинации ~~010111101100011b~~.

Тогда получим $27543_{16} = 10111101100011_2$.

Пример 2.26. Выполнить перевод числа 6354_8 в BIN систему счисления.

По табл. 2.2. имеем: $6_8 = 110_2$, $3_8 = 011_2$, $5_8 = 101_2$, $4_8 = 100_2$, а после объединения четырех двоичных триад в результирующей двоичной комбинации $110011101100b$.

Тогда получим $6354_{16} = 110011101100_2$.

2.3.2. Правила перевода правильных дробей

Алгоритм 1д. Из десятичной системы счисления (DEC, P=10) в двоичную (BIN, P=2), восьмеричную (OCT, P=8) и шестнадцатеричную (HEX, P=16).

Для осуществления процедуры преобразования исходной правильной дроби (дробной части числа) из десятичной системы счисления в P-ичную систему счисления необходимо воспользоваться рекуррентным алгоритмом 1д на рис. 2.8.

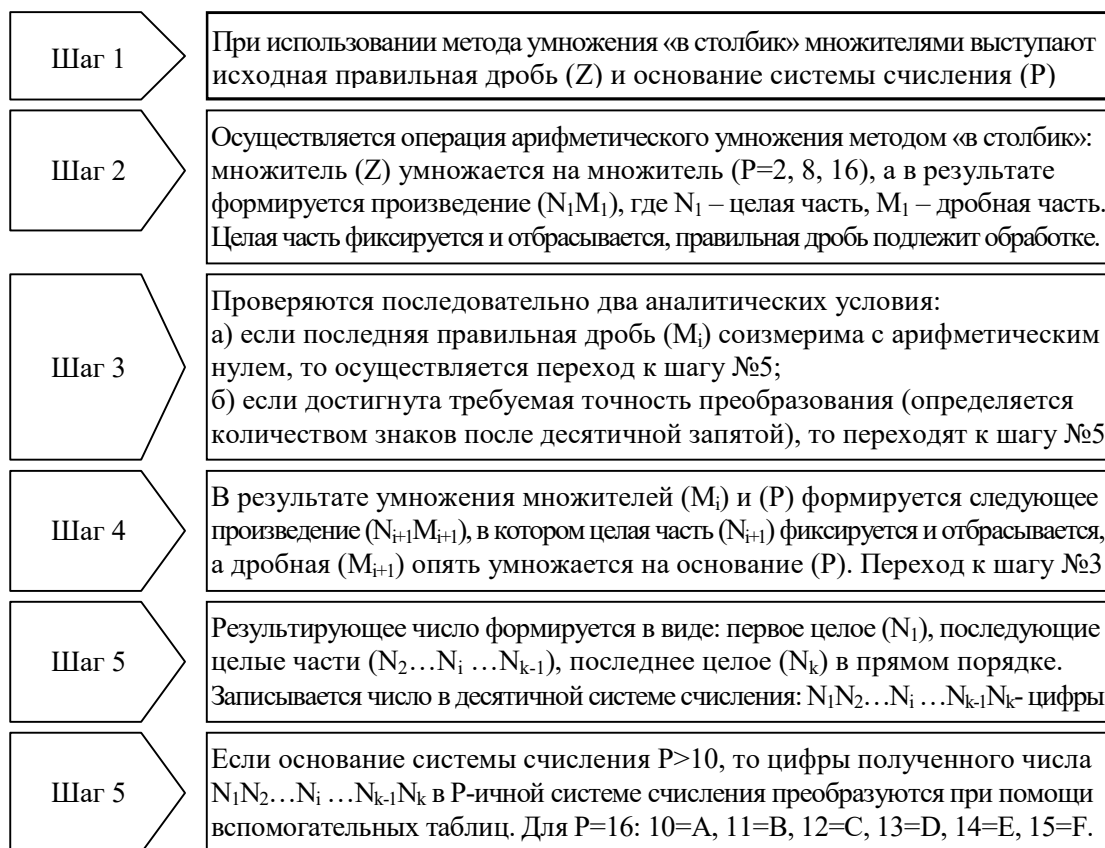


Рисунок 2.8. Алгоритм преобразования дробной части десятичного числа в P-ичную систему счисления

Преобразование исходной правильной дроби или дробной части числа из десятичной в P-ичную систему счисления осуществляется итеративно по шагам согласно алгоритму 1д, а в результате операции формируется правильная дробь.

Все полученные целые части (N_i), которые сформированы в результате умножения первого множителя (исходная правильная дробь – Z^{правильная дробь} = $0, a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m}$, а впоследствии правильные дроби – M_i) на второй множитель (основание системы счисления – P) целесообразно обводить кружками или обозначать маркерами (рис. 2.9), чтобы было удобнее сформировать результирующее число в виде последовательности его цифр: [первая целая часть числа – N₁][вторая часть числа – N₂]...[i-я целая часть числа – N_i]... [предпоследняя целая часть числа – N_{k-1}] [последняя часть числа – N_k]. Для систем счисления с основанием P≥10 используются таблицы соответствия между числами, представленными в разных системах счисления. В частности, для шестнадцатеричной системы счисления (P=16): A=10₁₀=1010₂, B=11₁₀=1011₂, C=12₁₀=1100₂, D=13₁₀=1101₂, E=14₁₀=1110₂, F=15₁₀=1111₂.

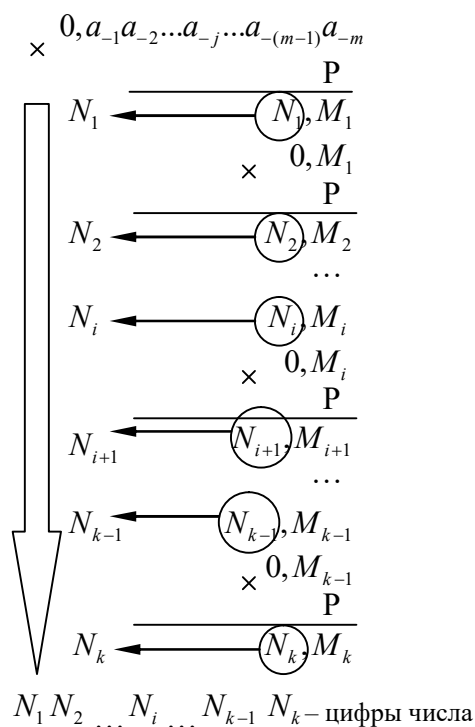


Рисунок 2.9. Метод умножения «в столбик» при преобразовании дробной части числа из десятичной системы счисления в P-ичную систему счисления

В результате преобразования дробной части числа (правильной дроби) по методу умножения «в столбик» образуется ряд $N_1 N_2 \dots N_i \dots N_{k-1} N_k$, в котором представлены в прямом порядке цифры результирующего числа в P-ичной системе счисления.

В процессе получения цифр правильной дроби в P-ичной системе счисления необходимо обратить внимание на количество знаков после десятичной запятой, которые определяют исходную точность представления числа в десятичной системе счисления. Точность полученного числа обычно принимается равной ± 1 относительно количества знаков после десятичной запятой в исходном заданном числе.

Операцию умножения исходной правильной дроби ($Z^{\text{правильная дробь}}$) на определенное основание системы счисления ($P=2$ – BINary, 8 – OCTal, 16 – HEXadecimal) продолжают до достижения требуемой точности преобразования или пока дробная часть числа не будет соизмерима с арифметическим нулем (умножение прекращают).

Результатом преобразования исходной правильной дроби из десятичной системы счисления в P-ичную (двоичную, восьмеричную, шестнадцатеричную) всегда является правильная дробь, которая формируется из набора цифр, образованных из последовательности целых частей ($N_1 N_2 \dots N_i \dots N_{k-1} N_k$) полученных в результате умножения исходной правильной дроби ($Z^{\text{правильная дробь}}$) и промежуточных правильных дробей ($M_1 M_2 \dots M_i \dots M_{k-1} M_k$) на основании определенной системы счисления (P).

В системах счисления с основанием $P \leq 10$ для формирования числа используются цифры от нуля до девяти, а в системах счисления с основанием $P > 10$ для обозначения цифр числа вводят буквы латинского алфавита (A, B, C, D и так далее), а затем допустимо использовать разные дополнительные символы (@, #, \$, %, * и прочие).

Рассмотрим подробно по шагам алгоритм 1д для преобразования правильной дроби.

Первый шаг

Выделяется первый и второй множители: исходная правильная дробь ($Z^{\text{правильная дробь}}$) и основание системы счисления (P) для осуществления преобразования.

Второй шаг

Исходная правильная дробь ($Z^{\text{правильная дробь}}$) умножается методом «в столбик» на основание системы счисления ($P=2, 8$ или 16), в которую осуществляется преобразование, а в результате на первой итерации получается дробное число $Z_1 = N_1, M_1$, где N_1 – целая часть, M_1 – дробная часть числа (является правильной дробью).

В полученном произведении ($Z_i = N_i, M_i$) целая часть дробного числа (N_i) фиксируется и не учитывается при преобразовании, а дробная часть (M_i) подлежит дальнейшему преобразованию с использованием представленного алгоритма 1д.

Третий шаг

Процедура арифметического умножения продолжается до достижения требуемой точности преобразования (получено определенное количество цифр после десятичной запятой $0, N_1 N_2 \dots N_i \dots N_{k-1} N_k$) или до тех пор пока полученная правильная дробь (дробная часть числа – M_i) не будет соизмерима с арифметическим нулем.

Четвертый шаг

Полученная правильная дробь (дробная часть числа – M_i) повторно умножается на заданное основание системы счисления (P), в результате формируется дробное число ($Z_{i+1} = N_{i+1}, M_{i+1}$), в котором по аналогии с предыдущим шагом целая часть (N_{i+1}) фиксируется и отбрасывается, а дробная часть (правильная дробь – M_{i+1}) вновь умножается на заданное основание системы счисления (P).

Пятый шаг

Результирующее число является правильной дробью и формируется в виде: $0, N_1 N_2 \dots N_i \dots N_{k-1} N_k$, где $N_1, N_2, \dots, N_i, \dots, N_{k-1}, N_k$ – цифры правильной дроби, которые образованы из последовательно зафиксированных целых частей (N_i).

Шестой шаг

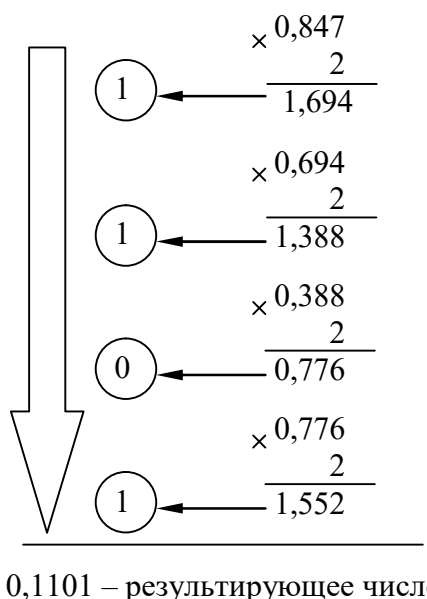
В системах счисления с основанием $P \geq 10$ для преобразования целой части (N_i) используются вспомогательные таблицы соответствия между системами счисления, которые позволяют преобразовать каждую цифру ($N_1 N_2 \dots N_i \dots N_{k-1} N_k$) в требуемую систему счисления (с основанием $P \geq 10$), где N_1 – старший разряд (цифра правильной дроби), N_k – младший разряд. Для $P=16$ необходимо использовать табл. 2.2.

Полученная в результате использования алгоритма преобразования правильная дробь ($0, N_1 N_2 \dots N_i \dots N_{k-1} N_k$) не соответствует точно исходному числу ($Z^{\text{правильная дробь}}$) поскольку процедура преобразования осуществляется не полностью и останавливается на определенном этапе: при достижении требуемой точности преобразования или при $M_i \approx 0$. Для проверки полученного результата необходимо использовать формулу разложения для правильной дроби (дробной части исходного числа).

$$Z^{\text{правильная дробь}} = \sum_{j=-m}^{-1} a_j P^j = a_{-1} P^{-1} + a_{-2} P^{-2} + \dots + a_{-j} P^{-j} + \dots + a_{-m} P^{-m},$$

где a_j – цифра числа, которая расположена в j -м разряде правильной дроби.

Пример 2.27. Выполнить перевод числа 0,847 из десятичной в двоичную систему счисления. Перевод выполнить до четырех значащих цифр после запятой.



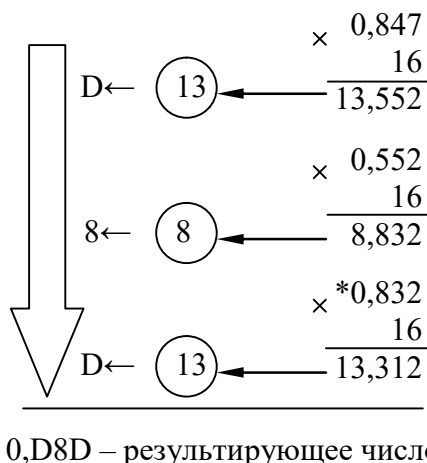
В данном примере процедура выполнения перевода прервана на четвертом шаге, поскольку получено требуемое число разрядов после запятой в результирующем числе.

Очевидно, это привело к потере ряда цифр результирующего числа.

Таким образом, $0,847 = 0,1101_2$.

Расхождение полученной правильной дроби в двоичной системе счисления с исходным представлением в DEC обусловлено тем, что процедура перевода из DEC в BIN прервана на четвертой итерации используемого алгоритма 1д.

Пример 2.28. Выполнить перевод числа 0,847 (DEC) в шестнадцатеричную систему счисления. Перевод выполнить до трех значащих цифр.



В данном примере процедура перевода также прервана на 3-ей итерации.

Таким образом, $0,847 = 0,D8D_{16}$.

Полученный результат в шестнадцатеричной системе счисления является правильной дробью и не соответствует исходному числу представленному в DEC, что обусловлено проведением процедуры перевода правильной дроби до трех знаков.

Пример 2.29. Выполнить перевод числа 0,847 из десятичной в восьмеричную систему счисления. Перевод выполнить до четырех значащих цифр.

Применим алгоритм преобразования для правильной дроби (дробной части числа):

$$\begin{array}{r}
 \times 0,847 \\
 \hline
 8 \\
 \hline
 6,776 \\
 \leftarrow 6 \\
 \\
 \times 0,776 \\
 \hline
 8 \\
 \hline
 6,208 \\
 \leftarrow 6 \\
 \\
 \times 0,208 \\
 \hline
 8 \\
 \hline
 1,664 \\
 \leftarrow 1 \\
 \\
 \times 0,664 \\
 \hline
 8 \\
 \hline
 5,312 \\
 \leftarrow 5 \\
 \hline
 \hline
 \end{array}$$

0,6615₈ – результирующее число

В данном примере процедура перевода прервана. Результат $0,847 = 0,6615_8$.

Расхождение полученного результата с исходным при использовании представленного алгоритма вызвано тем, что процедура перевода в восьмеричную систему счисления была прервана на четвертой итерации, т.к. не имеет смысла осуществлять преобразование дальше, поскольку задана точность исходного числа – три знака.

Пример 2.30. Выполнить перевод правильной дроби 0,167 из десятичной в шестнадцатеричную систему счисления. Перевод выполнить до трех значащих цифр.

$$\begin{array}{r}
 \times 0,167 \\
 \hline
 16 \\
 \hline
 2,672 \\
 \leftarrow 2 \\
 \\
 \times 0,672 \\
 \hline
 16 \\
 \hline
 10,752 \\
 \leftarrow A \\
 \\
 \times 0,752 \\
 \hline
 16 \\
 \hline
 12,032 \\
 \leftarrow C \\
 \\
 \times 0,032 \\
 \hline
 16 \\
 \hline
 0,512 \\
 \leftarrow 0 \\
 \hline
 \hline
 \end{array}$$

0,2AC – результирующее число.

В данном примере процедура перевода прервана на четвертой итерации.

Таким образом, $0,167 = 0,2AC_{16}$.

Расхождение полученного числа в шестнадцатеричной системе счисления с исходной правильной дробью в десятичной системе счисления вызвано тем, что процедура преобразования правильной дроби была прервана на четвертой итерации.

Алгоритм 2д. Из двоичной (BIN), восьмеричной (OCT) и шестнадцатеричной (HEX) систем счисления – в десятичную (DEC)

В исходной правильной дроби вводится нумерация всех разрядов разрядной сетки, в которых расположены номиналы всех цифр: a_{n-1}^{n-1} – старший и a_{-m}^{-m} – младший разряд.

$$\underbrace{a_{n-1} \dots a_i \dots a_1 a_0}_{\text{целая часть числа}} \underbrace{a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m}}_{\text{дробная часть числа}} \Leftrightarrow [0, a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m}]$$

В этом случае рассчитывается номинальное значение дробной части числа по формуле разложения для правильной дроби ($Z^{\text{правильная дробь}}$):

$$Z = \underbrace{a_{n-1}P^{n-1} + a_{n-2}P^{n-2} + \dots + a_1P^1 + a_0P^0}_{\text{разложение целой части числа}} + \underbrace{a_{-1}P^{-1} + a_{-2}P^{-2} + \dots + a_{-m}P^{-m}}_{\text{разложение правильной дроби}},$$

$$Z = Z^{\text{правильная_дробь}} = \sum_{j=-1}^{-m} a_j P^j .$$

Последовательность преобразования исходной правильной дроби следующая:

- исходная правильная дробь (дробная часть числа – $Z^{\text{правильная дробь}}$) в P-ичной системе счисления записывается поразрядно, а все разряды разрядной сетки заданного числа нумеруются слева направо от старшего (-1) до младшего (-m);
- выделяется основание системы счисления (P), в которой представлено исходное число ($Z^{\text{правильная дробь}}$) – правильная дробь или дробная часть числа:
 - для двоичного числа (BIN) – P=2;
 - для восьмеричного (OCT) – P=8;
 - для шестнадцатеричного (HEX) – P=16;
- применяется формула разложения для дробной части числа ($Z^{\text{правильная дробь}}$) для получения эквивалентного номинального значения исходного P-ичного числа (правильная дробь) в десятичной системе счисления (DEC);
- номинальное значение цифры числа (a_{-1}), которая расположена в старшем разряде (минус первый разряд) умножается на номинал основания системы счисления (P=2, 8, 16) возведенный в показатель степени, который соответствует номеру разряда цифры числа (-1), взятой в первом множителе (a_{-1});
- и так далее, номинальное значение цифры числа (a_{-j}), которая расположена в -j-м разряде умножается на номинал основания системы счисления (P) возведенный в показатель степени, который соответствует номеру разряда цифры числа (-j), взятой в первом множителе (a_{-j}) j-го произведения, и так далее;
- номинальное значение цифры числа (a_{-m}), которая расположена в -m-м разряде умножается на номинал основания системы счисления (P) возведенный в показатель степени, который соответствует номеру разряда цифры числа (-m), взятой в первом множителе (a_{-m}) последнего (-m-го) произведения.
- осуществляется перемножение всех слагаемых (первый множитель a_j и второй множитель P^j), а затем суммирование полученных результатов при $j \in [-1, -m]$;
- полученный результат суммирования указанных произведений соответствует номинальному значению правильной дроби (дробной части числа) в DEC.

Пример 2.31. Выполнить перевод правильной дроби $0,1101_2$ из двоичной системы счисления ($P=2$) в десятичную (DEC) с точностью до четырех знаков.

Имеем:

$$0,1101_2 = 1*2^{-1} + 1*2^{-2} + 0*2^{-3} + 1*2^{-4} = 0,5 + 0,25 + 0 + 0,0625 = 0,8125.$$

Таким образом, $0,1101_2 = 0,8125$.

Преобразование правильной дроби из BIN в DEC завершено.

Пример 2.32. Выполнить перевод правильной дроби $0,101_2$ из двоичной системы счисления ($P=2$) в десятичную (DEC) с точностью до четырех знаков.

Имеем:

$$0,101_2 = 1*2^{-1} + 0*2^{-2} + 1*2^{-3} = 0,5 + 0 + 0,125 = 0,625.$$

Таким образом, $0,101_2 = 0,625$.

Преобразование правильной дроби из BIN в DEC завершено.

Пример 2.33. Выполнить перевод правильной дроби $0,7543_8$ из восьмеричной системы счисления ($P=8$) в десятичную (DEC) с точностью до четырех знаков.

Имеем:

$$0,7543_8 = 7*8^{-1} + 5*8^{-2} + 4*8^{-3} + 3*8^{-4} = 7*0,125 + 5*0,015625 + 4*0,0019531 + 3*0,0002441 = 0,875 + 0,078125 + 0,0078124 + 0,0007323 = 0,9616697$$

Таким образом, $0,7543_8 = 0,9616(697) = 0,9617$.

Преобразование правильной дроби из OCT в DEC завершено.

Пример 2.34. Выполнить перевод правильной дроби $0,123_8$ из восьмеричной системы счисления ($P=8$) в десятичную (DEC) с точностью до четырех знаков.

Имеем:

$$0,123_8 = 1*8^{-1} + 2*8^{-2} + 3*8^{-3} = 1*0,125 + 2*0,015625 + 3*0,0019531 = 0,125 + 0,03125 + 0,0058593 = 0,1621093$$

Таким образом, $0,123_8 = 0,1621(093) = 0,1621$.

Преобразование правильной дроби из OCT в DEC завершено.

Пример 2.35. Выполнить перевод числа $0,D8D_{16}$ из шестнадцатеричной системы счисления (HEX) в десятичную (DEC).

Имеем:

$$0,D8D_{16} = 13*16^{-1} + 8*16^{-2} + 13*16^{-3} = 13*0,0625 + 8*0,003906 + 13*0,000244 = 0,84692.$$

Таким образом, $0,D8D_{16} = 0,84692$.

Преобразование правильной дроби из HEX в DEC завершено.

Пример 2.36. Выполнить перевод исходного числа $0,27AB_{16}$ из шестнадцатеричной системы счисления ($P=16$) в десятичную (DEC).

Имеем:

$$0,27AB_{16} = 2*16^{-1} + 7*16^{-2} + 10*16^{-3} + 11*16^{-4} = 2*0,0625 + 7*0,0039063 + 10*0,0002441 + 11*0,0000153 = 0,125 + 0,0273441 + 0,002441 + 0,0001683 = 0,1549534.$$

Таким образом, $0,27AB_{16} = 0,1549534$.

Преобразование правильной дроби из HEX в DEC завершено.

Алгоритм 3д. Из двоичной (BIN) системы счисления в шестнадцатеричную (HEX)

Первый шаг

В исходной правильной дроби выделяются двоичные тетрады (группа из четырех двоичных разрядов) слева направо относительно десятичной запятой.

Если количество цифр в дробной части исходного двоичного числа не кратно 4, то оно дополняется справа незначащими нулями до достижения кратности четырем.

Второй шаг

Каждая двоичная тетрада заменяется эквивалентной шестнадцатеричной цифрой (цифры 10, 11, 12, 13, 14, 15 заменяются соответственно символами-цифрами А, В, С, D, E, F), а для вспомогательных целей используется табл. 2.2, которая обеспечивает выявление соответствий между числами представленными в разных системах счисления.

Пример 2.37. Выполнить перевод числа $0,1101_2$ из двоичной (BIN) системы счисления в шестнадцатеричную (HEX).

Имеем:

$$0,1101_2 = 0,1101_2$$

→ первая и единственная двоичная тетрада (-1-й разряд)

В соответствии с табл. 2.2 $1101_2 = D_{16}$. Тогда имеем $0,1101_2 = 0,D_{16}$.

Пример 2.38. Выполнить перевод числа $0,0010101_2$ из двоичной (BIN) системы счисления в шестнадцатеричную (HEX).

Поскольку количество цифр дробной части не кратно четырем, добавим справа незначащий ноль:

$$0,0010101_2 = 0,00101010_2.$$

→ вторая тетрада (-2 разряд дробной части в HEX)

→ первая тетрада (-1 разряд дробной части в HEX)

В соответствии с табл. 2.2 $0010_2 = 2_{16}$ и $1010_2 = A_{16}$.

Тогда имеем $0,0010101_2 = 0,2A_{16}$.

Пример 2.39. Выполнить перевод числа $0,1101110101011101_2$ из двоичной (BIN) системы счисления в шестнадцатеричную (HEX).

Имеем:

$$0,1101110101011101_2$$

→ четвертая двоичная тетрада (-4 разряд) – D_{16}

→ третья двоичная тетрада (-3 разряд дробной части в HEX) – 5_{16}

→ вторая двоичная тетрада (-2 разряд дробной части в HEX) – D_{16}

→ первая двоичная тетрада (-1 разряд дробной части числа в HEX) – D_{16}

В соответствии с табл. 2.2: $1101_2 = D_{16}$.

Тогда имеем $0,1101110101011101_2 = 0,DD5D_{16}$.

Алгоритм 4д. Из шестнадцатеричной (HEX) системы счисления в двоичную (BIN)

Алгоритм позволяет реализовать непосредственное эквивалентное преобразование правильной дроби представленной в шестнадцатеричной системе счисления (HEX) в двоичную систему счисления (BIN) за несколько итераций (шагов).

Первый шаг

Каждая цифра исходной правильной дроби в шестнадцатеричной системе счисления заменяется тетрадой двоичных разрядов в двоичной системе счисления.

Для целей преобразования используется вспомогательная табл. 2.2.

Второй шаг

Незначащие нули в полученной двоичной правильной дроби, которые расположены справа в двоичной комбинации отбрасываются и не учитываются.

Пример 2.40. Выполнить перевод числа $0,2A_{16}$ из шестнадцатеричной (HEX) системы счисления в двоичную (BIN).

По табл. 2.2 имеем $2_{16} = 0010_2$ и $A_{16} = 1010_2$.

Тогда $0,2A_{16} = 0,00101010_2$.

Отбросим в полученной правильной дроби незначащий ноль справа и сформируем окончательный результат: $0,2A_{16} = 0,0010101_2$.

Пример 2.41. Выполнить перевод числа $0,ABCD_{16}$ из шестнадцатеричной (HEX) системы счисления в двоичную (BIN).

По табл. 2.2 имеем: $A_{16} = 1010_2$, $B_{16} = 1011_2$, $C_{16} = 1100_2$, $D_{16} = 1101_2$.

Тогда $0,ABCD_{16} = 0,1010101111001101_2$.

Получим окончательный результат в следующем виде:

$0,ABCD_{16} = 0,1010101111001101_2$.

Пример 2.42. Выполнить перевод числа $0,A5B7_{16}$ из шестнадцатеричной (HEX) системы счисления в двоичную (BIN).

По табл. 2.2 имеем: $A_{16} = 1010_2$, $5_{16} = 0101_2$, $B_{16} = 1011_2$.

Тогда $0,A5B7_{16} = 0,101001011011_2$.

Получим окончательный результат в следующем виде:

$0,A5B7_{16} = 0,101001011011_2$.

Пример 2.43. Выполнить перевод числа $0,FEDC_{16}$ из шестнадцатеричной (HEX) системы счисления в двоичную (BIN).

По табл. 2.2 имеем: $F_{16} = 1111_2$, $E_{16} = 1110_2$, $D_{16} = 1101_2$, $C_{16} = 1100_2$.

Тогда $0,FEDC_{16} = 0,1111110110111(00)_2$.

Отбросим в полученной правильной дроби два незначащих нуля справа и сформируем окончательный результат: $0,FEDC_{16} = 0,1111110110111_2$.

Алгоритм 5д. Из двоичной (BIN) системы счисления в восьмеричную (ОСТ)

Для преобразования правильной дроби из двоичной системы счисления в восьмеричную необходимо выполнить следующую последовательность действий.

Первый шаг

Исходная правильная дробь разбивается на двоичные триады (группа из трех двоичных разрядов) слева направо от -1-го разряда до -m-го разряда.

Если в последовательности двоичных разрядов, которые содержат цифры правильной дроби в двоичной системе счисления не достигается кратность трем, то необходимо осуществить дополнение незначащими нулями справа до достижения кратности трем и образования двоичной триады (соответствует цифре числа в ОСТ).

Второй шаг

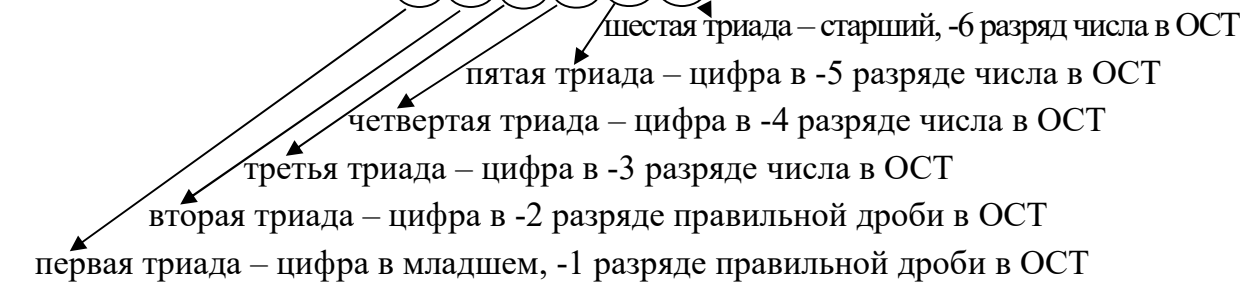
Каждая двоичная триада исходной двоичной правильной дроби преобразуется в цифру числа, которое представлено в восьмеричной (ОСТ) системе счисления.

Для вспомогательных целей преобразования используется табл. 2.2.

Пример 2.44. Выполнить перевод правильной дроби $0,1001101110100001_2$ из двоичной (BIN) в восьмеричную (ОСТ) систему счисления.

Дополним двоичную последовательность незначащим нулем справа.

$$0,1001101110100001_2 = 0,010011011101000010_2$$



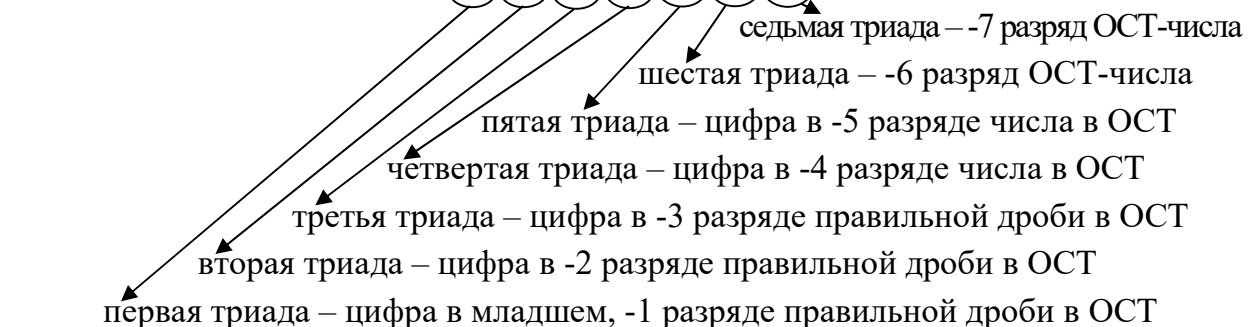
Воспользуемся табл. 2.2: $010_2 = 2_8$, $011_2 = 3_8$, $101_2 = 5_8$, $000_2 = 0_8$, $010_2 = 2_8$.

Удалим незначащий ноль справа, получим $0,01001101110100001_2 = 0,233502_8$.

Пример 2.45. Выполнить перевод правильной дроби $0,1101010010001111001_2$ из двоичной (BIN) в восьмеричную (ОСТ) систему счисления.

Дополним двоичную последовательность двумя незначащими нулями справа.

$$0,1101010010001111001_2 = 0,110101001000111100100_2$$



Воспользуемся табл. 2.2: $110_2 = 6_8$, $101_2 = 5_8$, $001_2 = 1_8$, $000_2 = 0_8$, $111_2 = 7_8$, $100_2 = 4_8$,

Удалим незначащие нули справа, получим $0,1101010010001111001_2 = 0,6510744_8$.

Алгоритм бд. Из восьмеричной (ОСТ) системы счисления в двоичную (BIN)

Для преобразования правильной дроби из восьмеричной системы счисления в двоичную необходимо выполнить определенную последовательность действий.

Первый шаг

Каждая цифра правильной дроби в восьмеричной системе счисления записывается в виде двоичной триады, которая представляет собой три двоичных разряда.

Второй шаг

Двоичные триады записываются в виде единой двоичной комбинации.

Незначащие нули, которые располагаются в результирующей двоичной комбинации справа отбрасываются, поскольку они не влияют на номинальное значение числа.

Пример 2.46. Выполнить перевод правильной дроби $0,13742_8$ из восьмеричной (ОСТ) в двоичную (BIN) систему счисления.

По табл. 2.2. имеем: $1_8 = 001_2$, $3_8 = 011_2$, $7_8 = 111_2$, $4_8 = 100_2$, $2_8 = 010_2$, а после удаления незначащего нуля справа в результирующей двоичной комбинации $0,001011111100010\cancel{0}_b$ имеем окончательный результат в виде:

$$0,13742_8 = 0,00101111110001_2.$$

Пример 2.47. Выполнить перевод правильной дроби $0,345432_8$ из восьмеричной (ОСТ) в двоичную (BIN) систему счисления.

По табл. 2.2. имеем: $3_8 = 011_2$, $4_8 = 100_2$, $5_8 = 101_2$, $4_8 = 100_2$, $3_8 = 011_2$, $2_8 = 010_2$, а после удаления незначащего нуля справа в результирующей двоичной комбинации $0,01110010110001101\cancel{0}_b$ имеем окончательный результат в виде:

$$\text{Тогда } 0,345432_8 = 0,01110010110001101_2.$$

Пример 2.48. Выполнить перевод правильной дроби $0,37467_8$ из восьмеричной (ОСТ) в двоичную (BIN) систему счисления.

По табл. 2.2. имеем: $3_8 = 011_2$, $7_8 = 111_2$, $4_8 = 100_2$, $6_8 = 110_2$, $7_8 = 111_2$, а после объединения всех двоичных триад, получим окончательный результат: $0,011111100110111_b$.

$$\text{Тогда } 0,37467_8 = 0,011111100110111_2.$$

Пример 2.49. Выполнить перевод правильной дроби $0,27543_8$ из восьмеричной (ОСТ) в двоичную (BIN) систему счисления.

По табл. 2.2. имеем: $2_8 = 010_2$, $7_8 = 111_2$, $5_8 = 101_2$, $4_8 = 100_2$, $3_8 = 011_2$, а затем после объединения всех двоичных триад, получим окончательный результат в следующем виде: $0,010111101100011_b$.

$$\text{Тогда } 0,27543_8 = 0,010111101100011_2.$$

2.3.3. Правила перевода дробных чисел

Преобразование дробных чисел имеет определенную специфику, которая зависит от вида и способа представления заданного числа в определенной системе счисления.

Для преобразования дробных чисел из одной системы счисления в другую используется комбинация алгоритмов для целой части (литера ц) и правильной дроби (литера д):

- для целой части числа – исходным числом для осуществления преобразования является целое число, а дробная часть при этом равна нулю:
 - для преобразования целой части числа из десятичной системы счисления (DEC, P=10) в двоичную (BIN, P=2), восьмеричную (OCT, P=8) и шестнадцатеричную (HEX, P=16) рекомендуется использовать **алгоритм 1ц**;
 - для преобразования целой части числа в десятичную систему счисления (DEC, P=10) из двоичной (BIN, P=2), восьмеричной (OCT, P=8) и шестнадцатеричной (HEX, P=16) рекомендуется применять **алгоритм 2ц**;
 - для преобразования целой части числа из двоичной системы счисления (BIN, P=2) в шестнадцатеричную (HEX, P=16) применяют **алгоритм 3ц**;
 - для преобразования целой части числа из шестнадцатеричной (HEX, P=16) в двоичную систему счисления (BIN, P=2) используют **алгоритм 4ц**;
 - для преобразования целой части числа из двоичной системы счисления (BIN, P=2) в восьмеричную (OCT, P=8) применяют **алгоритм 5ц**;
 - для преобразования целой части числа из восьмеричной системы счисления (OCT, P=8) в двоичную (BIN, P=2) используют **алгоритм 6ц**;
- для дробной части числа – исходным числом для осуществления процедуры преобразования выступает правильная дробь, целая часть равна нулю:
 - для преобразования дробной части числа из десятичной системы счисления (DEC, P=10) в двоичную (BIN, P=2), восьмеричную (OCT, P=8) и шестнадцатеричную (HEX, P=16) рекомендуется использовать **алгоритм 1д**;
 - для преобразования дробной части числа в десятичную систему счисления (DEC, P=10) из двоичной (BIN, P=2), восьмеричной (OCT, P=8), шестнадцатеричной (HEX, P=16) требуется применять **алгоритм 2д**;
 - для преобразования дробной части числа из двоичной системы счисления (BIN, P=2) в шестнадцатеричную (HEX, P=16) применяют **алгоритм 3д**;
 - для преобразования дробной части числа из шестнадцатеричной (HEX, P=16) в двоичную систему счисления (BIN, P=2) используют **алгоритм 4д**;
 - для преобразования дробной части числа из двоичной системы счисления (BIN, P=2) в восьмеричную (OCT, P=8) применяют **алгоритм 5д**;
 - для преобразования дробной части числа из восьмеричной системы счисления (OCT, P=8) в двоичную (BIN, P=2) используют **алгоритм 6д**.

После применения перечисленных алгоритмов преобразования из первой и второй группы необходимо осуществить сложение двух полученных результатов.

Некоторые алгоритмы преобразования сложны, поэтому необходимо производить проверку полученного результата, чтобы исключить возможные ошибки.

Пример 2.50. Выполнить перевод дробного числа 195,748 из десятичной системы счисления (DEC) в двоичную (BIN). Необходимо обеспечить заданную точность преобразования – три значащие цифры (разряда) после десятичной запятой.

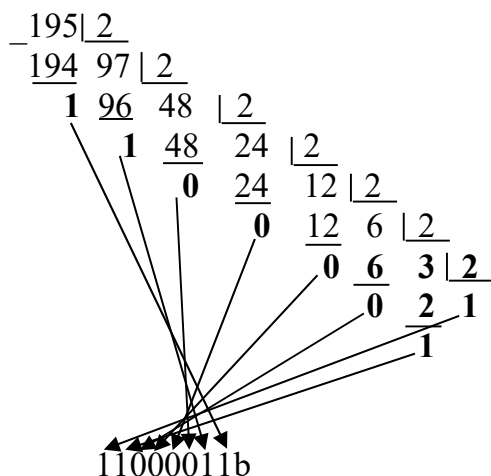
Основание системы счисления, в которую осуществляется перевод равно 2.

Представим исходное число как сумму целого числа и правильной дроби:

$$195,748 = 195 + 0,748.$$

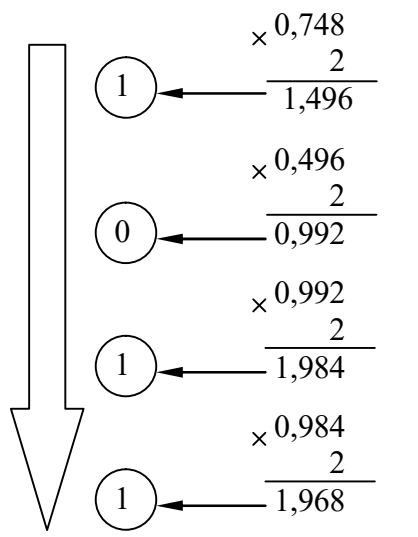
На первом этапе воспользуемся алгоритмом 1ц.

Выполнить перевод числа 195 в двоичную систему счисления:



На втором этапе воспользуемся алгоритмом 1д.

Выполнить перевод правильной дроби 0,748 в двоичную систему счисления:



0,1011 – результирующее число

Выполним сложение двух результатов, которые получены при помощи двух алгоритмов преобразования (для целой части и для дробной частей числа):

$$195 + 0,748 = 11000011_2 + 0,1011_2 = 11000011,1011_2.$$

Таким образом, $195,748 = 11000011,1011_2$.

Пример 2.51. Выполнить перевод дробного числа 195,748 из десятичной системы счисления (DEC) в шестнадцатеричную (HEX). Точность преобразования – три значащие цифры (разряда) после десятичной запятой.

Основание системы счисления, в которую осуществляется перевод равно 2.

Представим исходное число как сумму целого числа и правильной дроби:

$$195,748 = 195 + 0,748.$$

На первом этапе воспользуемся алгоритмом 1ц.

Выполнить перевод числа 195 в шестнадцатеричную систему счисления:

$$\begin{array}{r} \underline{195} \mid \underline{16} \\ \underline{192} \quad 12 \\ \hline 3 \end{array} \quad \begin{array}{l} \swarrow \\ \searrow \end{array} \quad 195_{10} = C3_{16}$$

Для преобразования целой части числа использовалась вспомогательная таблица соответствия между системами счисления (табл. 2.2): $12_{10} = C_{16}$

На втором этапе воспользуемся алгоритмом 1д.

Выполнить перевод правильной дроби 0,748 в шестнадцатеричную систему счисления:

$$\begin{array}{r} \times 0,748 \\ \quad \underline{16} \\ \hline V \leftarrow (11) \leftarrow 11,968 \\ \\ \times 0,968 \\ \quad \underline{16} \\ \hline F \leftarrow (15) \leftarrow 15,488 \\ \\ \times 0,488 \\ \quad \underline{16} \\ \hline 7 \leftarrow (7) \leftarrow 7,808 \\ \\ \times 0,808 \\ \quad \underline{16} \\ \hline C \leftarrow (12) \leftarrow 12,928 \end{array}$$

$0,BF7C$ – результирующее число

Выполним сложение двух результатов, которые получены при помощи двух алгоритмов преобразования:

$$195 + 0,748 = C3_{16} + 0,BF7C_{16} = C3,BF7C_{16}.$$

Для преобразования целой части числа использовалась вспомогательная таблица соответствия между различными системами счисления (табл. 2.2): $11_{10} = B_{16}$, $12_{10} = C_{16}$, $15_{10} = F_{16}$.

Таким образом, $195,748 = C3,BF7C_{16}$.

2.4. Правила выполнения простейших арифметических операций

Арифметические операции реализуются на аппаратном уровне в арифметико-логическом устройстве, которое располагается в основе центрального процессора компьютера, а также подразделяются на мультипликативные (умножение и деление) и аддитивные (сложение и вычитание), при этом основными операциями являются сложение и умножение, производными выступают вычитание и деление.

Особенности аппаратной реализации компонентов архитектуры ЭВМ обусловили приведение операций умножения и деления к операциям сложения и вычитания, а в дальнейшем сведение операции арифметического вычитания к операции сложения.

Для выполнения операций сложения, вычитания, умножения и деления над числами в системах счисления с основанием $P > 10$ используют таблицы сложения и умножения для определенной системы счисления ($P=2, 8, 16$), что связано со сложностью отслеживания переноса и заема из старшего разряда в младший и наоборот.

Арифметические операции (рис. 2.10) над двоичными (BIN) и шестнадцатеричными (HEX) числами выполняются по тем же правилам, что и для десятичных чисел (DEC).

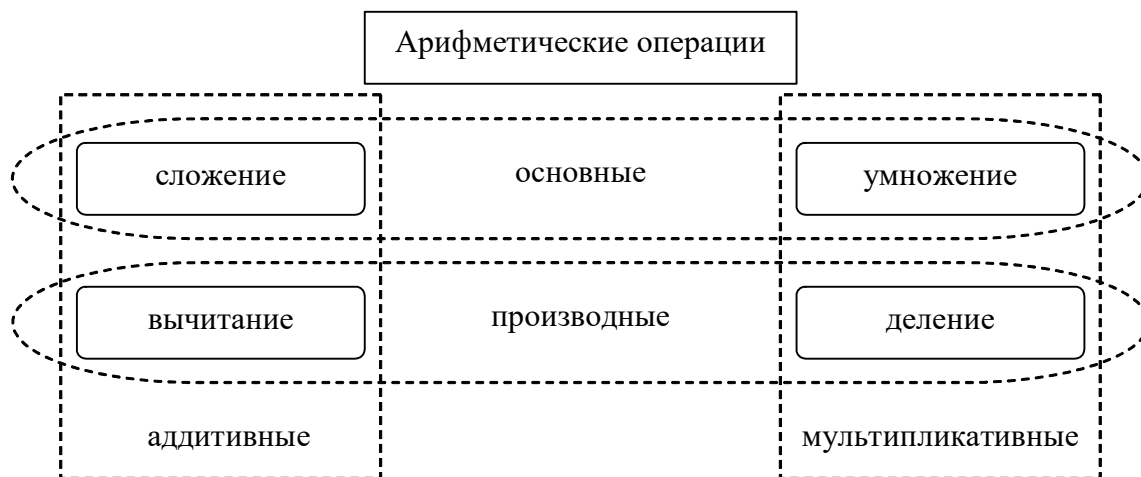


Рисунок 2.10. Классификация основных арифметических операций

Рассмотрим особенности выполнения основных и производных арифметических операций: сложение, вычитание, умножение и деление для целой и дробной частей.

Операция сложения в определенной системе счисления выполняется посредством использования метода «в столбик»: оба слагаемых записываются друг под другом, поразрядно складываются, а при достижении переполнения в младшем разряде осуществляется перенос соответствующего номинала в старший разряд.

При выполнении операции вычитания уменьшаемое и вычитаемое записываются друг под другом, если номинальное значение цифры в младшем разряде первого числа меньше номинала цифры в младшем разряде второго числа, то выполняется заем из следующего старшего разряда находящегося в первом числе.

Операция умножения выполняется по методу «в столбик» и сводится к операции сложения для двух, трех и более чисел, которые выступают слагаемыми.

Операция деления осуществляется по методу «угол» и приводится к операции вычитания после подбора частного от деления и формировании остатка на итерации.

2.4.1. Правила сложения

При осуществлении операции арифметического сложения необходимо отслеживать перенос из младшего в старший разряд, что существенно усложняется с ростом основания системы счисления, в которой представлены слагаемые данной операции.

$$\begin{array}{cccccccc}
 + d_n & & + d_2 + d_0 & & + d_{-(m-2)} & & & \\
 & + d_{i+1} & + d_1 & + d_{-(j-1)} & + d_{-(m-1)} & & & \\
 \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
 & n-1 & i & 1 & 0 & -1 & -j & -(m-1) & -m \\
 a_{n-1} \dots a_i \dots a_1 a_0, & a_{-1} \dots a_{-j} \dots a_{-(m-1)} & a_{-m} & & & & & & \\
 \sum & b_{n-1} \dots b_i \dots b_1 b_0, & b_{-1} \dots b_{-j} \dots b_{-(m-1)} & b_{-m} & & & & & \\
 & n-1 & i & 1 & 0 & -1 & -j & -(m-1) & -m \\
 c_{n-1} \dots c_i \dots c_1 c_0, & c_{-1} \dots c_{-j} \dots c_{-(m-1)} & c_{-m} & & & & & &
 \end{array}$$

Алгоритм сложения чисел в определенной системе счисления включает ряд шагов.

Первый шаг

Исходные дробные числа включают явную целую и дробную части записываются друг под другом, выступают в роли первого $(a_{n-1} \dots a_i \dots a_1 a_0, a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m})$

и второго слагаемых $(b_{n-1} \dots b_i \dots b_1 b_0, b_{-1} \dots b_{-j} \dots b_{-(m-1)} b_{-m})$ при осуществлении операции арифметического сложения, а в результате формируется сумма двух представленных

чисел А и В в следующем виде: $(c_{n-1} \dots c_i \dots c_1 c_0, c_{-1} \dots c_{-j} \dots c_{-(m-1)} c_{-m})$ – цифры числа.

Второй шаг

Осуществляется итеративное поразрядное сложение чисел А и В справа налево от младшего разряда (-m-й) к старшему (n-1-й), а если длина последовательности цифр в целой или дробной частях двух чисел различна, то осуществляется дополнение незначащими нулями слева для целого числа и справа для правильной дроби:

- осуществляется поразрядное сложение двух дробных частей исходных чисел:
 - складываются номиналы младшего (-m-го) разряда первого (a_{-m}^{-m}) и второго слагаемого (b_{-m}^{-m}) , формируются: новое значение в младшем разряде результирующего числа (c_{-m}^{-m}) и номинал для переноса в -(m-1)-й разряд $(+d_{-(m-1)})$;
 - складываются номинальные значения -(m-1)-го разряда первого $(a_{-(m-1)}^{-(m-1)})$ и второго слагаемого $(b_{-(m-1)}^{-(m-1)})$, а также номинал для учета переноса $(+d_{-(m-1)})$ на предыдущей итерации, формируются: новое значение в младшем разряде результирующего числа $(c_{-(m-1)}^{-(m-1)})$ и номинал для переноса в следующий разряд $(+d_{-(m-2)})$;
 - складываются номиналы -j-го разряда первого (a_{-j}^{-j}) и второго слагаемого (b_{-j}^{-j}) , а также учитывается номинал переноса $(+d_{-j})$ на предыдущей ите-

- рации, формируются: новое значение в младшем разряде результирующего числа (c_{-j}^{-j}) и номинал для учета переноса в $-(j-1)$ -й разряд ($+d_{-(j-1)}$);
- складываются номиналы -1 -го разряда первого (a_{-1}^{-1}) и второго слагаемого (b_{-1}^{-1}), а также учитывается номинал переноса ($+d_{-1}$) на предыдущей итерации, при этом формируются: новое значение в младшем разряде результирующего числа (c_{-1}^{-1}) и номинал для учета переноса ($+d_0$) в младший (0-й) разряд целой части представленного дробного числа;
 - осуществляется поразрядное сложение двух целых частей исходного числа:
 - складываются номиналы нулевого разряда первого (a_0^0) и второго слагаемого (b_0^0), а также учитывается номинал переноса ($+d_0$) на предыдущей итерации, формируются: новое значение в младшем разряде результирующего числа (c_0^0) и номинал для переноса в следующий (1-й) разряд ($+d_1$);
 - складываются номиналы первого разряда первого (a_1^1) и второго слагаемых (b_1^1), а также учитывается номинал переноса ($+d_1$) на предыдущей итерации, формируются: новое значение в младшем разряде результирующего числа (c_1^1) и номинал для переноса в следующий (2-й) разряд ($+d_2$);
 - складываются номиналы i -го разряда первого (a_i^i) и второго слагаемых (b_i^i), а также учитывается номинал переноса ($+d_i$) на предыдущей итерации, формируются: новое значение в младшем разряде результирующего числа (c_i^i) и номинал для переноса в следующий ($i+1$ -й) разряд ($+d_{i+1}$);
 - складываются номиналы старшего ($n-1$ -го) разряда первого (a_{n-1}^{n-1}) и второго слагаемых (b_{n-1}^{n-1}), а также учитывается номинал переноса ($+d_{n-1}$) на предыдущей итерации, формируются: новое значение в младшем разряде результирующего числа (c_{n-1}^{n-1}), а также номинал для учета переноса и дополнения комбинации слева относительно старшего ($n-1$ -го) разряда ($+d_n$).

Третий шаг

Формируется и записывается результат сложения целой и дробной частей числа.

Для осуществления арифметической операции сложения чисел в двоичной (BIN), восьмеричной (OCT) и шестнадцатеричной (HEX) системах счисления необходимо использовать таблицы сложения для формирования номинала в младшем разряде и отслеживания переноса в старший разряд, которые представлены в табл. 2.3, 2.4, 2.5.

Таблица 2.3

Таблица сложения чисел для двоичной системы счисления

Знак операции «+»		Значение первого слагаемого	
		0	1
Значение второго слагаемого	0	0	1
	1	1	+10

Таблица 2.4

Таблица сложения чисел для восьмеричной системы счисления

Знак операции «+»		Значение первого слагаемого							
		0	1	2	3	4	5	6	7
Значение второго слагаемого	0	0	1	2	3	4	5	6	7
	1	1	2	3	4	5	6	7	+10
	2	2	3	4	5	6	7	+10	+11
	3	3	4	5	6	7	+10	+11	+12
	4	4	5	6	7	+10	+11	+12	+13
	5	5	6	7	+10	+11	+12	+13	+14
	6	6	7	+10	+11	+12	+13	+14	+15
	7	7	+10	+11	+12	+13	+14	+15	+16

Таблица 2.5

Таблица сложения чисел для шестнадцатеричной системы счисления

Знак операции «+»		Значение первого слагаемого															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Значение второго слагаемого	0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	1	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	+10
	2	2	3	4	5	6	7	8	9	A	B	C	D	E	F	+10	+11
	3	3	4	5	6	7	8	9	A	B	C	D	E	F	+10	+11	+12
	4	4	5	6	7	8	9	A	B	C	D	E	F	+10	+11	+12	+13
	5	5	6	7	8	9	A	B	C	D	E	F	+10	+11	+12	+13	+14
	6	6	7	8	9	A	B	C	D	E	F	+10	+11	+12	+13	+14	+15
	7	7	8	9	A	B	C	D	E	F	+10	+11	+12	+13	+14	+15	+16
	8	8	9	A	B	C	D	E	F	+10	+11	+12	+13	+14	+15	+16	+17
	9	9	A	B	C	D	E	F	+10	+11	+12	+13	+14	+15	+16	+17	+18
	A	A	B	C	D	E	F	+10	+11	+12	+13	+14	+15	+16	+17	+18	+19
	B	B	C	D	E	F	+10	+11	+12	+13	+14	+15	+16	+17	+18	+19	+1A
	C	C	D	E	F	+10	+11	+12	+13	+14	+15	+16	+17	+18	+19	+1A	+1B
	D	D	E	F	+10	+11	+12	+13	+14	+15	+16	+17	+18	+19	+1A	+1B	+1C
	E	E	F	+10	+11	+12	+13	+14	+15	+16	+17	+18	+19	+1A	+1B	+1C	+1D
	F	F	+10	+11	+12	+13	+14	+15	+16	+17	+18	+19	+1A	+1B	+1C	+1D	+1E

Пример 2.52. Сложить двоичные числа 1101_2 и 11011_2 .

Первое и второе слагаемые записываются друг над другом, целая часть первого слагаемого дополняется незначащим нулем слева для выравнивания длины двоичных чисел, а их разряды нумеруются: младшему разряду присваивается значение 0 (нулевой разряд), старшему разряду присваивается значение 4 (четвертый разряд).

$$\begin{array}{r} \text{номера разрядов: } \mathbf{4\ 3\ 2\ 1\ 0} \\ +0\ 1\ 1\ 0\ 1_2 \\ \underline{1\ 1\ 0\ 1\ 1_2} \end{array}$$

Результат сложения двух чисел формируется следующим образом:

- нулевой разряд формируется следующим образом: $1_2^0 + 1_2^0 = +1_2^1 0_2^0$, где 1_2^0 – номинальное значение нулевого разряда первого и второго слагаемых; $+1_2^1$ – номинальное значение, которое переносится в следующий разряд (первый разряд); 0_2^0 – номинальное значение, которое остается в нулевом разряде числа;
- первый разряд формируется следующим образом: $0_2^1 + 1_2^1 + (+1_2^1) = +1_2^2 0_2^1$, где 0_2^1 – номинальное значение в первом разряде первого слагаемого; 1_2^1 – номинальное значение в первом разряде второго слагаемого; $(+1_2^1)$ – номинальное значение, которое учитывает перенос из нулевого разряда; $+1_2^2$ – номинальное значение, которое переносится во второй разряд; 0_2^1 – номинальное значение, которое остается в первом разряде результирующего числа;
- второй разряд формируется следующим образом: $1_2^2 + 0_2^2 + (+1_2^2) = +1_2^3 0_2^2$, где 1_2^2 – номинальное значение во втором разряде первого слагаемого; 0_2^2 – номинальное значение во втором разряде второго слагаемого; $(+1_2^2)$ – номинальное значение, которое учитывает перенос из первого разряда; $+1_2^3$ – номинальное значение, которое переносится в третий разряд числа; 0_2^2 – номинальное значение, которое остается во втором разряде результирующего числа;
- третий разряд формируется следующим образом: $1_2^3 + 1_2^3 + (+1_2^3) = +1_2^4 1_2^3$, где 1_2^3 – номинальное значение в третьем разряде первого слагаемого; 1_2^3 – номинальное значение в третьем разряде второго слагаемого; $(+1_2^3)$ – номинальное значение, которое учитывает перенос из второго разряда; $+1_2^4$ – номинальное значение, которое переносится в четвертый разряд числа; 1_2^3 – номинальное значение, которое остается в третьем разряде результирующего числа;

- четвертый разряд формируется следующим образом: $0_2^4 + 1_2^4 + (+1_2^4) = +1_2^5 0_2^4$, где 0_2^4 – номинальное значение в четвертом разряде первого слагаемого; 1_2^4 – номинальное значение в четвертом разряде второго слагаемого; $(+1_2^4)$ – номинальное значение, которое учитывает перенос из третьего разряда; $+1_2^5$ – номинальное значение, которое переносится в пятый («дополненный») разряд числа; 0_2^4 – номинальное значение, которое остается в четвертом разряде числа.

Таким образом:

номера разрядов

$$\begin{array}{r}
 +1+1+1+1+1 \\
 + \quad \mathbf{4\ 3\ 2\ 1\ 0} \\
 +0\ 1\ 1\ 0\ 1_2 \\
 \underline{1\ 1\ 0\ 1\ 1_2} \\
 (+1)0\ 1\ 0\ 0\ 0
 \end{array}$$

Получен результат 101000_2 , который затем необходимо проверить.

Для этого воспользуемся формулой разложения, которая позволяет получить сумму произведений всех номинальных значений цифр взятых из определенных разрядов числа на основание системы счисления возведенное в показатель степени соответствующий номеру разряда откуда была взята цифра заданного числа:

$$Z_{\text{целое}} = \sum_{i=0}^{n-1} a_i P^i = a_0 P^0 + a_1 P^1 + \dots + a_i P^i + \dots + a_{n-1} P^{n-1}$$

Первое слагаемое раскладывается и суммируется следующим образом:

- номинальное значение цифры 1 (единица) в третьем разряде первого числа умножается на основание системы счисления 2 возведенное в показатель степени 3 соответствующий номеру разряда откуда была взята эта цифра;
- номинальное значение цифры 1 (единица) во втором разряде первого числа умножается на основание системы счисления 2 возведенное в показатель степени 2 соответствующий номеру разряда откуда была взята эта цифра;
- номинальное значение цифры 0 (ноль) в первом разряде первого числа умножается на основание системы счисления 2 возведенное в показатель степени 1 соответствующий номеру разряда откуда была взята эта цифра;
- номинальное значение цифры 1 (единица) в нулевом разряде первого числа умножается на основание системы счисления 2 возведенное в показатель степени 0.

$$1101_2 = 1*2^3 + 1*2^2 + 0*2^1 + 1*2^0 = 8 + 4 + 0 + 1 = 13;$$

Второе слагаемое раскладывается и суммируется следующим образом:

$$11011_2 = 1*2^4 + 1*2^3 + 0*2^2 + 1*2^1 + 1*2^0 = 16 + 8 + 0 + 2 + 1 = 27;$$

Третье слагаемое раскладывается и суммируется следующим образом:

$$101000_2 = 1*2^5 + 0*2^4 + 1*2^3 + 0*2^2 + 0*2^1 + 0*2^0 = 32 + 0 + 8 + 0 + 0 + 0 = 40.$$

Поскольку $13d + 27d = 40d$, то полученный результат арифметического сложения в двоичной системе счисления (BIN) является корректным.

Пример 2.54. Выполнить сложение двух восьмеричных чисел 643_8 и 3543_8 .

Первое и второе слагаемые записываются друг над другом, целая часть первого слагаемого дополняется незначащим нулем слева для выравнивания длины двух чисел, а их разряды нумеруются: младшему разряду присваивается значение 0 (нулевой разряд), старшему разряду присваивается значение 3 (третий разряд).

$$\begin{array}{r} \text{номера разрядов: } \mathbf{3\ 2\ 1\ 0} \\ +0\ 6\ 4\ 3_8 \\ \underline{3\ 5\ 4\ 3_8} \end{array}$$

Результат сложения двух восьмеричных чисел рассчитывается последовательно:

- нулевой разряд формируется следующим образом: $3_8^0 + 3_8^0 = +0\ 6_2^0$, где 3_8^0 – номинальное значение нулевого разряда первого и второго слагаемых; $+0$ – номинальное значение, которое переносится в следующий (первый) разряд; 6_2^0 – номинальное значение, которое остается в нулевом разряде числа;
- первый разряд формируется следующим образом: $4_8^1 + 4_8^1 + (0_8^1) = +1\ 0_8^1$, где 4_8^1 – номинальное значение первого разряда первого и второго слагаемых (цифр двух целых чисел); $(+0_8^1)$ – номинальное значение, которое учитывает перенос из нулевого разряда; $+1$ – номинальное значение, которое переносится во второй разряд; 0_8^1 – номинальное значение, которое остается в первом разряде результирующего восьмеричного числа;
- второй разряд формируется следующим образом: $6_8^2 + 5_8^2 + (+1_8^2) = +1\ 4_8^2$, где 6_8^2 – номинальное значение во втором разряде первого слагаемого; 5_8^2 – номинальное значение во втором разряде второго слагаемого; $(+1_8^2)$ – номинальное значение, которое учитывает перенос из первого разряда; $+1$ – номинальное значение, которое переносится в третий разряд числа; 4_8^2 – номинальное значение, которое остается во втором разряде числа;
- третий разряд формируется следующим образом: $0_8^3 + 3_8^3 + (+1_8^3) = +0\ 4_8^3$, где 0_8^3 – номинальное значение в третьем разряде первого слагаемого; 3_8^3 – номинальное значение в третьем разряде второго слагаемого; $+1_8^3$ – номинальное значение, которое учитывает перенос из второго разряда; $+0$ – номинальное значение, которое переносится в четвертый разряд числа; 4_8^3 – номинальное значение, которое остается в третьем разряде числа.

Таким образом, на основе используемого алгоритма арифметического сложения:

$$\begin{array}{r}
 \text{номера разрядов} \\
 +1+1 \\
 \mathbf{3\ 2\ 1\ 0} \\
 +0\ 6\ 4\ 3_8 \\
 \underline{3\ 5\ 4\ 3_8} \\
 4\ 4\ 0\ 6_8
 \end{array}$$

Получен результат 4406_8 , который необходимо в дальнейшем проверить.

Для этого воспользуемся формулой разложения, которая позволяет получить сумму произведений номинального значения цифры взятой из определенного разряда числа на основание системы счисления возведенное в показатель степени соответствующий номеру разряда откуда взята определенная цифра в этом числе:

$$Z_{\text{целое}} = \sum_{i=0}^{n-1} a_i P^i = a_0 P^0 + a_1 P^1 + \dots + a_i P^i + \dots + a_{n-1} P^{n-1}$$

Первое слагаемое раскладывается и суммируется следующим образом:

- номинальное значение цифры 0 (ноль) в третьем разряде первого числа умножается на основание системы счисления 8 возведенное в показатель степени 3 соответствующий номеру разряда откуда была взята эта цифра;
- номинальное значение цифры 6 (шесть) во втором разряде первого числа умножается на основание системы счисления 8 возведенное в показатель степени 2 соответствующий номеру разряда откуда была взята эта цифра;
- номинальное значение цифры 4 (четыре) в первом разряде первого числа умножается на основание системы счисления 8 возведенное в показатель степени 1 соответствующий номеру разряда откуда была взята эта цифра;
- номинальное значение цифры 3 (три) в нулевом разряде первого числа умножается на основание системы счисления 8 возведенное в показатель степени 0 соответствующий номеру разряда откуда была взята определенная цифра.

Полученные произведения $a_i P^i$ последовательно суммируются для получения результирующего значения первого слагаемого в десятичной системе счисления.

$$0643_8 = 0 \cdot 8^3 + 6 \cdot 8^2 + 4 \cdot 8^1 + 3 \cdot 8^0 = 0 + 384 + 32 + 3 = 419_{10};$$

Второе слагаемое раскладывается и суммируется аналогичным образом:

$$3543_8 = 3 \cdot 8^3 + 5 \cdot 8^2 + 4 \cdot 8^1 + 3 \cdot 8^0 = 1536 + 320 + 32 + 3 = 1891_{10};$$

Третье слагаемое раскладывается и суммируется аналогичным образом:

$$4406_8 = 4 \cdot 8^3 + 4 \cdot 8^2 + 0 \cdot 8^1 + 6 \cdot 8^0 = 2048 + 256 + 0 + 6 = 2310_{10}.$$

Поскольку $419d + 1891d = 2310d$, то результат арифметического сложения двух восьмеричных чисел в двоичной системе счисления является корректным.

Если наблюдается несоответствие между десятичным представлением первого и второго слагаемого, а также результатом суммирования, то необходимо выполнить повторный расчет результата согласно используемому алгоритму рассмотренному ранее.

2.4.2. Правила вычитания

При осуществлении операции вычитания необходимо отслеживать заем из старшего в младший разряд, что существенно усложняется с ростом основания системы счисления, в которой представлены уменьшаемое и вычитаемое операции.

$$\begin{array}{r}
 -e_n \quad -e_2 \quad -e_0 \quad -e_{-(m-2)} \\
 \begin{array}{cccccccc}
 \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
 n-1 & i & 1 & 0 & -1 & -j & -(m-1) & -m \\
 a_{n-1} \dots a_i \dots a_1 a_0, a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m} \\
 - b_{n-1} \dots b_i \dots b_1 b_0, b_{-1} \dots b_{-j} \dots b_{-(m-1)} b_{-m} \\
 \hline
 f_{n-1} \dots f_i \dots f_1 f_0, f_{-1} \dots f_{-j} \dots f_{-(m-1)} f_{-m}
 \end{array}
 \end{array}$$

Алгоритм вычитания двух чисел, которые представлены в определенной системе счисления включает последовательность шагов вычислительной процедуры.

Первый шаг

Исходные дробные числа включают явную целую и дробную части, записываются непосредственно друг под другом, выступают в роли уменьшаемого

$$(a_{n-1} \dots a_i \dots a_1 a_0, a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m})$$

и вычитаемого

$$(b_{n-1} \dots b_i \dots b_1 b_0, b_{-1} \dots b_{-j} \dots b_{-(m-1)} b_{-m})$$

Второй шаг

Осуществляется поразрядное вычитание двух чисел справа налево от младшего разряда к старшему, если длина последовательности цифр в целой или дробной частях уменьшаемого и вычитаемого различна, то осуществляется дополнение незначащими нулями слева для целой (целое число) и справа для дробной частей (правильная дробь) с целью выравнивания длины всех цифр двух заданных чисел:

- осуществляется поразрядное вычитание дробных частей 2^x исходных чисел:
 - вычитаются номиналы младшего разряда ($-m$ -го) уменьшаемого (a_{-m}^{-m}) и вычитаемого (b_{-m}^{-m}), формируются: номинал заема из $(m-1)$ -го разряда ($-e_{-(m-1)}$) и новое значение в младшем разряде результирующего числа (f_{-m}^{-m});
 - вычитаются номинальные значения $-(m-1)$ -го разряда уменьшаемого ($a_{-(m-1)}^{-(m-1)}$) и вычитаемого ($b_{-(m-1)}^{-(m-1)}$) с учетом номинала заема ($-e_{-(m-1)}$) на предыдущей итерации, формируются: номинал заема из $-(m-2)$ -го разряда ($-e_{-(m-2)}$) и новое значение в младшем разряде результирующего числа ($f_{-(m-1)}^{-(m-1)}$);
 - вычитаются номиналы $-j$ -го разряда уменьшаемого (a_{-j}^{-j}) и вычитаемого

- (b_{-j}^{-j}) с учетом номинала заема из $-j$ -го разряда $(-e_{-j})$ на предыдущей итерации, формируются: номинал для учета заема из $-(j-1)$ -го разряда $(-e_{-(j-1)})$ и новое значение в младшем разряде результирующего числа (f_{-j}^{-j}) ;
- складываются номиналы -1 -го разряда уменьшаемого (a_{-1}^{-1}) и вычитаемого (b_{-1}^{-1}) с учетом номинала заема $(-e_{-1})$ на предыдущей итерации, формируются: номинал для учета заема из старшего (нулевого) разряда $(-e_0)$ и новое значение в младшем разряде результирующего числа (f_{-1}^{-1}) ;
 - осуществляется поразрядное вычитание двух целых частей исходного числа:
 - вычитаются номиналы нулевого разряда уменьшаемого (a_0^0) и вычитаемого (b_0^0) с учетом номинала заема $(-e_0)$ на предыдущей итерации, формируются: номинал для заема из следующего (первого) разряда $(-e_1)$ и новое значение в младшем разряде результирующего числа (f_0^0) ;
 - вычитаются номиналы первого разряда уменьшаемого (a_1^1) и вычитаемого (b_1^1) с учетом номинала заема $(-e_1)$ на предыдущей итерации, формируются: номинал для заема из следующего (2-го) разряда $(-e_2)$ и новое значение в младшем разряде результирующего числа (f_1^1) ;
 - вычитаются номиналы i -го разряда уменьшаемого (a_i^i) и вычитаемого (b_i^i) с учетом номинала заема $(-e_i)$ на предыдущей итерации, формируются: номинал для заема из следующего ($i+1$ -го) разряда $(-e_{i+1})$ и новое значение в младшем разряде результирующего числа (f_i^i) ;
 - вычитаются номиналы старшего ($n-1$ -го) разряда уменьшаемого (a_{n-1}^{n-1}) и вычитаемого (b_{n-1}^{n-1}) с учетом номинала заема $(-e_{n-1})$ на предыдущей итерации, формируются: новое значение в младшем разряде результирующего числа (f_{n-1}^{n-1}) , а также номинал для учета заема из расширенного старшего (n -го) разряда $(-e_n)$ (знак результирующего числа меняется).

Третий шаг

Последовательно формируется и записывается результат вычитания целой и дробной частей уменьшаемого и вычитаемого как их алгебраическая разность.

Для вспомогательных целей необходимо использовать таблицы сложения для BIN, OCT, HEX, которые позволяют обеспечить учет заема из старшего разряда числа.

Пример 2.55. Выполнить операцию вычитания двоичных чисел 11011_2 и 01101_2 .

Уменьшаемое и вычитаемое записываются друг под другом, целая часть вычитаемого дополняется незначащим нулем слева для выравнивания длины двух чисел, а их разряды нумеруются: младшему разряду присваивается значение 0 (нулевой разряд), старшему разряду присваивается значение 4 (четвертый разряд).

$$\begin{array}{r} \text{номера разрядов: } 4 \ 3 \ 2 \ 1 \ 0 \\ -1 \ 1 \ 0 \ 1 \ 1_2 \\ \underline{0 \ 1 \ 1 \ 0 \ 1_2} \end{array}$$

Разность уменьшаемого и вычитаемого рассчитывается последовательно:

- нулевой разряд формируется следующим образом: $1_2^0 - 1_2^0 = -0 \ 0_2^0$, где 1_2^0 – номинальное значение цифры в младшем (нулевом) разряде уменьшаемого и вычитаемого; -0 – номинальное значение, которое занимает из следующего старшего (первого) разряда; 0_2^0 – номинальное значение цифры, которая остается в младшем (нулевом) разряде результирующего числа;
- первый разряд формируется следующим образом: $1_2^1 - 0_2^1 + (-0_2^1) = -0 \ 1_2^1$, где 1_2^1 – номинальное значение цифры в первом разряде уменьшаемого; 0_2^1 – номинальное значение цифры в первом разряде вычитаемого; (-0_2^1) – номинальное значение, которое учитывает заем из нулевого разряда на предыдущей итерации; -0 – номинальное значение, которое занимает из второго разряда; 1_2^1 – номинальное значение цифры, которая остается в первом разряде числа;
- второй разряд формируется следующим образом: $0_2^2 - 1_2^2 + (-0_2^2) = -1 \ 1_2^2$, где 0_2^2 – номинальное значение цифры во втором разряде уменьшаемого; 1_2^2 – номинальное значение цифры во втором разряде вычитаемого; (-0_2^2) – номинальное значение, которое учитывает заем из первого разряда; -1 – номинальное значение, которое занимает из третьего разряда числа; 1_2^2 – номинальное значение цифры, которая остается во втором разряде числа;
- третий разряд формируется следующим образом: $1_2^3 - 1_2^3 + (-1_2^3) = -1 \ 1_2^3$, где 1_2^3 – номинальное значение цифры в третьем разряде уменьшаемого;

1_2^3 – номинальное значение цифры в третьем разряде вычитаемого; (-1_2^3) – номинальное значение, которое учитывает заем из второго разряда на предыдущей итерации; -1^4 – номинальное значение, которое занимает из четвертого разряда числа; 1_2^3 – номинальное значение цифры, которая остается в третьем разряде числа;

- четвертый разряд формируется следующим образом: $1_2^4 - 0_2^4 + (-1_2^4) = -0_2^5 0_2^4$, где 1_2^4 – номинальное значение цифры в четвертом разряде уменьшаемого; 0_2^4 – номинальное значение цифры в четвертом разряде вычитаемого; (-1_2^4) – номинальное значение, которое учитывает заем из третьего разряда на предыдущей итерации; -0^5 – номинальное значение, которое занимает из следующего (пятого) разряда числа (отсутствует в уменьшаемом); 0_2^4 – номинальное значение цифры, которая остается в четвертом разряде двоичного числа.

Таким образом:

номера разрядов

$$\begin{array}{r}
 -1-1 \\
 + \mathbf{4\ 3\ 2\ 1\ 0} \\
 + 1\ 1\ 0\ 1\ 1_2 \\
 \underline{0\ 1\ 1\ 0\ 1_2} \\
 0\ 1\ 1\ 1\ 0
 \end{array}$$

В результате получена разность 1110_2 , которую необходимо проверить.

Для этого воспользуемся формулой разложения, которая позволяет получить сумму произведений номинального значения цифры взятой из определенного разряда числа на основание системы счисления возведенное в показатель степени соответствующий номеру разряда откуда была взята цифра в этом числе:

$$11011_2 = 1*2^4 + 1*2^3 + 0*2^2 + 1*2^1 + 1*2^0 = 16 + 8 + 0 + 2 + 1 = 27;$$

$$01101_2 = 0*2^4 + 1*2^3 + 1*2^2 + 0*2^1 + 1*2^0 = 0 + 8 + 4 + 0 + 1 = 13;$$

$$01110_2 = 0*2^4 + 1*2^3 + 1*2^2 + 1*2^1 + 0*2^0 = 0 + 8 + 4 + 2 + 0 = 14.$$

В результате выполнения операции арифметического вычитания уменьшаемого и вычитаемого получена разность $27d - 13d = 14d$ в десятичной системе счисления, что позволяет говорить о корректности полученного двоичного результата в BIN.

Если имеет место отличие между уменьшаемым, вычитаемым и разностью в соответствующих системах счисления, то необходимо выполнить расчет снова.

Пример 2.56. Найти разность шестнадцатеричных чисел $7B_{16}$ и $1C_{16}$.

Уменьшаемое и вычитаемое записываются друг под другом, а их разряды последовательно нумеруются справа налево: младшему разряду присваивается значение 0 (нулевой разряд), старшему разряду присваивается значение 1 (первый разряд).

Процедура выполнения операции арифметического вычитания над двумя шестнадцатеричными числами приводится далее и предполагает использование вспомогательной таблицы вычитания для шестнадцатеричной системы счисления:

Результат вычитания двух чисел рассчитывается в следующей последовательности:

- нулевой разряд формируется следующим образом: $B_{16}^0 - C_{16}^0 = -1 F_{16}^0$, где $B_{16}^0 = 11_{10}$ – номинальное значение нулевого разряда уменьшаемого; $C_{16}^0 = 12_{10}$ – номинальное значение нулевого разряда вычитаемого; -1 – номинальное значение, которое занимает из следующего (первого) разряда; $F_{16}^0 = 15_{10}$ – номинальное значение, которое остается в нулевом разряде числа;
- первый разряд формируется следующим образом: $7_{16}^1 - 1_{16}^1 + (-1_{16}^1) = +0 5_{16}^1$, где 7_{16}^1 – номинальное значение в первом разряде уменьшаемого; 1_{16}^1 – номинальное значение в первом разряде вычитаемого; (-1_{16}^1) – номинальное значение, которое учитывает заем из нулевого разряда на предыдущей итерации; 0 – номинальное значение, которое занимает из второго разряда; 5_{16}^1 – номинальное значение, которое остается в первом разряде числа;

Таким образом, получим:

$$\begin{array}{r}
 \text{номера разрядов} \\
 -1 \\
 \mathbf{10} \\
 - 7 B_{16} \\
 \underline{1 C_{16}} \\
 5 F_{16}
 \end{array}$$

Получен результат $5F_{16}$, который необходимо в дальнейшем проверить.

Для этого воспользуемся формулой разложения, которая позволяет получить сумму произведений номинальных значений цифры взятых из определенного разряда числа на основание системы счисления возведенное в показатель степени соответствующий номеру разряда откуда была взята определенная цифра в этом числе:

$$7B_{16} = 7 \cdot 16^1 + 11 \cdot 16^0 = 112 + 11 = 123_{10};$$

$$1C_{16} = 1 \cdot 16^1 + 12 \cdot 16^0 = 16 + 12 = 28_{10};$$

$$5F_{16} = 5 \cdot 16^1 + 15 \cdot 16^0 = 80 + 15 = 95_{10}.$$

Поскольку достигнуто соответствие $123d - 28d = 95d$, то результат арифметического вычитания в шестнадцатеричной системе счисления является корректным.

Пример 2.57. Найти разность между восьмеричными числами 3543_8 и 643_8 .

Уменьшаемое и вычитаемое записываются друг под другом, целая часть вычитаемого дополняется незначащим нулем слева, а разряды последовательно нумеруются: младшему разряду присваивается значение 0 (нулевой разряд).

$$\begin{array}{r} \text{номиналы заема: } -1 \\ \text{номера разрядов: } \mathbf{3\ 2\ 1\ 0} \\ \quad \quad \quad -\ 3\ 5\ 4\ 3_8 \\ \quad \quad \quad \underline{0\ 6\ 4\ 3_8} \end{array}$$

Результат вычитания двух чисел рассчитывается итеративно по шагам:

- нулевой разряд формируется следующим образом: $3_8^0 - 3_8^0 = -0\ 0_8^0$, где 3_8^0 – номинальное значение нулевого разряда уменьшаемого и вычитаемого; -0 – номинальное значение, которое занимает из следующего разряда (первый разряд); 0_8^0 – номинальное значение, которое остается в нулевом разряде числа;
- первый разряд формируется следующим образом: $4_8^1 - 4_8^1 + (-0_8^1) = -0\ 0_8^1$, где 4_8^1 – номинальное значение первого разряда уменьшаемого и вычитаемого; (-0_8^1) – номинальное значение, которое учитывает заем из нулевого разряда на предыдущей итерации; -0 – номинальное значение, которое занимает из следующего (второго) разряда; 0_8^1 – номинальное значение, которое остается в первом разряде результирующего восьмеричного числа;
- второй разряд формируется следующим образом: $5_8^2 - 6_8^2 + (-0_8^2) = -1\ 7_8^2$, где 5_8^2 – номинальное значение во втором разряде уменьшаемого; 6_8^2 – номинальное значение во втором разряде вычитаемого; (-0_8^2) – номинальное значение, которое учитывает заем из первого разряда на предыдущей итерации; -1 – номинальное значение, которое занимает из третьего разряда числа; 7_8^2 – номинальное значение, которое остается во втором разряде числа;
- третий разряд формируется следующим образом: $3_8^3 - 0_8^3 + (-1_8^3) = -0\ 4_8^3$, где 3_8^3 – номинальное значение в третьем разряде уменьшаемого; 0_8^3 – номинальное значение в третьем разряде вычитаемого; (-1_8^3) – номинальное значение, которое учитывает заем из второго разряда на предыдущей итерации; -0 – номинальное значение, которое занимает из четвертого разряда числа; 4_8^3 – номинальное значение, которое остается в старшем (третьем) разряде числа.

Пример 2.58. Выполнить арифметическое вычитание чисел 97_{16} и $7B_{16}$.

Уменьшаемое и вычитаемое записываются друг под другом, их разряды нумеруются справа налево: младшему разряду присваивается значение 0 (нулевой разряд), старшему разряду присваивается значение 1 (первый разряд).

$$\begin{array}{r} \text{номиналы заема: } -1 \\ \text{номера разрядов: } \mathbf{1\ 0} \\ \phantom{\text{номера разрядов: }} -9\ 7_{16} \\ \phantom{\text{номера разрядов: }} \underline{7\ B_{16}} \end{array}$$

Результат вычитания двух чисел рассчитывается последовательно:

- нулевой разряд формируется следующим образом: $7_{16}^0 - B_{16}^0 = -1^1 B_{16}^0$, где
 7_{16}^0 – номинальное значение цифры в нулевом разряде уменьшаемого;
 B_{16}^0 – номинальное значение цифры в нулевом разряде вычитаемого;
 -1^1 – номинальное значение, которое занимает из следующего (1-го) разряда;
 B_{16}^0 – номинальное значение, которое остается в нулевом разряде числа;
- первый разряд формируется следующим образом: $9_{16}^1 - 7_{16}^1 - (1_{16}^1) = -0^2 1_{16}^1$, где
 9_{16}^1 – номинальное значение цифры в нулевом разряде уменьшаемого и вычитаемого;
 7_{16}^1 – номинальное значение, которое учитывает заем из нулевого разряда;
 -0^2 – номинальное значение, которое занимает из второго разряда;
 (1_{16}^1) – номинальное значение цифры, которая остается в 1-м разряде числа.

Таким образом:

$$\begin{array}{r} \text{номиналы заема: } -1 \\ \text{номера разрядов: } \mathbf{1\ 0} \\ \phantom{\text{номера разрядов: }} -9\ 7_{16} \\ \phantom{\text{номера разрядов: }} \underline{7\ B_{16}} \\ \phantom{\text{номера разрядов: }} 1\ C_{16} \end{array}$$

Получен результат $1C_{16}$, который необходимо проверить.

Проверка осуществления операции арифметического вычитания двух чисел:

$$97_{16} = 9 \cdot 16^1 + 7 \cdot 16^0 = 144 + 7 = 151_{10};$$

$$7B_{16} = 7 \cdot 16^1 + 11 \cdot 16^0 = 112 + 11 = 123_{10};$$

$$1C_{16} = 1 \cdot 16^1 + 12 \cdot 16^0 = 16 + 12 = 28_{10}.$$

Поскольку было достигнуто соответствие в десятичной системе счисления $151d - 123d = 28d$, то полученный результат операции арифметического вычитания двух чисел представленных в восьмеричной системе счисления является корректным.

Первый шаг

Исходные дробные числа включают явную целую и дробную части, записываются друг под другом, выступают в роли первого $(a_{n-1} \dots a_i \dots a_1 a_0, a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m})$ и второго множителей $(b_{n-1} \dots b_i \dots b_1 b_0, b_{-1} \dots b_{-j} \dots b_{-(m-1)} b_{-m})$ при операции арифметического умножения:

- переходят от операции умножения к операции арифметического сложения: при этом рассчитывают вспомогательные элементы, которые подлежат сложению при помощи перемножения номиналов цифр первого и второго множителей;
- в результате формируется последовательность цифр результирующего числа $(c_{n+6} c_{n+5} c_{n+4} c_{n+3} c_{n+2} c_{n+1} c_n c_{n-1} c_{n-2} \dots c_i c_{i+1} \dots c_1 c_0 c_{-1} c_{-(j-1)} c_{-j} c_{-(j+1)} c_{-(m-1)} c_{-m})$ — произведение первого и второго множителей как результат суммирования номинальных значений рассчитанных вспомогательных элементов (чисел).

Число (произведение) в определенной системе счисления состоит из последовательности цифр, которые получены в результате перемножения двух множителей:

- $c_{n-1} c_{n-2} \dots c_i c_{i-1} \dots c_1 c_0 c_{-1} c_{-(j-1)} c_{-j} c_{-(j+1)} c_{-(m-1)} c_{-m}$ — основной набор цифр числа;
- $c_{n+6} c_{n+5} c_{n+4} c_{n+3} c_{n+2} c_{n+1} c_n$ — расширенный набор цифр числа, которые дополнительно обеспечили увеличение длины числа, которое получено в результате перемножения первого и второго множителей и суммирования.

Второй шаг

Реализуется перемножение последовательности цифр первого и второго множителей:

- умножается цифра в младшем $(-m)$ разряде второго числа на все цифры в разрядах первого числа справа налево (от $-m$ до $n-1$), формируются вспомогательные элементы первой строки для последующей процедуры суммирования;
- умножается следующая цифра в $(-m-1)$ -м разряде второго числа на все цифры в разрядах первого числа справа налево (от $-m$ до $n-1$), осуществляется сдвиг элементов во второй строке на 1 разряд влево и записываются номинальные значения вспомогательных элементов для последующего суммирования;
- умножается цифра в $-j$ -м разряде второго числа на все цифры в разрядах первого числа справа налево (от $-m$ до $n-1$), осуществляется сдвиг элементов в третьей строке еще на 1 разряд влево и записываются номинальные значения вспомогательных элементов для последующего суммирования;
- последовательно умножаются цифры в -1 -м, затем в 0 -м, а затем в 1 -м разрядах второго числа на все цифры в разрядах первого числа справа налево (от $-m$ до $n-1$), осуществляется сдвиг элементов в сформированной строке еще на 1, 2, 3 разряд влево и записываются номинальные значения вспомогательных элементов для реализации последующего суммирования;
- умножается цифра в i -м разряде второго числа на все цифры в разрядах первого числа справа налево (от $-m$ до $n-1$), осуществляется сдвиг элементов

в i -й строке еще на 1 разряд влево и записываются номинальные значения вспомогательных элементов для реализации последующего суммирования;

- умножается цифра в старшем $((n-1)$ -м) разряде второго числа на все цифры в разрядах первого числа справа налево (от $-m$ до $n-1$), осуществляется сдвиг элементов в последней $(n-1)$ строке еще на 1 разряд влево и записываются номинальные значения вспомогательных элементов для обеспечения процедуры последующего суммирования и расчета результата;
- полученные вспомогательные элементы подлежат дальнейшему суммированию.

Третий шаг

Осуществляется поразрядное сложение цифр от младшего разряда $(-m$ -й) к старшему $((n-1)$ -й), которые расположены в полученных строках, а если длина последовательности цифр в целой или дробной частях различна в двух числах, то осуществляется дополнение незначащими нулями слева для целого числа и справа для правильной дроби с целью выравнивания длины всех слагаемых результата:

- осуществляется поразрядное сложение двух дробных частей исходных чисел:
 - номинальное значение цифры в младшем $(-m-m)$ разряде (c_{-m}^{-m}) равно $b_{-m}^{-m} a_{-m}^{-m}$, формируется номинал для учета переноса в следующий разряд $(+d_{-(m-1)}^{-(m-1)})$;
 - складываются номиналы цифр следующего $(-(m-1)$ -го) разряда первого $(b_{-m}^{-m} a_{-(m-1)}^{-(m-1)})$ и второго слагаемых $(b_{-(m-1)}^{-(m-1)} a_{-m}^{-m})$, а также номинал для учета переноса $(+d_{-(m-1)}^{-(m-1)})$ на предыдущей итерации, а затем формируются: новое значение в младшем $(-(m-1)$ -м) разряде результирующего числа $(c_{-(m-1)}^{-(m-1)})$ и номинал для переноса в следующий $(-(m-2)$ -й) разряд $(+d_{-(m-2)}^{-(m-2)})$;
 - складываются номинальные значения цифр $-(j+1)$ -го разряда первого $(b_{-m}^{-m} a_{-(j+1)}^{-(j+1)})$, второго $(b_{-(m-1)}^{-(m-1)} a_{-(m-1)}^{-(m-1)})$ и третьего слагаемых $(b_{-j}^{-j} a_{-m}^{-m})$, а также номинал для учета переноса $(+d_{-(j+1)}^{-(j+1)})$ на предыдущей итерации, формируются: новое значение цифры в младшем $(-(j+1)$ -м) разряде результирующего числа $(c_{-(j+1)}^{-(j+1)})$ и номинал для переноса в следующий $(-j$ -й) разряд $(+d_{-j}^{-j})$;
 - складываются номиналы цифр $-j$ -го разряда первого $(b_{-m}^{-m} a_{-j}^{-j})$, второго $(b_{-(m-1)}^{-(m-1)} a_{-(j+1)}^{-(j+1)})$, третьего $(b_{-j}^{-j} a_{-(m-1)}^{-(m-1)})$ и четвертого слагаемого $(b_{-1}^{-1} a_{-m}^{-m})$, а также учитывается номинал переноса $(+d_{-j}^{-j})$ на предыдущей итерации, формируются: новое значение цифры в младшем $(-j$ -м) разряде результирующего числа (c_{-j}^{-j}) и номинал для учета переноса в следующий $(-(j-1)$ -й) разряд $(+d_{-(j-1)}^{-(j-1)})$;
 - складываются номиналы цифр $-(j-1)$ -го разряда первого $(b_{-m}^{-m} a_{-(j-1)}^{-(j-1)})$, второго

- $(b_{-(m-1)}^{-(m-1)} a_{-j}^{-j})$, третьего $(b_{-j}^{-j} a_{-(j+1)}^{-(j+1)})$, четвертого $(b_{-1}^{-1} a_{-(m-1)}^{-(m-1)})$ и пятого слагаемых $(b_0^0 a_{-m}^{-m})$, а также учитывается номинал переноса $(+d_{-(j-1)})$ на предыдущей итерации, формируются: новое значение в младшем $(-j-1)$ -м разряде результирующего числа $(c_{-(j-1)}^{-(j-1)})$ и номинал для учета переноса в $(-j-2)$ -й разряд $(+d_{-(j-2)})$;
- складываются номиналы цифр в -1 -м разряде первого $(b_{-m}^{-m} a_{-1}^{-1})$, второго $(b_{-(m-1)}^{-(m-1)} a_{-(j-1)}^{-(j-1)})$, третьего $(b_{-j}^{-j} a_{-j}^{-j})$, четвертого $(b_{-1}^{-1} a_{-(j+1)}^{-(j+1)})$, пятого $(b_0^0 a_{-(m-1)}^{-(m-1)})$ и шестого слагаемых $(b_1^1 a_{-m}^{-m})$, а также учитывается номинал переноса $(+d_{-1})$ на предыдущей итерации, формируются: новое значение цифры в младшем разряде результирующего числа (c_{-1}^{-1}) и номинал для учета переноса $(+d_0)$ в следующий старший разряд (нулевой разряд целой части числа);
 - продолжается поразрядное сложение двух целых частей исходного числа:
 - складываются номиналы цифр нулевого разряда первого $(b_{-m}^{-m} a_0^0)$, второго $(b_{-(m-1)}^{-(m-1)} a_{-1}^{-1})$, третьего $(b_{-j}^{-j} a_{-(j-1)}^{-(j-1)})$, четвертого $(b_{-1}^{-1} a_{-j}^{-j})$, пятого $(b_0^0 a_{-(j+1)}^{-(j+1)})$, шестого $(b_1^1 a_{-(m-1)}^{-(m-1)})$ и седьмого слагаемых $(b_i^i a_{-m}^{-m})$, а также учитывается номинал переноса $(+d_0)$ на предыдущей итерации, формируются: новое значение цифры в младшем (нулевом) разряде результирующего числа (c_0^0) и номинал для переноса в следующий (первый) разряд $(+d_1)$;
 - складываются номиналы цифр первого разряда первого $(b_{-m}^{-m} a_1^1)$, второго $(b_{-(m-1)}^{-(m-1)} a_0^0)$, третьего $(b_{-j}^{-j} a_{-1}^{-1})$, четвертого $(b_{-1}^{-1} a_{-(j-1)}^{-(j-1)})$, пятого $(b_0^0 a_{-j}^{-j})$, шестого $(b_1^1 a_{-(j+1)}^{-(j+1)})$, седьмого $(b_i^i a_{-(m-1)}^{-(m-1)})$ и восьмого слагаемых $(b_{n-1}^{n-1} a_{-m}^{-m})$, а также учитывается номинал переноса $(+d_1)$ на предыдущей итерации, формируются: новое значение цифры в младшем (первом) разряде результирующего числа (c_1^1) и номинал для переноса в следующий (второй) разряд $(+d_2)$;
 - складываются номиналы цифр $(i-1)$ -го разряда первого $(b_{-m}^{-m} a_{i-1}^{i-1})$, второго $(b_{-(m-1)}^{-(m-1)} a_1^1)$, третьего $(b_{-j}^{-j} a_0^0)$, четвертого $(b_{-1}^{-1} a_{-1}^{-1})$, пятого $(b_0^0 a_{-(j-1)}^{-(j-1)})$, шестого $(b_1^1 a_{-j}^{-j})$, седьмого $(b_i^i a_{-(j+1)}^{-(j+1)})$ и восьмого слагаемых $(b_{n-1}^{n-1} a_{-(m-1)}^{-(m-1)})$, а также учитывается номинал переноса $(+d_{i-1})$ на предыдущей итерации, формируются: новое значение цифры в младшем $((i-1)$ -м) разряде результирующего числа (c_{i-1}^{i-1}) и номинал для переноса в следующий (i) -й разряд $(+d_i)$;

- складываются номиналы цифр i -го разряда первого ($b_{-m}^{\quad i} a_i$), второго ($b_{-(m-1)}^{\quad i-1} a_{i-1}$), третьего ($b_{-j}^{\quad -j \quad 1} a_{i-1}$), четвертого ($b_{-1}^{\quad -1 \quad 0} a_0$), пятого ($b_0^{\quad 0 \quad -1} a_{-1}$), шестого ($b_1^{\quad 1 \quad -(j-1)} a_{-(j-1)}$), седьмого ($b_i^{\quad i \quad -j} a_{-j}$) и восьмого слагаемых ($b_{n-1}^{\quad n-1 \quad -(j+1)} a_{-(j+1)}$), а также учитывается номинал переноса ($+d_i$) на предыдущей итерации, формируются: новое значение цифры в младшем (i -м) разряде результирующего числа ($c_i^{\quad i}$) и номинал для переноса в следующий ($(i+1)$ -й) разряд ($+d_{i+1}$);
- складываются номиналы цифр $(i+1)$ -го разряда первого ($b_{-m}^{\quad -m \quad i+1} a_{i+1}$), второго ($b_{-(m-1)}^{\quad -(m-1) \quad i} a_i$), третьего ($b_{-j}^{\quad -j \quad i-1} a_{i-1}$), четвертого ($b_{-1}^{\quad -1 \quad 1} a_1$), пятого ($b_0^{\quad 0 \quad 0} a_0$), шестого ($b_1^{\quad 1 \quad -1} a_{-1}$), седьмого ($b_i^{\quad i \quad -(j-1)} a_{-(j-1)}$) и восьмого слагаемых ($b_{n-1}^{\quad n-1 \quad -j} a_{-j}$), а также учитывается номинал переноса ($+d_{i+1}$) на предыдущей итерации, формируются: новое значение цифры в младшем ($(i+1)$ -м) разряде результирующего числа ($c_{i+1}^{\quad i+1}$) и номинал для переноса в следующий ($(n-1)$ -й) разряд ($+d_{n-1}$);
- складываются номиналы цифр старшего ($n-1$ -го) разряда первого ($b_{-m}^{\quad -m \quad n-1} a_{n-1}$), второго ($b_{-(m-1)}^{\quad -(m-1) \quad i+1} a_{i+1}$), третьего ($b_{-j}^{\quad -j \quad i} a_i$), четвертого ($b_{-1}^{\quad -1 \quad i-1} a_{i-1}$), пятого ($b_0^{\quad 0 \quad 1} a_1$), шестого ($b_1^{\quad 1 \quad 0} a_0$), седьмого ($b_i^{\quad i \quad -1} a_{-1}$) и восьмого слагаемых ($b_{n-1}^{\quad n-1 \quad -(j-1)} a_{-(j-1)}$), а также учитывается номинал переноса ($+d_{n-1}$) на предыдущей итерации, формируются: новое значение цифры в младшем ($(n-1)$ -м) разряде результирующего числа ($c_{n-1}^{\quad n-1}$), а также номинал для обеспечения учета переноса в следующий (n -й) разряд ($+d_n$) при выполнении операции;
- складываются номиналы цифр старшего (n -го) разряда первого (ноль), второго ($b_{-(m-1)}^{\quad -(m-1) \quad n-1} a_{n-1}$), третьего ($b_{-j}^{\quad -j \quad i+1} a_{i+1}$), четвертого ($b_{-1}^{\quad -1 \quad i} a_i$), пятого ($b_0^{\quad 0 \quad i-1} a_{i-1}$), шестого ($b_1^{\quad 1 \quad 1} a_1$), седьмого ($b_i^{\quad i \quad 0} a_0$) и восьмого слагаемых ($b_{n-1}^{\quad n-1 \quad -1} a_{-1}$), а также учитывается номинал переноса ($+d_n$) на предыдущей итерации, формируются: новое значение в младшем (n -м) разряде результирующего числа ($c_n^{\quad n}$), а также номинал для учета переноса в следующий ($(n+1)$ -й) разряд ($+d_{n+1}$);
- складываются номиналы цифр старшего ($n+1$ -го) разряда первого (ноль), второго (ноль), третьего ($b_{-j}^{\quad -j \quad n-1} a_{n-1}$), четвертого ($b_{-1}^{\quad -1 \quad i+1} a_{i+1}$), пятого ($b_0^{\quad 0 \quad i} a_i$), шестого ($b_1^{\quad 1 \quad i-1} a_{i-1}$), седьмого ($b_i^{\quad i \quad 1} a_1$) и восьмого слагаемых ($b_{n-1}^{\quad n-1 \quad 0} a_0$), а также учитывается номинал переноса ($+d_{n+1}$) на предыдущей итерации, формируются: новое

- значение в младшем ((n+1)-м) разряде результирующего числа (c_{n+1}^{n+1}), а также номинал для учета переноса в следующий старший ((n+2)-й) разряд ($+d_{n+2}$);
- складываются номиналы цифр старшего (n+2-го) разряда первого (ноль), второго (ноль), третьего (ноль), четвертого ($b_{-1}^{n-1} a_{n-1}$), пятого ($b_0^{i+1} a_{i+1}$), шестого ($b_1^{i+1} a_{i+1}$), седьмого ($b_i^{i-1} a_{i-1}$) и восьмого слагаемых ($b_{n-1}^{n-1} a_1$), а также учитываются номинал переноса ($+d_{n+2}$) на предыдущей итерации, формируются: новое значение цифры в младшем ((n+2)-м) разряде результирующего числа (c_{n+2}^{n+2}), а также номинал для учета переноса в следующий ((n+3)-й) разряд ($+d_{n+3}$);
 - складываются номиналы цифр старшего (n+3-го) разряда первого (ноль), второго (ноль), третьего (ноль), четвертого (ноль), пятого ($b_0^{n-1} a_{n-1}$), шестого ($b_1^{i+1} a_{i+1}$), седьмого ($b_i^{i+1} a_{i+1}$) и восьмого слагаемых ($b_{n-1}^{n-1} a_{i-1}$), а также учитывается номинал переноса ($+d_{n+3}$) на предыдущей итерации, формируются: новое значение цифры в младшем разряде результирующего числа (c_{n+3}^{n+3}), а также номинал для учета переноса в следующий ((n+4)-й) разряд ($+d_{n+4}$);
 - складываются номиналы старшего (n+4-го) разряда первого (ноль), второго (ноль), третьего (ноль), четвертого (ноль), пятого (ноль), шестого ($b_1^{n-1} a_{n-1}$), седьмого ($b_i^{i+1} a_{i+1}$) и восьмого слагаемых ($b_{n-1}^{n-1} a_i$), а также учитывается номинал переноса ($+d_{n+4}$) на предыдущей итерации, формируются: новое значение в младшем ((n+4)-м) разряде результирующего числа (c_{n+4}^{n+4}), а также номинал для учета переноса в следующий ((n+5)-й) разряд ($+d_{n+5}$);
 - складываются номиналы старшего (n+5-го) разряда первого (ноль), второго (ноль), третьего (ноль), четвертого (ноль), пятого (ноль), шестого (ноль), седьмого ($b_i^{n-1} a_{n-1}$) и восьмого слагаемых ($b_{n-1}^{n-1} a_{i+1}$), а также учитывается номинал переноса ($+d_{n+5}$) на предыдущей итерации, формируются: новое значение цифры в младшем разряде результирующего числа (c_{n+5}^{n+5}), а также номинал для учета переноса в следующий ((n+6)-й) разряд ($+d_{n+6}$);
 - номинальное значение старшего (n+6-го) разряда (c_{n+6}^{n+6}) равно $b_{n-1}^{n-1} a_{n-1}$, при этом учитывается номинал переноса ($+d_{n+6}^{n+6}$) на предыдущей итерации, формируется номинал для переноса в следующий разряд ($+d_{n+7}^{n+7}$) с целью дополнения комбинации цифр слева относительно старшего разряда.

Необходимо выполнить проверку корректности операции сложения всех вспомогательных номинальных значений, которые были рассчитаны согласно данному алгоритму, а также позволили в дальнейшем получить результат – цифры итогового числа в определенной системе счисления с заданным основанием (P):

$$c_{n+6}, c_{n+5}, c_{n+4}, c_{n+3}, c_{n+2}, c_{n+1}, c_n, c_{n-1}, c_{n-2}, c_i, c_{i+1}, c_1, c_0, c_{-1}, c_{-(j-1)}, c_{-j}, c_{-(j+1)}, c_{-(m-1)}, c_{-m}.$$

Вспомогательные (промежуточные) номинальные значения и полученные номинальные значения цифр результирующего числа необходимо точно записать относительно введенной разрядной сетки для исключения возможных ошибок на этапах расчета и сложения, которые обеспечивают формирование цифр итогового числа.

Четвертый шаг

Формируется и записывается результат сложения целой и дробной частей числа.

Представленный алгоритм является рекуррентным и позволяет осуществить перемножение двух дробных чисел методом «в столбик», а каждый из множителей при этом допускает отсутствие целой или дробной части (равняется нулю), требуется:

- записать исходные числа (первый и второй множитель) для последующего умножения относительно введенной разрядной сетки, в которой каждая цифра обоих чисел расположена в определенном разряде с заданным номером;
- получить набор номинальных значений вспомогательных чисел, которые позволяют осуществить переход от операции умножения к операции сложения;
- корректно записать полученные номинальные значения вспомогательных чисел относительно используемой разрядной сетки, а также представить их в удобной форме для последующей интерпретации и дальнейшей работы;
- реализовать переход от операции умножения двух множителей к операции поразрядного сложения всех полученных номинальных значений вспомогательных чисел, которые сосредоточены в нескольких сформированных строках;
- обеспечить поразрядное сложение вспомогательных (промежуточных) чисел, которые расположены в нескольких строках, а также сформировать ряд цифр результирующего числа в системе счисления с заданным основанием (P);
- осуществить быструю проверку полученного числа посредством использования формулы разложения, поскольку все цифры исходных чисел (первого и второго множителей), а также результат записаны относительно введенной разрядной сетки, что позволяет воспользоваться рекуррентной формулой.

Для осуществления арифметической операции умножения чисел в двоичной (BIN), восьмеричной (OCT) и шестнадцатеричной (HEX) системах счисления необходимо использовать таблицы умножения для формирования номинала в младшем разряде и отслеживания переноса в старший разряд, которые представлены в табл. 2.6, 2.7, 2.8.

Для упрощения аналитически-численных расчетов над числами, которые представлены в восьмеричной (OCT) и шестнадцатеричной (HEX) системах счисления допустимо использование двоичной системы счисления (BIN) как вспомогательной.

Таблица 2.6

Таблица умножения чисел для двоичной системы счисления

Знак операции «*»		Значение первого множителя	
		0	1
Значение второго множителя	0	0	1
	1	1	1

Таблица 2.7

Таблица умножения чисел для восьмеричной системы счисления

Знак операции «*»		Значение первого множителя								
		0	1	2	3	4	5	6	7	
Значение второго множителя	0	0	0	0	0	0	0	0	0	0
	1	0	1	2	3	4	5	6	7	
	2	0	2	4	6	+10	+12	+14	+16	
	3	0	3	6	+11	+14	+17	+22	+25	
	4	0	4	+10	+14	+20	+24	+30	+34	
	5	0	5	+12	+17	+24	+31	+36	+44	
	6	0	6	+14	+22	+30	+36	+44	+52	
	7	0	7	+16	+25	+34	+43	+52	+61	

Таблица 2.8

Таблица умножения чисел для шестнадцатеричной системы счисления

Знак операции «*»		Значение первого множителя															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Значение второго множителя	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	2	0	2	4	6	8	A	C	E	+10	+12	+14	+16	+18	+1A	+1C	+1E
	3	0	3	6	9	C	F	+12	+15	+18	+1B	+1E	+21	+24	+27	+2A	+2D
	4	0	4	8	C	+10	+14	+18	+1C	+20	+24	+28	+2C	+30	+34	+38	+3C
	5	0	5	A	F	14	19	1E	23	28	2D	32	+37	+3C	+41	+46	+4B
	6	0	6	C	12	18	1E	24	2A	30	36	+3C	+42	+48	+4E	+54	+5A
	7	0	7	E	15	16	23	2A	31	38	+3F	+46	+4D	+54	+5B	+62	+69
	8	0	8	10	18	20	28	30	38	+40	+48	+50	+58	+60	+68	+70	+78
	9	0	9	12	1B	24	2D	36	+3F	+48	+51	+5A	+63	+6C	+75	+7E	+87
	A	0	A	14	1E	28	32	+3C	+46	+50	+5A	+64	+6E	+78	+82	+8C	+96
	B	0	B	16	21	2C	+37	+42	+4D	+58	+63	+6E	+79	+84	+8F	+9A	+A5
	C	0	C	18	24	30	3C	48	54	60	6C	78	84	90	9C	A8	B4
	D	0	D	1A	27	34	41	4E	5B	68	75	82	8F	9C	A9	B6	C3
	E	0	E	1C	2A	38	46	54	62	70	7E	8C	9A	A8	B6	C4	D2
	F	0	F	1E	2D	3C	4B	5A	69	78	87	96	A5	B4	C3	D2	E1

Пример 2.59. Выполнить умножение числа $101,101_2$ на число $101,000_2$.

Первый и второй множители записываются друг под другом, дробная часть второго числа дополняется тремя незначащими нулями справа для выравнивания длины двух чисел, а затем разряды нумеруются: младшему разряду присваивается значение -3 (-3 разряд), старшему разряду присваивается значение 2 (2 разряд).

$$\begin{array}{r} 2\ 1\ 0\ -1\ -2\ -3 \\ \times 1\ 0\ 1,\ 1\ 0\ 1_2 \\ \hline 1\ 0\ 1,\ 0\ 0\ 0_2 \end{array}$$

Результат умножения двух чисел формируется в рамках нескольких этапов:

- формирование набора вспомогательных (промежуточных) чисел для обеспечения возможности перехода от операции алгебраического умножения к операции сложения с целью упрощения используемых вычислительных процедур;
 - умножается цифра ноль в младшем (-3-м) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от -3 до 2), формируется ряд вспомогательных элементов в первой строке для последующей реализации процедуры арифметического суммирования (000,000);
 - умножается цифра ноль в следующем (-2-м) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от -3 до 2), осуществляется сдвиг всех элементов во второй строке на 1 разряд влево и записываются номинальные значения вспомогательных элементов для реализации процедуры последующего арифметического суммирования (000,000);
 - умножается цифра ноль в следующем (-1-м) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от -3 до 2), осуществляется сдвиг всех элементов в сформированной строке еще на 1 разряд влево и записываются номинальные значения вспомогательных элементов для реализации последующего арифметического суммирования (000,000);
 - умножается цифра 1 в следующем (0-м) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от -3 до 2), осуществляется сдвиг всех элементов в сформированной строке еще на 1 разряд влево и записываются номинальные значения вспомогательных элементов для реализации процедуры последующего суммирования (101,101);
 - умножается цифра ноль в следующем (1-м) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от -3 до 2), осуществляется сдвиг всех элементов в сформированной строке еще на 1 разряд влево и записываются номинальные значения вспомогательных элементов для реализации последующего арифметического суммирования (000,000);
 - умножается цифра 1 в следующем (2-м) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от -3 до 2), осуществляется сдвиг всех элементов в сформированной строке еще на 1 разряд влево и записываются номинальные значения вспомогательных элементов для реализации последующего арифметического суммирования (101,101).

При выполнении операции арифметического умножения целой и дробной частей двух множителей результирующее значение формируется последовательно:

номера разрядов:

$$\begin{array}{r}
 \mathbf{2\ 1\ 0\ -1\ -2\ -3} \\
 \times \quad 1\ 0\ 1, 0\ 0\ 0_2 \\
 \hline
 \mathbf{+4\ 1\ 0\ 1, 1\ 0\ 1} \\
 +4\ 0\ 0\ 0, 0\ 0\ 0 \\
 \Sigma \quad +4\ 1\ 0\ 1, 1\ 0\ 1 \\
 \quad \quad +4\ 1\ 0\ 1, 1\ 0\ 1 \\
 \quad \quad \quad +3\ 0\ 0\ 0, 0\ 0\ 0 \\
 \quad \quad \quad \quad +3\ 1\ 0\ 1, 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0 \\
 \hline
 1\ 1\ 0\ 1\ 1, 1\ 1\ 1\ 0\ 1\ 0\ 0\ 1 = 27,91015625_{10} \\
 \mathbf{4\ 3\ 2\ 1\ 0\ -1\ -2\ -3\ -4\ -5\ -6\ -7\ -8}
 \end{array}$$

Суммирование вспомогательных элементов реализуется последовательно:

- складываются номиналы цифр младшего -8-го разряда первого слагаемого (1) и формируются: новое значение в младшем разряде результирующего числа (1), а также номинал для учета переноса (+0) в следующий старший (-7-й) разряд;
- складываются номиналы цифр -7-го разряда первого (0) и второго (0) слагаемых с учетом номинала переноса (+0) на предыдущей итерации, формируются: новое значение в младшем разряде результирующего числа (0), а также номинал для учета переноса (+0) в следующий старший (-6-й) разряд;
- складываются номиналы цифр -6-го разряда первого (1), второго (0) и третьего (1) слагаемых с учетом номинала переноса (+0) на предыдущей итерации, затем формируются: новое значение цифры в младшем разряде результирующего числа (0), а также номинал для учета переноса (+1) в следующий (-5-й) разряд;
- складываются номиналы цифр -5-го разряда первого (1), второго (0), третьего (0), четвертого (1) слагаемых с учетом номинала переноса (+1) на предыдущей итерации, формируются: новое значение в младшем разряде результирующего числа (1), а также номинал для обеспечения учета переноса (+1) в следующий (-4-й) разряд;
- складываются номиналы цифр -4-го разряда первого (0), второго (0), третьего (1), четвертого (0) и пятого (0) слагаемых с учетом номинала переноса (+1) на предыдущей итерации, затем формируются: новое значение цифры в младшем (-4-м) разряде результирующего числа (0), а также номинал для обеспечения учета переноса (+1) в следующий старший (-3-й) разряд;
- складываются номиналы цифр -3-го разряда первого (1), второго (0), третьего (1), четвертого (1), пятого (0) и шестого (1) слагаемых с учетом номинала переноса (+1) на предыдущей итерации, затем формируются: новое значение цифры в младшем (-3-м) разряде результирующего числа (1), а также номинал для обеспечения учета переноса (+4) в следующий старший (-2-й) разряд;

- складываются номиналы цифр -2-го разряда первого (0), второго (0), третьего (1), четвертого (0) и пятого (0) слагаемых с учетом номинала переноса (+4) полученного на предыдущей итерации, затем формируются: новое значение цифры в младшем (-2-м) разряде результирующего числа (1), а также номинал для обеспечения учета переноса (+4) в следующий старший (-1-й) разряд;
- складываются номиналы цифр -1-го разряда первого (1), второго (0), третьего (0) и четвертого (1) слагаемых с учетом номинала переноса (+4) на предыдущей итерации, формируются: новое значение в младшем разряде результирующего числа (1), а также номинал для учета переноса (+4) в старший (0-й) разряд;
- складываются номиналы цифр нулевого разряда первого (1), второго (0) и третьего (1) слагаемых с учетом номинала переноса (+4) на предыдущей итерации, формируются: новое значение в младшем разряде результирующего числа (1), а также номинал для учета переноса (+4) в старший (1-й) разряд;
- складываются номиналы цифр 1-го разряда первого (0) и второго (0) слагаемых с учетом номинала переноса (+4) на предыдущей итерации, затем формируются: новое значение цифры в младшем разряде результирующего числа (1), а также номинал для обеспечения учета переноса (+3) в старший (2-й) разряд;
- складываются номиналы цифр 2-го разряда первого (1) слагаемого с учетом номинала переноса (+3) на предыдущей итерации, затем формируются: новое значение цифры в младшем разряде результирующего числа (0), а также номинал для поддержки учета переноса (+3) в старший (3-й) разряд;
- полученный номинал переноса +3 на предыдущей итерации записывается в 3-м и 4-м разрядах результирующего числа, а затем формируются: новое значение цифры в третьем разряде результирующего числа (1) и новое значение цифры расположенной в четвертом разряде результирующего числа (1).

В результате операции арифметического сложения вспомогательных элементов формируется результат в двоичной (BIN) системе счисления: $(11011,11101001)_2$.

Возникает необходимость выполнения проверки полученного результата при помощи преобразования в DEC систему счисления посредством формулы разложения:

$$\begin{aligned}
 & \mathbf{4\ 3\ 2\ 1\ 0\ -1\ -2\ -3\ -4\ -5\ -6\ -7\ -8} \text{ номера разрядов} \\
 & (1\ 1\ 0\ 1\ 1, 1\ 1\ 1\ 0\ 1\ 0\ 0\ 1)_2 = \\
 & = 1*2^4 + 1*2^3 + 0*2^2 + 1*2^1 + 1*2^0 + 1*2^{-1} + 1*2^{-2} + 1*2^{-3} + 0*2^{-4} + 1*2^{-5} + 0*2^{-6} + 0*2^{-7} + 1*2^{-8} = \\
 & = 16 + 8 + 0 + 2 + 1 + 0,5 + 0,25 + 0,125 + 0 + 0,03125 + 0 + 0 + 0,00390625 = 27,91015625
 \end{aligned}$$

$\mathbf{2\ 1\ 0\ -1\ -2\ -3}$ номера разрядов

$$(1\ 0\ 1, 0\ 0\ 0)_2 = 1*2^2 + 0*2^1 + 1*2^0 + 0*2^{-1} + 0*2^{-2} + 0*2^{-3} = 4 + 0 + 1 + 0 + 0 + 0 = 5$$

$\mathbf{2\ 1\ 0\ -1\ -2\ -3}$ номера разрядов

$$(1\ 0\ 1, 1\ 0\ 1)_2 = 1*2^2 + 0*2^1 + 1*2^0 + 1*2^{-1} + 0*2^{-2} + 1*2^{-3} = 4 + 0 + 1 + 0,5 + 0 + 0,125 = 5,625$$

В результате умножения первого и второго множителей в DEC $5*5,625=28,125$.

Незначительное отличие результата обусловлено особенностями процедуры преобразования и представлением результирующего числа в BIN системе счисления.

Ответ: $11011,11101001_2$.

Пример 2.60. Выполнить умножение восьмеричных чисел 123_8 и 12_8 .

Для реализации процедуры арифметического умножения первый и второй множители записываются друг под другом, целая часть второго множителя дополняется незначащим нулем слева, разряды чисел нумеруются: младшему разряду присваивается значение 0 (нулевой разряд), среднему разряду присваивается значение 1 (первый разряд), старшему разряду присваивается значение 2 (второй разряд).

номера разрядов:

$$\begin{array}{r} \mathbf{2\ 1\ 0} \\ \times 1\ 2\ 3_8 \\ \hline \times 0\ 1\ 2_8 \end{array}$$

Результат умножения двух чисел формируется в рамках нескольких этапов:

- формирование последовательности вспомогательных (промежуточных) чисел для обеспечения возможности перехода от операции умножения к операции арифметического сложения с целью упрощения используемых вычислительных процедур;
 - умножается цифра 2 в младшем (нулевом) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от 0 до 2) и записываются номинальные значения вспомогательных элементов для последующей реализации процедуры арифметического суммирования (246);
 - умножается цифра 1 в следующем (первом) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от 0 до 2), осуществляется сдвиг элементов в сформированной строке на 1 разряд влево и записываются номинальные значения вспомогательных элементов для обеспечения возможности реализации последующего суммирования (123);
 - умножается цифра 0 в старшем (втором) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от 0 до 2), осуществляется сдвиг элементов в сформированной строке еще на 1 разряд влево и записываются номинальные значения вспомогательных элементов для последующей реализации процедуры суммирования (000).

При выполнении операции арифметического умножения целой и дробной частей двух множителей результирующее значение формируется последовательно:

номера разрядов:

$$\begin{array}{r} \mathbf{2\ 1\ 0} \\ \times 1\ 2\ 3_8 \\ \hline \times 0\ 1\ 2_8 \\ \hline 2\ 4\ 6 \\ 1\ 2\ 3 \\ 0\ 0\ 0 \\ \hline 0\ 1\ 4\ 7\ 6_8 \\ \mathbf{4\ 3\ 2\ 1\ 0} \end{array}$$

Суммирование вспомогательных элементов реализуется последовательно:

- складываются номиналы цифр младшего (нулевого) разряда первого слагаемого (6), а затем формируются: новое значение в младшем разряде результирующего числа (6) и номинал для учета переноса (+0) в следующий по порядку старший (первый) разряд результирующего восьмеричного (ОСТ) числа;
- складываются номиналы цифр следующего (первого) разряда первого (4) и второго (3) слагаемых с учетом номинала переноса (+0) на предыдущей итерации, а затем формируются: новое значение в младшем разряде результирующего числа (7) и номинал для учета переноса (+0) в следующий по порядку старший (второй) разряд результирующего восьмеричного числа;
- складываются номиналы цифр следующего (второго) разряда первого (2), второго (2) и третьего (0) слагаемых с учетом номинала переноса (+0) на предыдущей итерации, а затем формируются: новое значение в младшем разряде результирующего числа (4) и номинал для учета переноса (+0) в следующий по порядку старший (третий) разряд результирующего числа;
- складываются номиналы цифр следующего (третьего) разряда первого (1) и второго (0) слагаемых с учетом номинала переноса (+0) на предыдущей итерации, а затем формируются: новое значение в младшем разряде результирующего числа (1) и номинал для учета переноса (+0) в следующий по порядку старший (четвертый) разряд результирующего восьмеричного числа;
- складываются номиналы цифр четвертого разряда первого (0) слагаемого с учетом номинала переноса (+0) на предыдущей итерации, а затем формируются: новое значение в младшем разряде результирующего числа (0), а также номинал для учета переноса (+0) в старший (пятый) разряд (0).

В результате операции арифметического сложения полученных вспомогательных элементов формируется результат в восьмеричной системе счисления: $(1476)_8$.

Возникает необходимость выполнения проверки полученного результата при помощи преобразования в DEC систему счисления посредством формулы разложения:

В результате применения формулы поразрядного разложения целой части числа:

$$(1476)_8 = 1 \cdot 8^3 + 4 \cdot 8^2 + 7 \cdot 8^1 + 6 \cdot 8^0 = 512 + 256 + 56 + 6 = 830_{10}$$

номера разрядов

2 1 0

$$(1\ 2\ 3)_8 = 1 \cdot 8^2 + 2 \cdot 8^1 + 3 \cdot 8^0 = 64 + 16 + 3 = 83_{10}$$

номера разрядов

1 0

$$(1\ 2)_8 = 1 \cdot 8^1 + 2 \cdot 8^0 = 8 + 2 = 10_{10}$$

В результате умножения первого и второго множителей в DEC $83 \cdot 10 = 830$

Результат операции арифметического умножения двух чисел в восьмеричной и десятичной системах счисления полностью идентичен, преобразование корректно.

Ответ: 1476_8 .

Пример 2.61. Выполнить умножение шестнадцатеричных чисел $1C_{16}$ и $7B_{16}$.

Для реализации операции арифметического умножения двух шестнадцатеричных чисел первый и второй множители записываются друг под другом, их разряды последовательно нумеруются: младшему разряду присваивается значение 0 (нулевой разряд), старшему разряду присваивается значение 1 (первый разряд).

номера разрядов:

$$\begin{array}{r} 10 \\ * 1C_{16} \\ \hline 7B_{16} \end{array}$$

Результат умножения двух чисел вычисляется в рамках нескольких этапов:

- формирование последовательности вспомогательных (промежуточных) чисел для обеспечения перехода от операции алгебраического умножения к операции сложения с целью упрощения вычислительных процедур;
 - умножается цифра B в младшем (нулевом) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от 0 до 1) и записываются номинальные значения полученных вспомогательных элементов для реализации процедуры последующего суммирования (134);
 - умножается цифра 7 в следующем (первом) разряде второго множителя на все цифры во всех разрядах первого множителя справа налево (от 0 до 1), осуществляется сдвиг элементов в сформированной строке на 1 разряд влево и записываются номинальные значения полученных вспомогательных элементов для обеспечения возможности последующего суммирования (C4).
- арифметическое сложение вспомогательных чисел для формирования цифр числа.

При выполнении операции арифметического умножения целой и дробной частей двух множителей результирующее значение формируется последовательно, при этом рекомендуется использовать вспомогательные таблицы умножения для ОСТ и НЕХ систем счисления с целью повышения точности проведения вычислений:

номера разрядов:

$$\begin{array}{r} 10 \\ \times 1C_{16} \\ \hline 7B_{16} \\ \hline 134 \\ \hline C4 \\ \hline D74_{16} \end{array}$$

При расчете номинальных значений вспомогательных элементов необходимо использовать таблицу умножения для шестнадцатеричной системы счисления или калькулятор, которые позволяют отслеживать номинал для переноса в старший разряд.

Полученные номинальные значения всех вспомогательных элементов записываются друг под другом и в дальнейшем подлежат процедуре арифметического суммирования, которое выполняется последовательно и поразрядно справа налево от младшего разряда (нулевой) к старшему (второй). Результат – конечное число.

Суммирование вспомогательных элементов реализуется последовательно:

- складываются номиналы цифр младшего (нулевого) разряда первого слагаемого (4), а затем формируются: новое значение в младшем (0-м) разряде результирующего числа (4) и номинал для учета переноса (+0) в следующий по порядку старший (первый) разряд результирующего HEX числа;
- складываются номиналы цифр следующего (первого) разряда первого (3) и второго (4) слагаемых с учетом номинала переноса (+0) на предыдущей итерации, а затем формируются: новое значение в младшем (1-м) разряде результирующего числа (7) и номинал для учета переноса (+0) в следующий по порядку старший (второй) разряд результирующего HEX числа;
- складываются номиналы цифр следующего (второго) разряда первого (1) и второго (C) слагаемых с учетом номинала переноса (+0) на предыдущей итерации, а затем формируются: новое значение в младшем (2-м) разряде результирующего числа (D) и номинал для учета переноса (+0) в следующий по порядку старший (третий) разряд результирующего HEX числа;
- складываются номиналы цифр третьего разряда первого (0) слагаемого и второго слагаемого (номинал переноса +0 на предыдущей итерации), формируются: новое значение в младшем разряде результирующего HEX числа (0), а также номинал для учета переноса (+0) в старший (четвертый) разряд.

В результате выполнения операции арифметического сложения вспомогательных элементов формируется результат в шестнадцатеричной системе счисления: $(D74)_{16}$.

Возникает необходимость выполнения проверки полученного результата при помощи преобразования в DEC систему счисления посредством формулы разложения:

$$(D74)_{16} = 13 * 16^2 + 7 * 16^1 + 4 * 16^0 = 3328 + 112 + 4 = 3444_{10}$$

номера разрядов

1 0

$$1C_{16} = 1 * 16^1 + 12 * 16^0 = 16 + 12 = 28_{10}$$

номера разрядов

1 0

$$7B_{16} = 7 * 16^1 + 11 * 16^0 = 112 + 11 = 123_{10}$$

В результате умножения первого и второго множителей в DEC $29 * 123 = 3444$

Результат выполнения операции арифметического умножения двух чисел в HEX и DEC системах счисления полностью идентичен, преобразование корректно.

Ответ: $D74_{16}$.

Если полученный результат в двоичной (BIN), восьмеричной (OCT) или шестнадцатеричной (HEX) системах счисления не совпадает со значением, которое было получено непосредственно при проверке в десятичной (DEC) системе счисления, то необходимо повторно осуществить расчет результирующего значения.

В двоичной системе счисления (BIN) допустимо отклонение полученного результата вследствие особенностей системы счисления и алгоритма преобразования.

2.4.4. Правила деления

Для осуществления операции арифметического деления согласно правилу (методу) деления «углом» необходимо осуществить переход к операции арифметического вычитания, выполнить ее с постоянным отслеживанием заема из старшего разряда в младший, что существенно усложняется с ростом основания заданной системы счисления (P), в которой представлены исходные числа: делимое и делитель.

Делимым выступает дробное число в виде: $A = a_{n-1} \dots a_i \dots a_1 a_0, a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m}$.

Делителем является дробное число в виде: $B = b_{n-1} \dots b_i \dots b_1 b_0, b_{-1} \dots b_{-j} \dots b_{-(m-1)} b_{-m}$.

Результат операции арифметического деления включает несколько компонентов:

- частное от деления $(c_{n-1} c_{n-2} \dots c_i \dots c_1 c_0 c_{-1} \dots c_{-j} \dots c_{-(m-1)} c_{-m})$ – дробное число (целая и дробная части), которое выступает результатом операции деления;
- остатки от деления $(r_{n-1} r_{n-2} \dots r_i \dots r_1 r_0 r_{-1} \dots r_{-j} \dots r_{-(m-1)} r_{-m})$ – целые числа, которые образованы в процессе выполнения операции деления над делимым и делителем.

$$\begin{array}{r}
 a_{n-1} a_{n-2} \dots a_i \dots a_1 a_0 a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m} \Big| B \\
 \underline{B c_{n-1}} \downarrow \\
 r_{n-1} a_{n-2} \\
 \underline{B c_{n-2}} \downarrow \\
 r_{n-2} a_i \\
 \underline{B c_i} \downarrow \\
 r_i \dots a_1 \\
 \underline{B c_1} \downarrow \\
 r_1 a_0 \\
 \underline{B c_0} \downarrow \\
 r_0 a_{-1} \\
 \underline{B c_{-1}} \downarrow \\
 r_{-1} \dots a_{-j} \\
 \underline{B c_{-j}} \downarrow \\
 r_{-j} \dots a_{-(m-1)} \\
 \underline{B c_{-(m-1)}} \downarrow \\
 r_{-(m-1)} a_{-m} \\
 \underline{B c_{-m}} \\
 r_{-m}
 \end{array}$$

При реализации вычислительных процедур и арифметических операций над числами в различных системах счисления с основанием более 10 используются специальные таблицы сложения и умножения, а также вспомогательные системы счисления: двоичная (BIN) и десятичная (DEC)

Представленные правила арифметического деления рекомендуется использовать только для двоичных чисел (BIN), поскольку деление восьмеричных (OCT) и шестнадцатеричных (HEX) чисел проще выполнять только после их перевода в десятичную (DEC) систему счисления или в двоичную систему счисления как вспомогательную.

Первый шаг

Исходное дробное число (правильная дробь) является делимым, включает явную целую и дробную части, записывается в виде $A = a_{n-1} \dots a_i \dots a_1 a_0, a_{-1} \dots a_{-j} \dots a_{-(m-1)} a_{-m}$, а делителем выступает целое или дробное число $B = b_{n-1} \dots b_i \dots b_1 b_0, b_{-1} \dots b_{-j} \dots b_{-(m-1)} b_{-m}$.

В результате операции арифметического деления исходного делимого (А) и делителя (В), которые представлены в определенной системе счисления итеративно формируется частное от деления $(c_{n-1} c_{n-2} \dots c_i \dots c_1 c_0 c_{-1} \dots c_{-j} \dots c_{-(m-1)} c_{-m})$ и образуется последовательность остатков от деления $r_{n-1} r_{n-2} \dots r_i \dots r_1 r_0 r_{-1} \dots r_{-j} \dots r_{-(m-1)} r_{-m}$.

Второй шаг

Процедура деления делимого (А) на делитель (В) выполняется последовательно:

- на n-1-м этапе реализуется операция вычитания двух слагаемых: числа образованного одной цифрой (допустимо более) исходного числа (a_{n-1}^{n-1}) и результата произведения делителя (В) на первое подобранное частное от деления (c_{n-1}^{n-1}) , а в итоге формируется первый остаток от деления (r_{n-1}^{n-1}) и первое частное от деления (c_{n-1}^{n-1}) , затем переносится следующая цифра исходного числа (a_{n-2}^{n-2}) для продолжения процедуры арифметического деления;
- на n-2-м этапе реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (r_{n-1}^{n-1}) , который дополнен справа следующей цифрой исходного числа (a_{n-2}^{n-2}) и результата произведения делителя (В) на n-2-е подобранное частное от деления (c_{n-2}^{n-2}) , а в итоге формируется n-2-й остаток от деления (r_{n-2}^{n-2}) и n-2-е частное от деления (c_{n-2}^{n-2}) , затем переносится следующая цифра исходного числа (a_i^i) для продолжения выполнения процедуры арифметического деления;
- на i-м этапе реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (r_{n-2}^{n-2}) , который дополнен справа следующей цифрой исходного числа (a_i^i) и результата произведения делителя (В) на i-е подобранное частное от деления (c_i^i) , а в итоге формируется i-й остаток от деления (r_i^i) и i-е частное от деления (c_i^i) , затем переносится следующая цифра исходного числа (a_1^1) для непосредственного обеспечения продолжения выполнения процедуры арифметического деления двух чисел в определенной системе счисления;

- на 1-м этапе реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (r_i^i) , который дополнен справа следующей цифрой исходного числа (a_1^1) и результата произведения делителя (В) на 1-е подобранное частное от деления (c_1^1) , а в итоге формируется 1-й остаток от деления (r_1^1) и 1-е частное от деления (c_1^1) , затем переносится следующая цифра исходного числа (a_0^0) для продолжения выполнения процедуры арифметического деления;
- на 0-м этапе реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (r_1^1) , который дополнен справа следующей цифрой исходного числа (a_0^0) и результата произведения делителя (В) на 0-е подобранное частное от деления (c_0^0) , а в итоге формируется 0-й остаток от деления (r_0^0) и 0-е частное от деления (c_0^0) , затем переносится следующая цифра исходного числа (a_{-1}^{-1}) для продолжения выполнения процедуры арифметического деления;
- на -1-м этапе реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (r_0^0) , который дополнен справа следующей цифрой исходного числа (a_{-1}^{-1}) и результата произведения делителя (В) на -1-е подобранное частное от деления (c_{-1}^{-1}) , а в итоге формируется -1-й остаток от деления (r_{-1}^{-1}) и -1-е частное от деления (c_{-1}^{-1}) , затем переносится следующая цифра исходного числа (a_{-j}^{-j}) для продолжения выполнения процедуры арифметического деления;
- на -j-м этапе реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (r_{-1}^{-1}) , который дополнен справа следующей цифрой исходного числа (a_{-j}^{-j}) и результата произведения делителя (В) на -j-е подобранное частное от деления (c_{-j}^{-j}) , а в итоге формируется -j-й остаток от деления (r_{-j}^{-j}) и -j-е частное от деления (c_{-j}^{-j}) , затем переносится следующая цифра исходного числа $(a_{-(m-1)}^{-(m-1)})$ для продолжения выполнения процедуры арифметического деления;
- на -(m-1)-м этапе реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления

(r_{-j}^{-j}) , который дополнен справа следующей цифрой исходного числа $(a_{-(m-1)}^{-(m-1)})$ и результата произведения делителя (В) на $-(m-1)$ -е подобранное частное от деления $(c_{-(m-1)}^{-(m-1)})$, а в итоге формируется $-(m-1)$ -й остаток от деления $(r_{-(m-1)}^{-(m-1)})$ и $-(m-1)$ -е частное от деления $(c_{-(m-1)}^{-(m-1)})$, затем переносится следующая цифра исходного числа (a_{-m}^{-m}) для продолжения выполнения арифметического деления;

- на $-m$ -м этапе реализуется операция вычитания уменьшаемого и вычитаемого: числа образованного полученным на предыдущей итерации остатком от деления $(r_{-(m-1)}^{-(m-1)})$, который дополнен справа следующей цифрой исходного числа (a_{-m}^{-m}) и результата произведения делителя (В) на $-m$ -е подобранное частное от деления (c_{-m}^{-m}) , а в итоге формируется последний ($-m$ -й) остаток от деления (r_{-m}^{-m}) и последнее ($-m$ -е) частное от арифметического деления (c_{-m}^{-m}) .

Третий шаг

Процедура арифметического деления делимого (А) на делитель (В) представленные в системе счисления с произвольным основанием (Р) завершается, при этом формируется частное от деления (набор цифр $- c_{n-1}^{n-1} c_{n-2}^{n-2} \dots c_i^i \dots c_1^1 c_0^0 c_{-1}^{-1} \dots c_{-j}^{-j} \dots c_{-(m-1)}^{-(m-1)} c_{-m}^{-m}$) и последовательность остатков от деления $(r_{n-1}^{n-1} r_{n-2}^{n-2} \dots r_i^i \dots r_1^1 r_0^0 r_{-1}^{-1} \dots r_{-j}^{-j} \dots r_{-(m-1)}^{-(m-1)} r_{-m}^{-m})$.

Четвертый шаг

Формируется и записывается результат операции деления дробных чисел.

Представленный алгоритм является рекуррентным и позволяет осуществить деление двух дробных чисел методом «угол», а делимое и(или) делитель при этом допускают отсутствие целой или дробной части, которая равняется нулю.

Таким образом, представленный алгоритм при его использовании позволяет:

- записать исходные числа (делимое А, делитель В) представленные в системе счисления с определенным основанием (Р) для последующего деления относительно введенной разрядной сетки, в которой каждая цифра обоих чисел расположена в определенном разряде с заранее заданным номером;
- получить набор номинальных значений вспомогательных чисел, которые позволяют осуществить переход от операции деления к операции вычитания;
- корректно записать полученные номинальные значения вспомогательных чисел относительно используемой разрядной сетки, а также представить их в удобной форме для последующей интерпретации и дальнейшей работы;
- реализовать переход от операции умножения двух множителей к операции поразрядного сложения всех полученных номинальных значений вспомогательных чисел, которые сосредоточены в нескольких строках (рядах).

Пример 2.62. Выполнить арифметическое деление чисел 11100101_2 и 10_2 .

Для выполнения операции деления делимое (11100101_2) и делитель (10_2) записываются друг за другом, их разряды нумеруются: младшему разряду присваивается значение 0 (нулевой разряд), старшему разряду присваивается значение 1 (первый разряд).

$$\begin{array}{r}
 \begin{array}{cccccccc}
 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\
 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 \\
 \hline
 1 & 0 & & & & & & \\
 \hline
 & 1 & 1 & & & & & \\
 \hline
 & & 1 & 0 & & & & \\
 \hline
 & & & 1 & 0 & & & \\
 \hline
 & & & & 1 & 0 & & \\
 \hline
 & & & & & 0 & 0 & 1 & 0 \\
 \hline
 & & & & & & 1 & 0 & 0 & 1
 \end{array}
 \end{array}$$

Результат деления двух чисел формируется в рамках нескольких шагов:

- на первом шаге реализуется операция вычитания уменьшаемого и вычитаемого: числа образованного двумя цифрами исходного числа (1_2^7 и 1_2^6) и результата произведения делителя (10_2) на первое подобранное частное от деления (1_2), а в итоге формируется первое частное от деления (1_2) и первый остаток от деления (1_2), затем переносится следующая цифра из пятого разряда исходного числа (1_2) для продолжения процедуры деления;
- на втором шаге реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (1_2), который дополнен справа следующей цифрой из пятого разряда исходного числа (1_2) и результата произведения делителя (10_2) на 2-е подобранное частное от деления (1_2), а в итоге формируется 2-й остаток от деления (1_2) и 2-е частное от деления (1_2), затем переносится следующая цифра из четвертого разряда исходного числа (0_2) для продолжения процедуры деления;
- на третьем шаге реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (1_2), который дополнен справа следующей цифрой из четвертого разряда исходного числа (0_2) и результата произведения делителя (10_2) на 3-е подобранное частное от деления (1_2), а в итоге формируется 3-й остаток от деления (0_2) и 3-е частное от деления (1_2), затем переносится следующая цифра из третьего разряда исходного числа (0_2) для выполнения деления;
- на четвертом шаге реализуется дополнение справа полученного на предыдущей итерации остатка от деления (0_2) следующей цифрой из третьего разряда исходного числа (0_2), а в итоге формируется промежуточное число (00_2) и записывается 4-е частное от деления (0_2), констатируется факт невозможности продолжения процедуры арифметического деления без переноса следующей цифры из последовательности цифр в составе делимого;

- на пятом шаге непосредственно реализуется перенос дополнительной цифры из второго разряда делимого (1_2) и записывается 5-е частное от деления (0_2), а также констатируется факт невозможности продолжения процедуры арифметического деления без переноса следующей цифры из делимого;
- на шестом шаге возникает необходимость переноса еще одной дополнительной цифры из первого разряда исходного числа (0_2) и подбора 6-го частного от деления (1_2) для продолжения процедуры арифметического деления;
- на седьмом шаге реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (0_2), который дополнен справа тремя цифрами исходного числа ($0_2, 1_2, 0_2$) и результата произведения делителя (10_2) на 6-е подобранное частное от деления (1_2), а в итоге формируется 5-й остаток от деления (0_2) и 6-е частное от деления (1_2), затем переносится следующая цифра из нулевого разряда исходного числа (1_2), которая выступает 4-м остатком от деления, а процедура выполнения арифметического деления сразу прекращается, поскольку дальнейшее выполнение процедуры деления невозможно.

В ходе выполнения операции арифметического деления делимого (11100101_2) и делителя (10_2) последовательно формируется результат в двоичной системе счисления в виде двух частей: частное от деления (111001_2) и остаток от деления (01_2).

Возникает необходимость выполнения проверки полученного результата при помощи преобразования в DEC систему счисления посредством формулы разложения:

номера разрядов

$$\begin{array}{cccccccccccc}
 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 & & 1 & 0 & & 5 & 4 & 3 & 2 & 1 & 0 & & 1 & 0 & & 0 \\
 (1 & 1 & 1 & 0 & 0 & 1 & 0 & 1)_2 / (1 & 0)_2 = (1 & 1 & 1 & 0 & 0 & 1)_2 * (1 & 0)_2 + (1)_2 = \\
 (1*2^5 + 1*2^4 + 1*2^3 + 0*2^2 + 0*2^1 + 1*2^0)_{10} * (1*2^1 + 0*2^0)_{10} + (1*2^0)_{10} = \\
 = (32 + 16 + 8 + 0 + 0 + 1)_{10} * (2 + 0)_{10} + (1)_{10} = (57)_{10} * (2)_{10} + (1)_{10} = (115)_{10}
 \end{array}$$

Далее преобразуем делимое и делитель в десятичную систему счисления посредством использования формулы разложения для целой части, выполним арифметическое деление и сопоставим полученный ранее результат с представленным ниже.

номера разрядов

$$\begin{array}{cccccccc}
 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\
 (1 & 1 & 1 & 0 & 0 & 1 & 0 & 1_2) / (1 & 0_2) = \\
 = (1*2^7 + 1*2^6 + 1*2^5 + 0*2^4 + 0*2^3 + 1*2^2 + 0*2^1 + 1*2^0)_{10} / (1*2^1 + 0*2^0)_{10} = \\
 = (128 + 64 + 32 + 0 + 0 + 4 + 0 + 1)_{10} / (2 + 0)_{10} = (229)_{10} / (2)_{10} = 114,5_{10}
 \end{array}$$

В результате деления делителя на делимое в DEC получим: $229_{10} / 2_{10} = 114,5_{10}$

Результат операции арифметического деления двух чисел в двоичной и десятичной системах счисления не идентичен (115 и $114,5$), что связано с особенностями алгоритма преобразования и представлением чисел в двоичной системе счисления, поэтому преобразование является корректным, а полученный результат верным.

Ответ: в результате процедуры арифметического деления делимого 11100101_2 на делитель 10_2 формируется частное от деления 111001_2 , а также остаток от деления 1_2 .

Пример 2.63. Выполнить арифметическое деление числа 123_8 на число 11_8 .

Делимое (123_8) и делитель (11_8) записываются друг за другом в нарисованный «угол», их разряды последовательно нумеруются: младшему разряду присваивается значение 0 (нулевой разряд), старшему разряду присваивается значение 2 (второй разряд).

номера разрядов

$$\begin{array}{r} 210 \ 10 \\ \underline{123 \ 11} \\ 11 \ 11 \\ \underline{ 13} \\ 11 \\ \underline{ 2} \end{array}$$

Результат деления двух чисел формируется в рамках нескольких шагов:

- на первом шаге реализуется операция вычитания двух слагаемых: числа образованного двумя цифрами исходного числа (1_8^2 и 2_8^1) и результата произведения делителя (11_8) на первое подобранное частное от деления (1_8), а в итоге формируется первое частное от деления (1_8) и первый остаток от деления (1_8), затем переносится следующая цифра исходного числа (3_8) для обеспечения возможности продолжения процедуры арифметического деления;
- на втором шаге реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (1_8), который дополнен справа следующей цифрой исходного числа (3_8) и результата произведения делителя (11_8) на 2-е подобранное частное от деления (1_8), а в итоге формируется 2-й остаток от деления (2_8) и 2-е частное от деления (1_8), при этом операция арифметического деления прекращается.

В ходе последовательного выполнения операции арифметического деления делимого (123_8) и делителя (11_8) формируется результат в двоичной системе счисления (BIN) в виде двух частей: частное от деления (11_8) и остаток от деления (2_8).

Возникает необходимость выполнения проверки полученного результата при помощи преобразования в DEC систему счисления посредством формулы разложения:

$$210 \ 10 \ 10$$

$$(123)_8 = (11)_8 * (11)_8 + (2)_8 = (1*8^1 + 1*8^0)_{10} * (1*8^1 + 1*8^0)_{10} + (2*8^0)_{10} = 83_{10}$$

Далее преобразуем делимое и делитель в десятичную систему счисления посредством использования формулы разложения для целой части, выполним арифметическое деление и сопоставим полученный ранее результат с представленным ниже.

$$210$$

$$(123)_8 / (11)_8 = (1*8^2 + 2*8^1 + 3*8^0)_{10} / (1*8^1 + 1*8^0)_{10} = 64 + 16 + 3 = 83_{10}$$

Результат операции арифметического деления двух чисел в восьмеричной и десятичной системах счисления полностью идентичен, преобразование корректно.

Ответ: в результате процедуры арифметического деления делимого 123_8 на делитель 11_8 формируется частное от деления 11_8 , а также остаток от деления 2_8 .

Пример 2.64. Выполнить арифметическое деление числа 1234567_{16} на число 11_{16} .

Делимое (1234567_{16}) и делитель (11_{16}) записываются друг за другом в нарисованный «угол», их разряды нумеруются: младшему разряду присваивается значение 0 (нулевой разряд), старшему разряду присваивается значение 6 (шестой разряд).

номера разрядов														
	6	5	4	3	2	1	0	1	0					
-	1	2	3	4	5	6	7		1	1				
	<u>1</u>	1							1	1	2	2	3	3
		1	3											
			<u>1</u>	1										
				2	4									
					<u>2</u>	2								
						2	5							
							<u>2</u>	2						
								3	6					
									<u>3</u>	3				
										3	7			
											<u>3</u>	3		
												4		

Результат деления двух чисел формируется в рамках нескольких шагов:

- на первом шаге реализуется операция вычитания двух слагаемых: числа образованного двумя цифрами исходного числа (1_{16}^6 и 2_{16}^5) и результата произведения делителя (11_{16}) на первое подобранное частное от деления (1_{16}), а в итоге формируется первое частное от деления (1_{16}) и первый остаток от деления (3_{16}), затем переносится следующая цифра исходного числа (3_{16}) для продолжения выполнения процедуры арифметического деления чисел;
- на втором шаге реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (1_{16}), который дополнен справа следующей цифрой исходного числа (3_{16}) и результата произведения делителя (11_{16}) на 2-е подобранное частное от деления (1_{16}), а в итоге формируется 2-й остаток от деления (2_{16}) и 2-е частное от деления (1_{16}), затем переносится следующая цифра исходного числа (4_{16}) для реализации возможности продолжения операции арифметического деления;
- на третьем шаге реализуется операция вычитания двух слагаемых: числа образованного полученным на предыдущей итерации остатком от деления (2_{16}), который дополнен справа следующей цифрой исходного числа (4_{16}) и результата произведения делителя (11_{16}) на 3-е подобранное частное от деления (2_{16}), а в итоге формируется 3-й остаток от деления (2_{16}) и 3-е частное от деления (2_{16}), затем переносится следующая цифра исходного числа (5_{16}) для продолжения выполнения процедуры арифметического деления чисел;

3. Логические основы цифровых автоматов

Алгебра логики подразделяется на два основных направления или раздела:

- неформальная – основана на вербальных высказываниях и повествовательных предложениях, которые содержат истинные или ложные факты;
- формальная – исчисление высказываниями и предикатами первого и второго порядка, которые формализованы в виде логических уравнений и функций.

Формальная алгебра логики или Булева алгебра логики создана в XIX веке английским математиком Джорджем Булем и основана на операндах и операциях.

Определение

Алгебра логики – это математический аппарат, с помощью которого записывают, вычисляют, упрощают и преобразовывают логические выражения или высказывания.

Алгебра логики (формальная и неформальная) рассматривает любое высказывание только с одной точки зрения – является ли оно истинным или ложным.

Определение

Логическое высказывание – любое повествовательное предложение или выражение, в отношении которого можно однозначно сказать, истинно оно или ложно.

Логические операции и кванторы находятся в формальных логических выражениях (\vee , $\&$, \neg , \forall , \exists) и вербальных высказываниях (ИЛИ, И, НЕ, любой, существует).

В результате выполнения любой логической операции над двумя двоичными (бинарными) числами формируется новое число в двоичной системе счисления.

Аппаратная реализация устройства осуществляющего вычисление результата логической функции рассматривается как функциональный преобразователь на входы которого подаются сигналы согласно двум логическим состояниям, а на выходах достигается набор уровней напряжения соответствующих бинарным значениям (рис. 3.1).

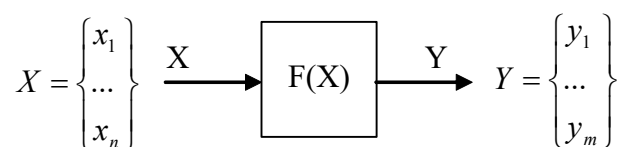


Рисунок 3.1. Логическая функция как функциональный преобразователь

Входы указанного функционального преобразователя соответствуют аргументам (операндам X) простой или сложной логической функции; $F(X)$ – логическая функция, которая задает соответствие между входами и выходами устройства; выходы отражают установившееся состояние функционального преобразователя и выступают результирующими значениями заданной логической функции $Y=F(X)$.

Вектор входа (X) представляет собой совокупность сигнальных линий (x_i), которые обеспечивают поступление входных воздействий соответствующих двоичным комбинациям. Вектор выхода (Y) отражает состояние набора сигнальных линий (y_i), которым соответствуют уровни сигналов на выходах логического устройства.

Определение

Логическими или булевыми операндами и функциями называют бинарные переменные и функции, которые принимают два логических состояния (0 и 1).

3.1. Логические переменные и функции алгебры логики

Алгебра логики обеспечивает создание, преобразование и упрощение логических функций посредством использования простых и сложных логических операций.

Определение

Операндом (в программировании) называют константу или переменную, которая соответственно статична или динамична по отношению к ее значению.

Определение

Функцией называют совокупность операндов и связей между ними, которые соответствуют набору операций разного вида: аддитивные, мультипликативные, степенные, логарифмические, тригонометрические, логические или бинарные.

Определение

Логическим операндом называют константу или переменную, которая принимает одно из двух устойчивых логических состояния: 1 – истина, 0 – ложь.

Определение

Функцией алгебры логики, переключательной или логической функцией называют функцию одного или более логических операндов и логических связей (операций) между ними, результирующее (возвращаемое) значение которой соответствует двум устойчивым логическим состояниям: 1 – истина(ность) или 0 – ложь(ность).

Совокупность значений логических операндов (x_1, x_2, \dots, x_n) и результирующих значений логической функции (y_1, y_2, \dots, y_m) представляются в виде наборов двоичных комбинаций, которые можно записать в таблицу истинности (табл. 3.1).

Таблица 3.1

Двоичные комбинации значений логических операндов и логических функций

Номер комбинации	Значение операнда 1	Значение операнда 2	Значение операнда n	Значение логической функции 1	Значение логической функции 2	Значение логической функции m
1	$a_{11}=0$	$a_{12}=0$	$a_{1n}=1$	$y_{11}=1$	$y_{12}=0$	$y_{1m}=1$
2	$a_{21}=1$	$a_{22}=1$	$a_{2n}=0$	$y_{21}=0$	$y_{22}=0$	$y_{2m}=0$
...
N,M	$a_{N1}=1$	$a_{N2}=0$	$a_{Nn}=0$	$y_{M1}=1$	$y_{M2}=1$	$y_{Mm}=0$

Запись из набора значений полей, которые отражают состояния логических операндов и функций называются соответственно двоичными входными и выходными комбинациями, где коэффициенты a_{in} – значения логических операндов, а y_{jm} – результирующие значения логических функций, которые равны 0 или 1.

Количество разных наборов элементов (двоичных комбинаций) конечномерно:

- для значений логических операндов – $N=2^n$;
- для результирующих значений логических функций – $M=2^m$.

Покажем это на элементарном примере, который включает несколько наборов.

набор 1: $\{(0), (0), (1)\}$; набор 1: $\{(1), (0), (1)\}$ – 2 комбинации по 3 значения;

набор 2: $\{(1), (1), (0)\}$; набор 2: $\{(0), (0), (0)\}$ – 2 комбинации по 3 значения;

... ; ...

набор N: $\{(1), (0), (0)\}$; набор M: $\{(1), (1), (0)\}$ – 2 комбинации по 3 значения.

3.2. Способы задания и реализации логических функций

Любую функцию алгебры логики (логическую функцию) можно задать одним из следующих способов (рис. 3.2): табличный (таблица истинности), аналитический (формальный), физический (аппаратная реализация), графический (функциональная схема и электрическая принципиальная схема логического устройства).

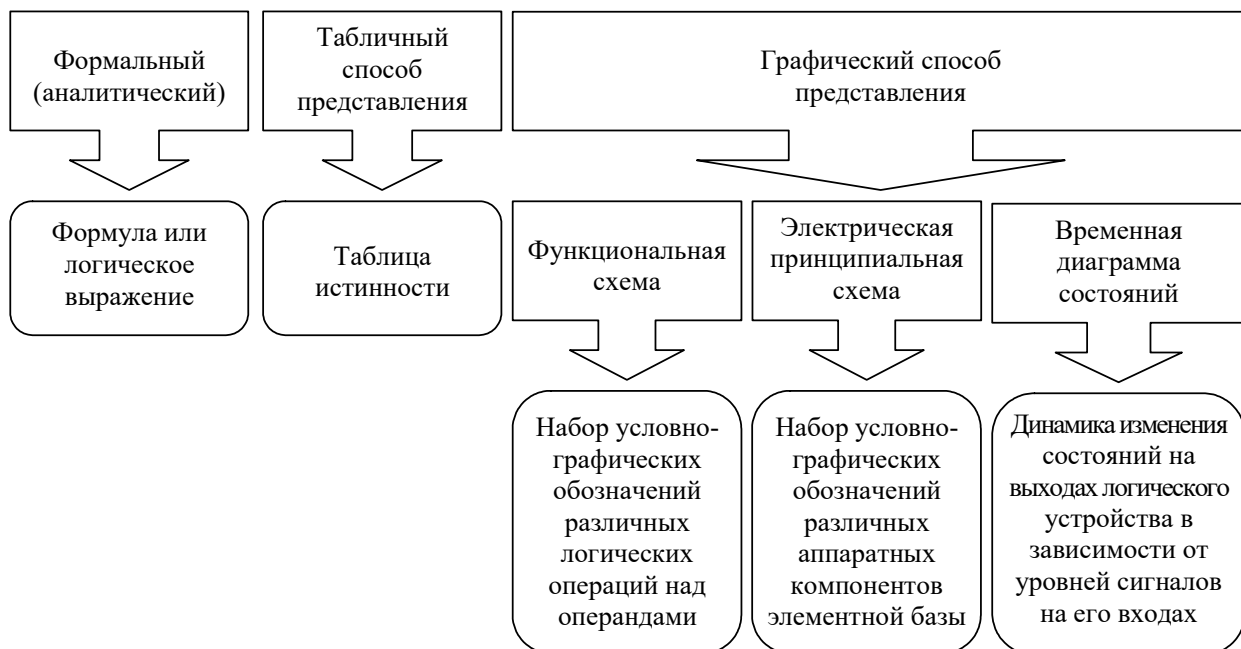


Рисунок 3.2. Способы задания и реализации логических функций

Определение

Таблицей истинности называют матрицу, с помощью которой задают взаимно однозначное соответствие между комбинациями значений логических операндов (состояний на входах логического устройства) и возвращаемыми значениями логической функции (достигнутыми состояниями на выходах логического устройства).

Таблица истинности имеет обобщенный вид и представлена в табл. 3.2

Таблица 3.2

Обобщенная структура типовой таблицы истинности логической функции

Номер набора (комбинации) логических переменных	Аргументы логической функции (номиналы логических переменных)				Результирующее значение логической функции			
	x_1	x_2	x_i	x_n	$y_1=F_1(X)$	$y_2=F_2(X)$	$y_j=F_j(X)$	$y_m=F_m(X)$
k	x_1	x_2	x_i	x_n	$y_1=F_1(X)$	$y_2=F_2(X)$	$y_j=F_j(X)$	$y_m=F_m(X)$
0	0	0	0	0	0	1	1	1
1	0	0	0	1	1	1	0	0
...
N	1	1	1	1	0	1	0	0

Для систематизации и удобства поиска двоичной комбинации значений логических операндов ($X = \{x_1, x_2, \dots, x_i, \dots, x_n\}$) и функций ($X = \{y_1, y_2, \dots, y_j, \dots, y_m\}$) в таблице истинности вводится левый столбец (k), который содержит номер набора (двоичной комбинации) и позволяет исключить возможную ошибку при записи и интерпретации строки значений при существенном увеличении количества наборов ($k=N|M \rightarrow \infty$).

Правило построения таблицы

1. По известному количеству входов (n) и выходов (m) можно определить разрядность вектора входа и вектора выхода как совокупность сигнальных линий, а также число операндов (n) и логических операций (m) над ними, которые отражают число различных наборов бинарных элементов N и M в таблице истинности.

2. Нумеруют наборы бинарных элементов k целыми числами начиная с 0 до N и M .

3. Каждый элемент бинарного набора (значение логического операнда i или логической операции j) рассматривается в виде двоичного числа разрядности n или m .

4. Каждый набор различных значений бинарных переменных и логических функций записывают в соответствующей строке сформированной таблицы истинности.

5. Предыдущее действие повторяют для каждого бинарного набора и получают слева в таблице истинности входные наборы, соответствующие значениям логических операндов (X), а затем заполняют правый столбец таблицы истинности на основании заданных функциональных зависимостей в виде логических функций (Y).

Определение

Формулой логической функции называют математическое выражение алгебры логики, включающее совокупность логических операндов (переменных) и логических операций над ними, которые обозначены различными символами.

Любое логическое высказывание в виде повествовательного предложения можно формализовать, то есть заменить его аналитическим представлением в виде формулы.

При аналитическом (формальном) способе задания переключательной функции используется принцип суперпозиции: обеспечивается подстановка одной логической функции вместо аргумента другой. Используя принцип суперпозиции можно получить любую сколь угодно сложную функцию из линейной комбинации простых.

В аналитическом выражении порядок выполнения логических операций определяется приоритетом логических функций (операций), а если его необходимо изменить, то используют круглые скобки. Наивысший приоритет в последовательности выполнения логических операций над логическими операндами имеет функция НЕ, скобки, затем И – конъюнкция, а наименьший имеет функция ИЛИ – дизъюнкция.

Пример 3.1. Необходимо записать две представленные логические функции с использованием принципа суперпозиции на основе логического выражения

Пусть заданы две логические функции, которые взаимосвязаны между собой:
 $a = b \& x_1 \vee x_2 \& b \vee x_1 \& b \vee x_2 \vee b$, а также $b = x_1 \vee x_2$.

При подстановке логической функции b в логическую функцию a , имеем:
 $a = x_1 \vee x_2 \& x_1 \vee x_2 \& x_1 \vee x_2 \vee x_1 \& x_1 \vee x_2 \vee x_2 \vee x_1 \vee x_2$.

Пример 3.2. Записать логические функции на основе принципа суперпозиции

Пусть заданы две логические функции, которые взаимосвязаны между собой:
 $a = b \& (x_1 \vee x_2) \& b \vee x_1 \& b \vee (x_2 \vee b) \& (x_1 \vee b)$, а также $b = x_1 \vee x_2 \& 1$.

При подстановке логической функции b в логическую функцию a , имеем:
 $a = x_1 \vee x_2 \& 1 \& (x_1 \vee x_2) \& x_1 \vee x_2 \& 1 \vee x_1 \& x_1 \vee x_2 \& 1 \vee (x_2 \vee x_1 \vee x_2 \& 1) \& (x_1 \vee x_1 \vee x_2 \& 1)$.

Пример 3.3. Необходимо записать две логические функции с использованием принципа суперпозиции по отношению к третьей логической функции

Пусть заданы две логические функции в основе системы логических уравнений:

$$\begin{cases} a = x_1 \& x_2, \\ b = x_1 \vee x_2. \end{cases}$$

Задана аналитическая зависимость между несколькими логическими операндами:

$$c = a \& (b \vee x_1) \& b \vee a \& x_2 \& (x_1 \vee a)$$

Тогда, учитывая результат подстановки двух логических функций a и b в логическую функцию c , имеем аналитическое представление логического выражения:

$$c = x_1 \& x_2 \& (x_1 \vee x_2 \vee x_1) \& x_1 \vee x_2 \vee x_1 \& x_2 \& x_2 \& (x_1 \vee x_1 \& x_2).$$

Аналитический способ представления (формула) позволяет задать сложную логическую функцию значительно компактнее, чем табличный (таблица истинности). Однако табличный способ удобно и наглядно отражает зависимость возвращаемых значений логической функции в зависимости от значений логических операндов.

Аппаратные устройства (преобразователи) и логические операции, которые реализуют переключательные функции изображаются графически в виде различных схем (электрических принципиальных и функциональных) с использованием стандартизованных условно-графических обозначений разных логических операций.

Определение

Функциональная схема логического устройства (рис. 3.3), обеспечивающего вычислительную процедуру адекватно заложенной в его основу логической функции, предназначена для структурного представления взаимосвязанных логических операций над операндами в виде совокупности условно-графических обозначений.



Рисунок 3.3. Условно-графические обозначения логических операций на функциональных схемах

Определение

Электрическая принципиальная схема аппаратной реализации логического устройства представляет собой набор условно-графических обозначений полупроводниковых элементов в рамках используемой элементной базы, которые обеспечивают реализацию простой или сложной логической операции или функции.

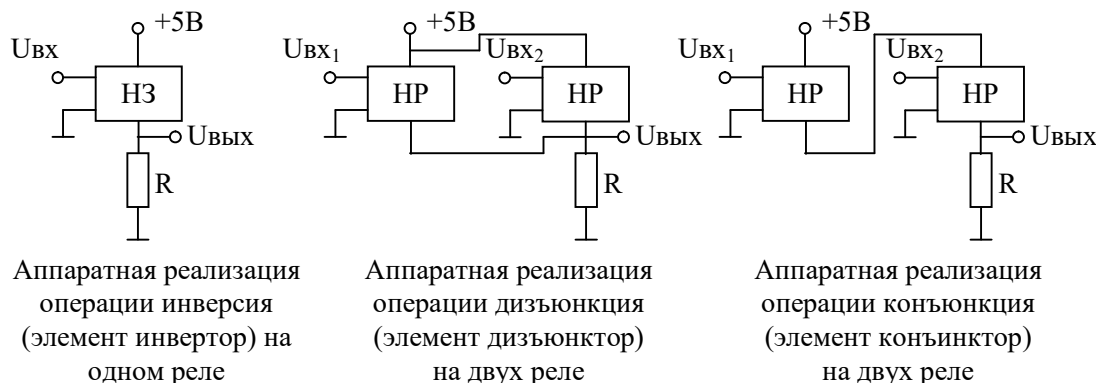


Рисунок 3.4. Условно-графические обозначения логических операций на функциональных схемах

На рис. 3.4 представлены три электрические принципиальные схемы аппаратной реализации различных логических устройств, которые содержат условно-графические обозначения некоторых простейших логических операций (логических функций) над одним или несколькими значениями логических операндов:

- инверсия (НЕ) – операция логического отрицания, элемент инвертор;
- дизъюнкция (ИЛИ) – операция логического сложения, элемент дизъюнктор;
- конъюнкция (И) – операция логического умножения, элемент конъюнктор.

Каждая электрическая принципиальная схема простейшего логического устройства собрана на основе релейной элементной базы и содержит ряд обозначений:

- +5В – входной контакт реле, на который подается опорное напряжение номиналом 5В для обеспечения питания заданной электрической схемы;
- $U_{ВХi}$ – управляющий контакт реле, который предназначен для подачи напряжения номиналом 5В, что инициирует замыкание нормально разомкнутого реле и размыкание нормально замкнутого реле на разных схемах;
- $U_{ВЫХi}$ – выходной контакт реле, который предназначен для регистрации эффективного сигнала в виде напряжения номиналом 5В, после замыкания нормально разомкнутого реле и размыкания нормально замкнутого реле;
- НР – нормально разомкнутое реле обеспечивает коммутацию сигнальной линии при подачи опорного напряжения на управляющие контакты $U_{ВХi}$ и ноль;
- НЗ – нормально замкнутое реле обеспечивает коммутацию сигнальной линии при отсутствии опорного напряжения на управляющих контактах;
- R – сопротивление большого номинала (от 100КОм), которое обеспечивает возможность гальванической развязки для регистрации разности потенциалов (напряжения) между клеммами +5В и ноль, $U_{ВХi}$ и ноль, $U_{ВЫХi}$ и ноль.

Определение

Временная диаграмма отражает динамику изменения состояний уровней сигналов на выходах логического устройства в зависимости от комбинаторных сочетаний уровней сигналов на его входах, при этом аппаратная реализация устройства представлена в виде электрической принципиальной схемы или функциональной схемы.

Временная диаграмма содержит в своей основе несколько графиков функций непрерывного или дискретного времени, которые отражают динамику переключения состояний на выходах логического элемента (устройства) в зависимости от уровней сигналов на его входах. На рис. 3.5-3.7 представлены временные диаграммы инвертора (операция НЕ), дизъюнктора (операция ИЛИ) и конъюнктора (операция И).

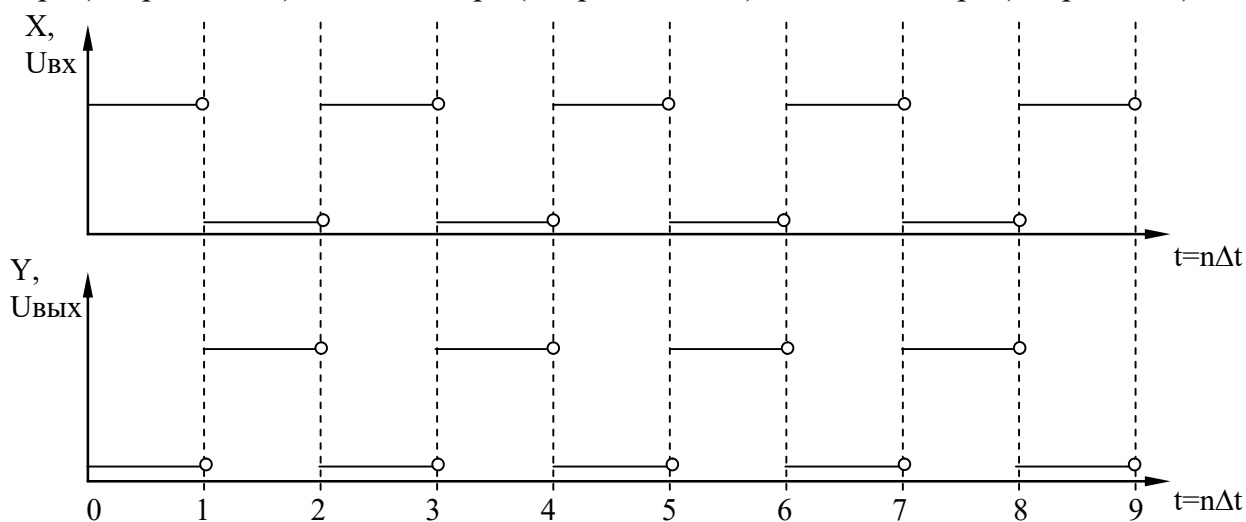


Рисунок 3.5. Временная диаграмма состояний инвертора

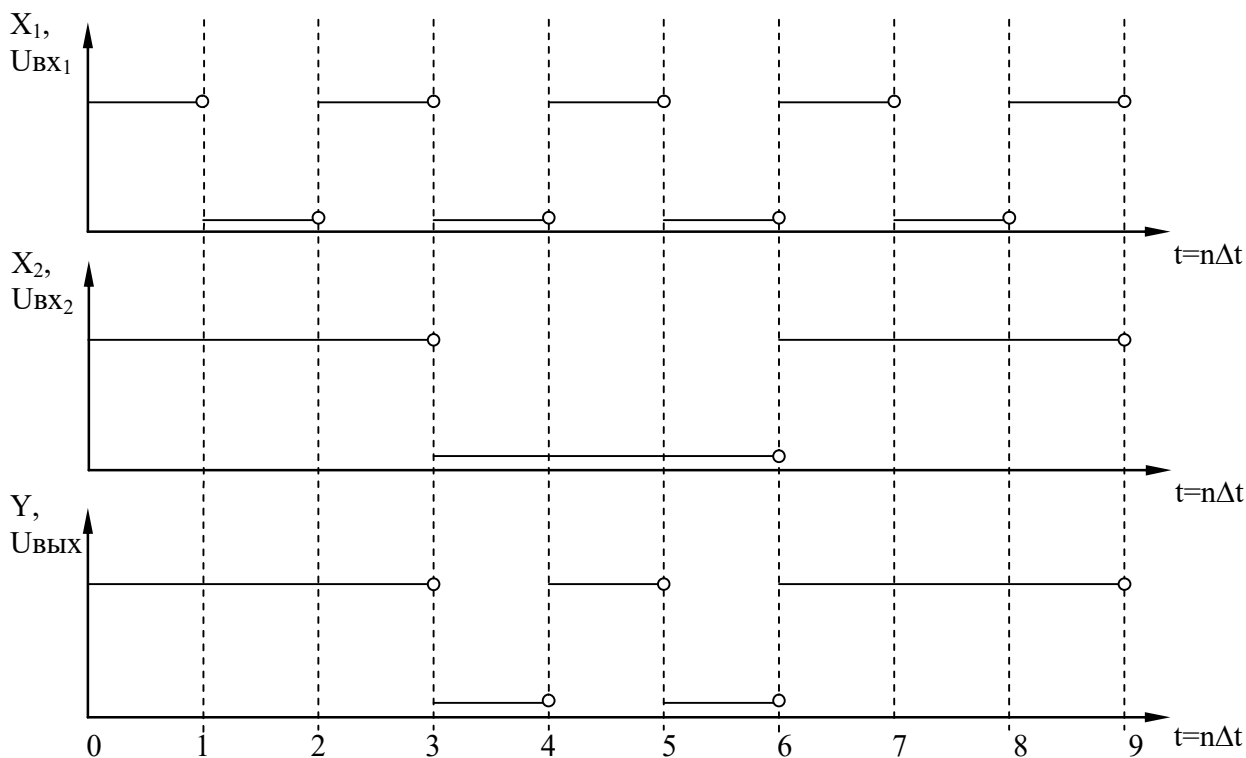


Рисунок 3.6. Временная диаграмма состояний дизъюнктора

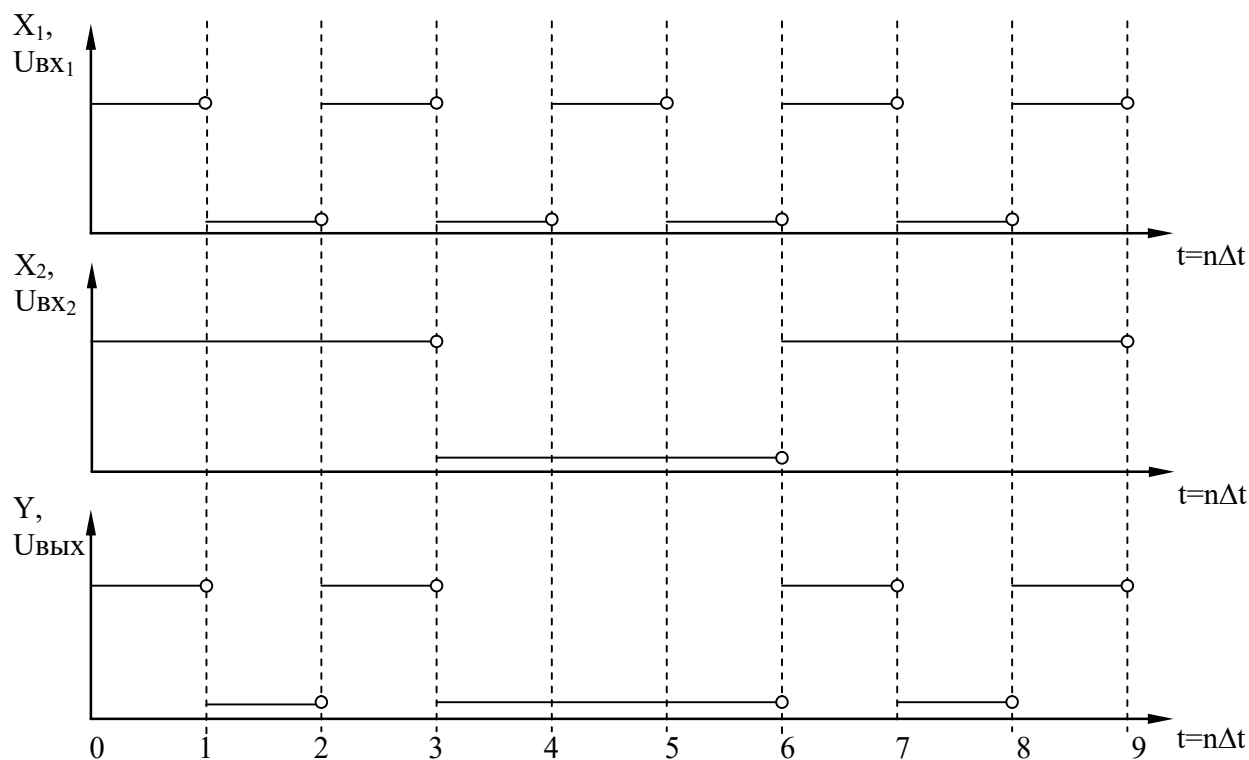


Рисунок 3.7. Временная диаграмма состояний конъюнктора

Временная диаграмма изменения состояний отражает поведение непрерывного или дискретного сигнала в пространстве двух координат, при этом возможно:

- определение значения уровня сигнала в любой произвольный или фиксированный момент времени соответственно t или $n\Delta t$ с помощью графика;
- проверка соответствия таблицы истинности и принципа функционирования логического устройства, которое реализует определенную простую или сложную логическую операцию или логическую функцию над операндами;
- проверка принципа функционирования логического устройства согласно разработанной функциональной схеме, электрической принципиальной схеме, имеющейся таблицы истинности, а также теоретической или практической временной диаграммы изменения состояний выходов и входов;
- разработка путей повышения эффективности функционирования имеющихся функциональных и электрических принципиальных схем на основе современной элементной базы, которая используется для их реализации;
- усовершенствование алгоритмического и программного обеспечения в основе современной архитектуры компьютеров и информационных систем;
- разработка программного обеспечения для реализации электрических принципиальных схем и функциональных схем, которые отражают принципы и алгоритмы функционирования различных логических устройств;
- разработка нового и модификация существующего алгоритмического обеспечения для эффективной работы арифметико-логического устройства расположенного в основе процессора современных цифровых автоматов.

3.3. Основные логические операции

Логическая функция выступает оператором преобразования значений логических операндов или сигналов на входе логического устройства в совокупность результирующих значений вектора выхода, который формируется на основе совокупности логических операций, задающих способ преобразования значений логических операндов.

Существует связь между неформальной и формальной алгеброй логики, что позволяет на основе простого или сложного логического выражения сформировать логическую функцию, которая включает основные и производные логические операции.

Определение

Сложное логическое выражение можно структурно декомпозировать к совокупности простых путем использования набора правил эквивалентного преобразования, которыми выступают **законы алгебры логики (Булевой алгебры логики)**.

Сложная логическая функция как правило включает набор сложных и производных (гибридных) логических операций, которые образуются из набора простых логических операций и выступают ее логически неделимыми частями (рис. 3.8).



Рисунок 3.8. Простые и гибридные логические операции и функции

Структура логического выражения задает вид функции алгебры логики (рис. 3.9).

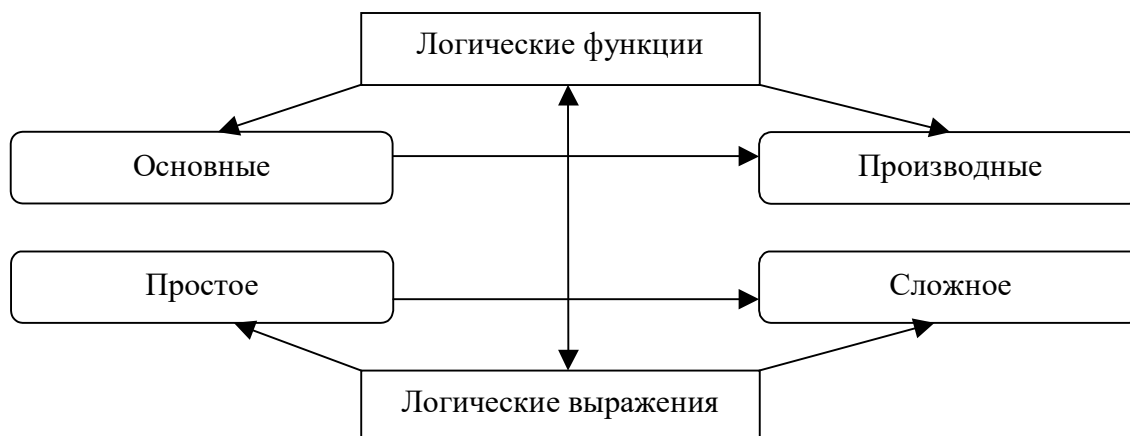


Рисунок 3.9. Взаимосвязь между структурой логического выражения и видом функции алгебры логики

Последовательность преобразования исходного вербального логического выражения в логическую функцию или операцию представляется в следующем виде:

- проверка соответствия исходного вербального выражения или предложения на возможность его преобразования в аналитическую функцию алгебры логики:
 - исходное логическое выражение проверяется на соответствие возможности формирования логической функции: его формулировка содержит последовательность заданных фактов как прототипов логических операндов;
 - существует возможность ввода набора логических связок между введенными логическими операндами и их значениями вместо логических выражений;
 - исходное логическое выражение имеет истинный или ложный результат;
 - формирование логической функции из исходного логического выражения:
 - структурная декомпозиция исходного логического выражения (предложения);
 - создание перечня фактов на основе разных частей логического выражения, об истинности или ложности которых можно точно и однозначно судить;
 - определение перечня логических операций, которые выступают связками между различными фактами (операндами) на основании исходного выражения;
 - запись логического выражения посредством использования формулы, которая выступает его аналитическим представлением посредством разных символов:
 - формирование аналитического логического выражения, которое включает последовательность операндов и логических операций между каждым из них;
 - проверка полученного логического выражения в виде аналитической формулы:
 - формирование разных таблиц истинности и функциональных схем на основе исходного вербального выражения и полученной логической функции;
 - подстановка значений логических операндов в полученную аналитическую логическую функцию, выполнение логических операций над операндами;
 - подача номиналов уровней сигналов (напряжения) на входы логического устройства, последующее измерение достигнутых уровней сигналов на его выходах.
- Далее предлагается рассмотреть совокупность простых логических операций.

3.3.1. Операция инверсии НЕ, функция логического отрицания

Операция логического отрицания (НЕ) на функциональных схемах обозначается в виде условно-графического обозначения инверсии, а также аппаратно реализуется посредством использования инвертора, который имеет один вход и один выход.

Определение

Логическое отрицание (НЕ) аппаратно реализуется посредством инвертора, является функцией одного логического операнда и определяет инверсию его значения:

- значение функции истинно в случае ложности значения логического операнда;
- значение логической функции ложно тогда, когда истинно значение операнда.

Аналитический способ представления посредством формулы

$$Y = \text{НЕ}(X), \quad y = \bar{x}$$

При использовании аналитической записи функции логического отрицания (НЕ) в виде формулы операцию логического отрицания (инверсию) обозначают в виде черты над соответствующим логическим операндом независимо от контекста.

Табличный способ представления при помощи таблицы истинности

Соответствие между результирующими значениями операции логической инверсии и значениями логического операнда задаются посредством таблицы истинности.

Таблица истинности операции логической инверсии представлена в табл. 3.3:

Таблица 3.3

**Таблица истинности функции логического отрицания (НЕ)
или операции логической инверсии значения логического операнда**

Значение логического операнда X	Значение логической функции $Y = \text{НЕ}(X)$
0	1
1	0

Функциональная схема

На функциональной схеме операция инверсии (НЕ) изображается посредством следующего стандартизованного условно-графического изображения (рис. 3.10).

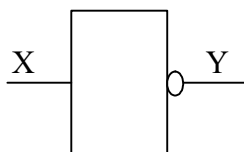


Рисунок 3.10. Условно-графическое обозначение операции логической инверсии, функции логического отрицания или логического элемента «инвертор»

Электрическая принципиальная схема

Для физической (аппаратной) реализации функции логического отрицания (НЕ), операции инверсия, а также элемента инвертор используется элементная база микроэлектроники (реле, полупроводниковый транзистор, диод, резистор и прочие).

Электрическая принципиальная схема аппаратной реализации операции логического отрицания (НЕ) на базе транзистора и резистора (рис. 3.11, а) и нормально-замкнутого реле и резистора (рис. 3.11, б), временная диаграмма состояний инвертора (рис. 3.12) и эпюры уровней напряжения на его входе и выходе (табл. 3.4) представлены далее.

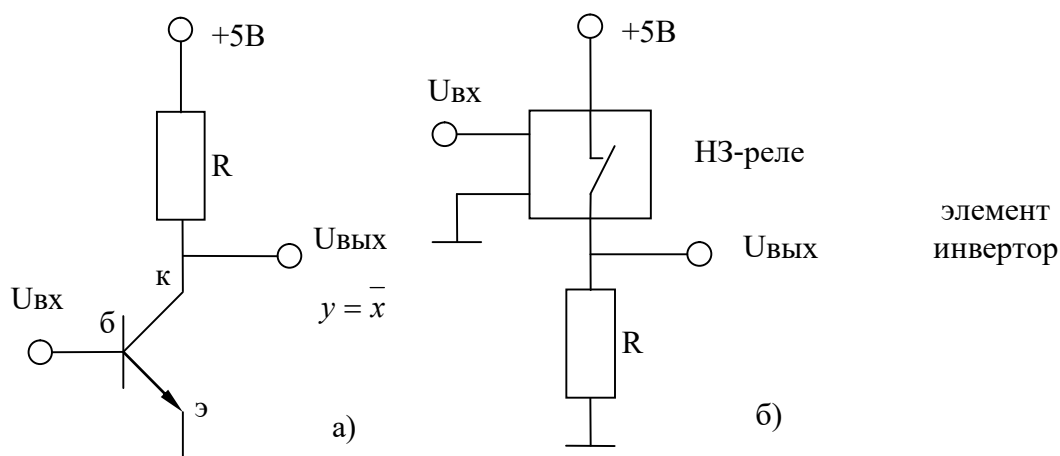


Рисунок 3.11. Электрическая принципиальная схема аппаратной реализации инвертора, функции логического отрицания, логической операции инверсия (НЕ)

Таблица 3.4

Таблица истинности и эпюры напряжений на входах и выходах инвертора

Значение логического операнда X	Уровень напряжения на входе Uвх	Значение логической функции Y=НЕ(X)	Уровень напряжения на выходе Uвых
0	0В	1	5В
1	5В	0	0В

Таблица истинности элемента инвертор отражает взаимно однозначное соответствие между пороговыми уровнями напряжения (значениями логического операнда) на входе логического устройства (логической функции) и достигнутыми состояниями на выходе устройства (результатирующими значениями логической функции).

Приведем некоторые пояснения к таблице истинности элемента «инвертор».

Электрический ток представляет собой упорядоченное движение заряженных частиц и направлен противоположно движению потока электронов по проводнику.

На рис. 3.11 используются следующие обозначения элементов и контактов:

- эмиттер – контакт, выступающий источником носителей заряда (упорядоченного движения частиц): электронов в ррр- или дырок в рпн-транзисторе;
- коллектор – электрод, который является стоком носителей заряда (электронов) поступающих от эмиттера транзистора как полупроводникового элемента;
- база – управляющий электрод, в котором сигнал на входе влияет на величину тока заряженных частиц, протекающего через транзистор или на интенсивность управляемого стока носителей заряда от эмиттера к коллектору.

Транзистор в этой схеме работает в режиме ключа и имеет два состояния:

- закрытое состояние транзистора – через полупроводниковый прибор не протекает электрический ток, а его внутреннее сопротивление стремится к бесконечности при отсутствии управляющего напряжения на клемме Uвх (база);
- открытое состояние транзистора – полупроводниковый прибор динамически переходит в режим насыщения и через него протекает максимально возможный ток, а при этом его внутреннее сопротивление стремится к нулю.

Временная диаграмма

Представленная на рис. 3.12 временная диаграмма графически отражает динамику изменения уровней напряжения соответствующих логическим состояниям (0 и 1), которые достигаются на выходе инвертора под влиянием изменения уровней напряжения на его входе при реализации инверсии значения логического операнда.

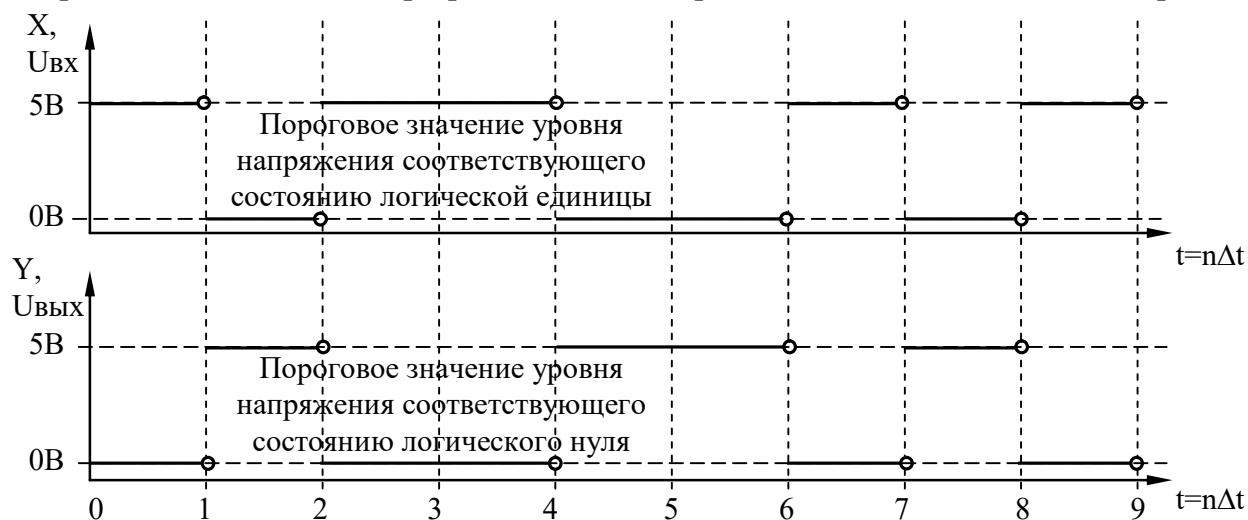


Рисунок 3.12. Временная диаграмма динамики изменения состояний на выходах инвертора в зависимости от уровней сигналов на его входах

Прокомментируем полученную временную диаграмму состояний инвертора:

- на интервале между нулевым и первым дискретными моментами времени ($t|_{n=0,1} = n\Delta t$) пороговое значение напряжения на входе инвертора X ($U_{вх}$) соответствует состоянию логической единицы (1), при этом на выходе инвертора Y ($U_{вых}$) достигается взаимно инверсное состояние логического нуля (кружок на графике означает исключение точки в момент переключения устройства при $n=0, 1$);
- на интервале между первым и вторым дискретными моментами времени ($t|_{n=1,2} = n\Delta t$) пороговое значение напряжения на входе инвертора X ($U_{вх}$) соответствует состоянию логического нуля (0), при этом на выходе инвертора Y ($U_{вых}$) достигается взаимно инверсное состояние логической единицы (1);
- на интервале между вторым и третьим дискретными моментами времени ($t|_{n=2,3} = n\Delta t$) пороговое значение напряжения на входе инвертора X ($U_{вх}$) соответствует состоянию логической единицы (1), при этом на выходе инвертора Y ($U_{вых}$) достигается взаимно инверсное состояние логического нуля (0);
- на интервале между третьим и четвертым дискретными моментами времени ($t|_{n=3,4} = n\Delta t$) пороговое значение напряжения на входе инвертора X ($U_{вх}$) соответствует состоянию логической единицы (1), при этом на выходе инвертора Y ($U_{вых}$) достигается взаимно инверсное состояние логического нуля (0);
- на последующих интервалах расчет логических состояний на входе и выходе инвертора, реализующего функцию логической инверсии (НЕ) аналогичен.

3.3.2. Операция конъюнкции, функция логического умножения И

Определение

Операцией конъюнкции (И) является функция логического умножения значений двух логических операндов, возвращаемое значение которой истинно тогда и только тогда, когда одновременно достигается истинность обоих значений входящих в нее логических операндов, при этом аппаратно реализуется посредством конъюнктора.

Эта логическая функция имеет несколько названий: логическое умножение, конъюнкция, логическая операция или функция И, логическая связка &.

Аналитический способ представления

При использовании аналитической записи посредством формулы функция логического умножения (конъюнкция, операция И) изображается посредством символа &.

Запись “ $x_1 \& x_2$ ” читается как « x_1 И x_2 ».

Табличный способ представления посредством таблицы истинности

Таблица истинности элемента конъюнктор, операции конъюнкции (И), функции логического умножения двух значений логических операндов представлена в табл. 3.5.

Таблица 3.5

Таблица истинности операции конъюнкция (И) или функции логического умножения значений двух логических операндов

Значение первого логического операнда X_1	Значение второго логического операнда X_2	Значение логической функции $Y=X_1 \& X_2$
0	0	0
0	1	0
1	0	0
1	1	1

Таблица истинности задает взаимно однозначное соответствие между двоичными значениями логических операндов и результирующим значением операции И.

В таблице истинности представлено 4 бинарных комбинации значений входных операндов и соответствующие им результирующие значения логической функции.

Функциональная схема

На функциональной схеме логического устройства или функции операция конъюнкции двух значений логических операндов изображается посредством использования следующего стандартизованного условно-графического обозначения (рис. 3.13).

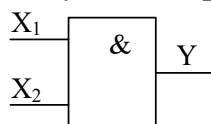


Рисунок 3.13. Условно-графическое обозначение элемента конъюнктор, операции конъюнкция (И), функции логического умножения значений двух операндов

Представленное обозначение операции логического умножения И используется на функциональных схемах логических устройств и функций в вертикальном исполнении, при этом допустимо отсутствие обозначений логических операндов.

Уровни напряжения однозначно соответствуют значениям логических операндов на входе и достигнутым логическим состояниям на выходе аппаратной реализации конъюнкции, что отражает таблица истинности и временная диаграмма.

Электрическая принципиальная схема

Далее представлены простейшая электрическая принципиальная схема, содержащая аппаратную реализацию конъюнкции двух операндов (рис. 3.14), а также таблица эшюров напряжений на входах и выходе конъюнктора (табл. 3.4) и временная диаграмма динамики изменения бинарных состояний на выходах и входах (рис. 3.15).

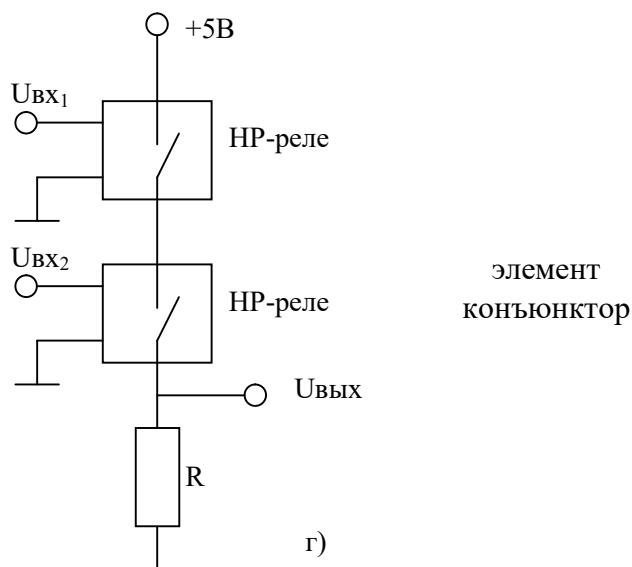


Рисунок 3.14. Электрическая принципиальная схема аппаратной реализации конъюнктора, функции логического умножения, операции конъюнкции (И)

На представленной электрической принципиальной схеме введены обозначения:

- клемма +5В – обеспечивает подачу опорного напряжения номиналом 5В;
- клеммы $U_{вх1}$ – предназначены для подачи напряжения, обеспечивающего управление коммутацией нормально разомкнутого реле: подача напряжения 5В реализует замыкание контакта, снятие – размыкание контакта;
- два нормально разомкнутых (НР) реле – пропускают эффективный сигнал на $U_{вых}$ только при подаче опорного напряжения на управляющие входы $U_{вх1}$, $U_{вх2}$;
- клемма $U_{вых}$ – позволяет обеспечить регистрацию напряжения на выходе.

Регистрация эффективного сигнала (напряжения) на $U_{вых}$ возможна только в случае наличия опорного напряжения между клеммами +5В и ноль, а также при одновременной подаче управляющего напряжения на клеммах $U_{вх1}$ и $U_{вх2}$, что обеспечивает коммутацию обоих контактов двух нормально-разомкнутых реле.

Далее представлена табл. 3.6, которая отражает соответствие между таблицей истинности конъюнкции двух логических операндов и эшюрами напряжений на входах и выходах аппаратной реализации операции логического умножения (И).

Таблица 3.6

Таблица истинности и эшюры напряжений на входах и выходах конъюнктора

X_1	X_2	$U_{вх1}$	$U_{вх2}$	$Y=X_1 \& X_2$	$U_{вых}$
0	0	0В	0В	0	0В
0	1	0В	5В	0	0В
1	0	5В	0В	0	0В
1	1	5В	5В	1	5В

Временная диаграмма

Временная диаграмма состояний конъюнктора (рис. 3.15) как аппаратной реализации конъюнкции двух логических операндов отражает динамику изменения уровней сигналов соответствующих двум логическим состояниям (0 и 1), которые достигаются на выходе ($U_{\text{вых}}$) под влиянием уровней напряжения на двух входах ($U_{\text{вх}_1}$ и $U_{\text{вх}_2}$).

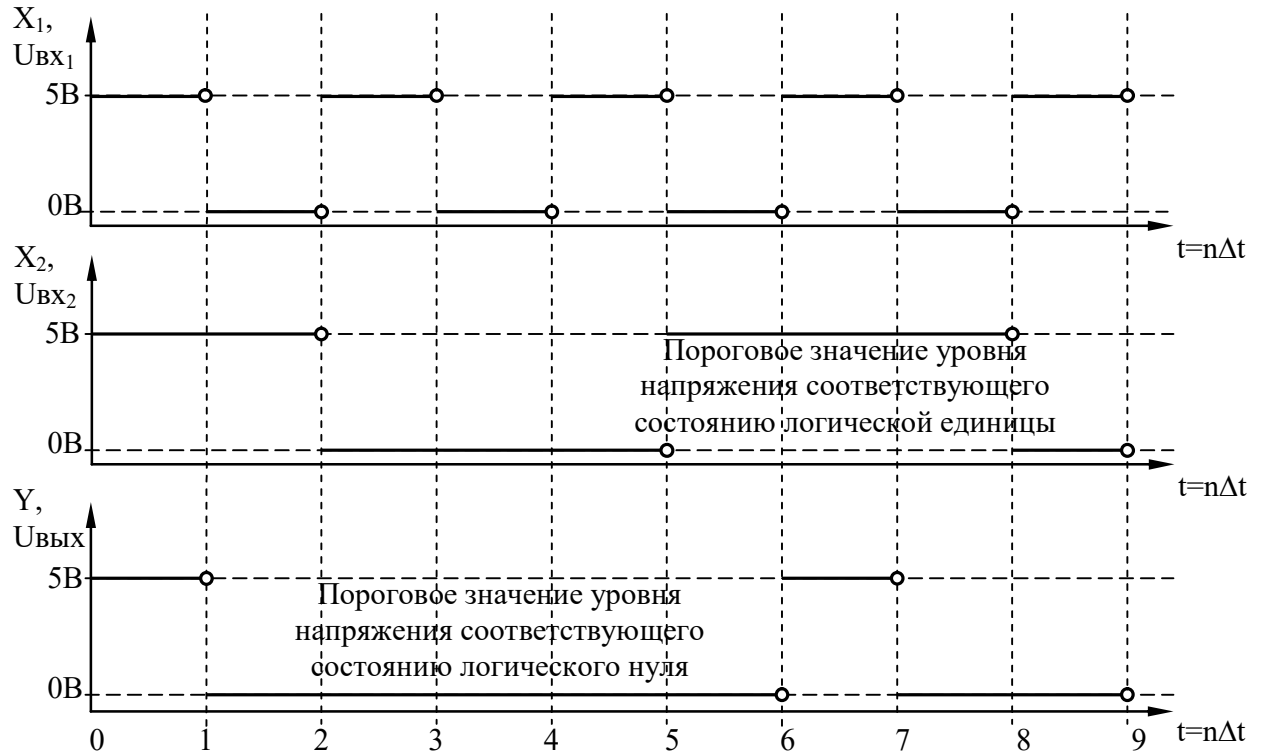


Рисунок 3.15. Временная диаграмма динамики изменения состояний на выходе конъюнктора в зависимости от уровней сигналов на его двух входах

Прокомментируем полученную временную диаграмму состояний конъюнкции:

- на первом и втором графиках задано изменение состояний на входах ($U_{\text{вх}_1}$, $U_{\text{вх}_2}$);
- на третьем графике $Y(X_1, X_2)$ пороговое значение напряжения соответствующее состоянию логической единицы (1) достигается на выходе ($U_{\text{вых}}$) только если достигнуто пороговое значение напряжения соответствующее логической единице одновременно на обоих входах ($U_{\text{вх}_1}$, $U_{\text{вх}_2}$) конъюнктора (первый и второй операнды – X_1 , X_2);
 - на интервале между нулевым и первым дискретными моментами времени ($t|_{n=0,1} = n\Delta t$) пороговое значение напряжения на обоих входах конъюнктора $X = \{X_1, X_2\}$ соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $Y = X_1 \& X_2$ достигается состояние логической единицы (кружок на каждом графике означает исключение точки в момент переключения логического устройства при $n=0, 1, 2, \dots, n$);
 - на интервале между первым и вторым дискретными моментами времени ($t|_{n=1,2} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе конъюнктора X_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $Y = X_1 \& X_2$ достигается состояние логического нуля;

- на интервале между вторым и третьим дискретными моментами времени ($t|_{n=2,3} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе конъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $Y = X_1 \& X_2$ достигается состояние логического нуля;
- на интервале между третьим и четвертым дискретными моментами времени ($t|_{n=3,4} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе конъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $Y = X_1 \& X_2$ достигается состояние логического нуля;
- на интервале между четвертым и пятым дискретными моментами времени ($t|_{n=4,5} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе конъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $Y = X_1 \& X_2$ достигается состояние логического нуля;
- на интервале между пятым и шестым дискретными моментами времени ($t|_{n=5,6} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе конъюнктора X_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $Y = X_1 \& X_2$ достигается состояние логического нуля;
- на интервале между шестым и седьмым дискретными моментами времени ($t|_{n=6,7} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе конъюнктора X_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $Y = X_1 \& X_2$ достигается состояние логической единицы;
- на интервале между седьмым и восьмым дискретными моментами времени ($t|_{n=7,8} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе конъюнктора X_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $Y = X_1 \& X_2$ достигается состояние логического нуля;
- на интервале между восьмым и девятым дискретными моментами времени ($t|_{n=7,8} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе конъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $Y = X_1 \& X_2$ достигается состояние логического нуля.

Таким образом, истинность результирующего значения операции логического умножения (И) достигается только в случае истинности значений двух операндов.

3.3.3. Операция дизъюнкции, функция логического сложения ИЛИ

Определение

Операцией дизъюнкции (ИЛИ) называют функцию логического сложения значений двух операндов, истинность результирующего значения которой достигается в случае истинности по крайней мере одного из двух значений логических операндов.

Эта логическая функция имеет несколько названий: логическое сложение, дизъюнкция, логическая связка (функция) ИЛИ, логическая операция V .

Аналитический способ представления

При использовании аналитической записи посредством формулы функция логического сложения (дизъюнкция, операция ИЛИ) изображается символом « V ».

Запись « $x_1 V x_2$ » читается как « x_1 ИЛИ x_2 ».

Табличный способ представления посредством таблицы истинности

Таблица истинности элемента дизъюнктор, операции дизъюнкция (ИЛИ), функции логического сложения двух значений логических операндов представлена в табл. 3.7.

Таблица 3.7

Таблица истинности операции дизъюнкция (ИЛИ) или функции логического сложения значений двух логических операндов

Значение первого логического операнда X_1	Значение второго логического операнда X_2	Значение логической функции $Y=X_1 V X_2$
0	0	0
0	1	1
1	0	1
1	1	1

Таблица истинности задает взаимно однозначное соответствие между бинарными значениями логических операндов и результирующим значением операции ИЛИ.

В таблице истинности представлено 4 двоичных комбинации значений логических операндов и соответствующие им результирующие значения логической функции.

Таблица истинности формируется посредством подстановки значений операндов в логическую функцию и расчета ее результирующего значения, которые заносятся в соответствующие информационные поля и образуют запись по строке.

Функциональная схема

На функциональной схеме логического устройства (функции) операция дизъюнкции двух значений логических операндов изображается посредством использования стандартизованного условно-графического обозначения, которое включает два входа (X_1 , X_2) для каждого логического операнда и один выход (Y) для регистрации возвращаемого (результирующего) значения логической операции ИЛИ (рис. 3.16).

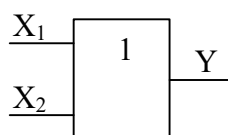


Рисунок 3.16. Условно-графическое обозначение логической операции дизъюнкция (ИЛИ)

При использовании указанного обозначения операции логического сложения значений двух логических операндов допускается сочетание других простых и сложных логических операций, что позволяет реализовать их каскадное подключение.

Электрическая принципиальная схема

Далее представлены простейшая электрическая принципиальная схема, содержащая аппаратную реализацию дизъюнкции значений двух операндов (рис. 3.17), а также таблица эпюров напряжений на входах и выходе дизъюнктора (табл. 3.8) и временная диаграмма динамики изменения состояний на выходе и входах (рис. 3.18).

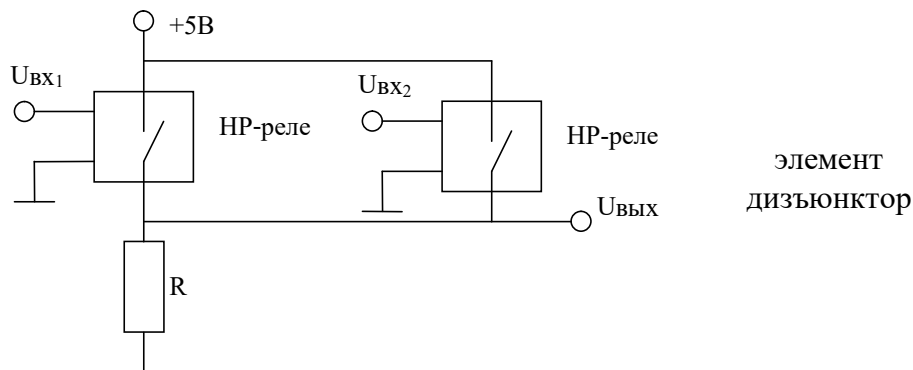


Рисунок 3.17. Электрическая принципиальная схема аппаратной реализации дизъюнктора, эпюры напряжений на его входах и выводах

На представленной электрической принципиальной схеме изображены:

- клемма +5В – обеспечивает подачу опорного напряжения номиналом 5В;
- клеммы $U_{вх1}$ – предназначены для подачи входного напряжения, обеспечивающего управление коммутацией нормально разомкнутого реле: подача напряжения 5В реализует замыкание контакта, снятие – размыкание контакта;
- два нормально разомкнутых (НР) реле – пропускают эффективный сигнал только при подаче опорного напряжения на управляющие входы $U_{вх1}$, $U_{вх2}$;
- клемма $U_{вых}$ – позволяет обеспечить регистрацию напряжения на выходе $U_{вых}$.

Регистрация эффективного сигнала на $U_{вых}$ возможна только в случае наличия опорного напряжения между клеммами +5В и ноль, а также при подаче управляющего напряжения хотя бы на одну из двух клемм ($U_{вх1}$ или $U_{вх2}$), что обеспечивает коммутацию контакта хотя-бы одного из представленных нормально-разомкнутого реле на схеме.

Далее представлена табл. 3.8, которая отражает соответствие между таблицей истинности дизъюнкции значений двух логических операндов и эпюрами напряжений на входах и выходе аппаратной реализации операции логического сложения ИЛИ.

Таблица 3.8

Таблица истинности и эпюры напряжений на входах и выходах дизъюнктора

X_1	X_2	$U_{вх1}$	$U_{вх2}$	$Y=X_1 \& X_2$	$U_{вых}$
0	0	0В	0В	0	0В
0	1	0В	5В	1	5В
1	0	5В	0В	1	5В
1	1	5В	5В	1	5В

Таблица истинности дизъюнктора задает взаимно однозначное соответствие между двоичной комбинацией на входе и установившимся значением на выходе: пороговое значение напряжения равное 5В соответствует состоянию логической единицы, пороговое значение напряжения равное 0В соответствует логическому нулю.

Временная диаграмма

Временная диаграмма состояний дизъюнктора (рис. 3.18) как аппаратной реализации операции дизъюнкции (ИЛИ), функции логического сложения двух значений логических операндов отражает динамику изменения уровней сигналов соответствующих двум логическим состояниям (0 и 1), которые достигаются на выходе ($U_{\text{вых}}$) под влиянием уровней напряжения заданным на каждом из двух входов ($U_{\text{вх1}}$ и $U_{\text{вх2}}$).

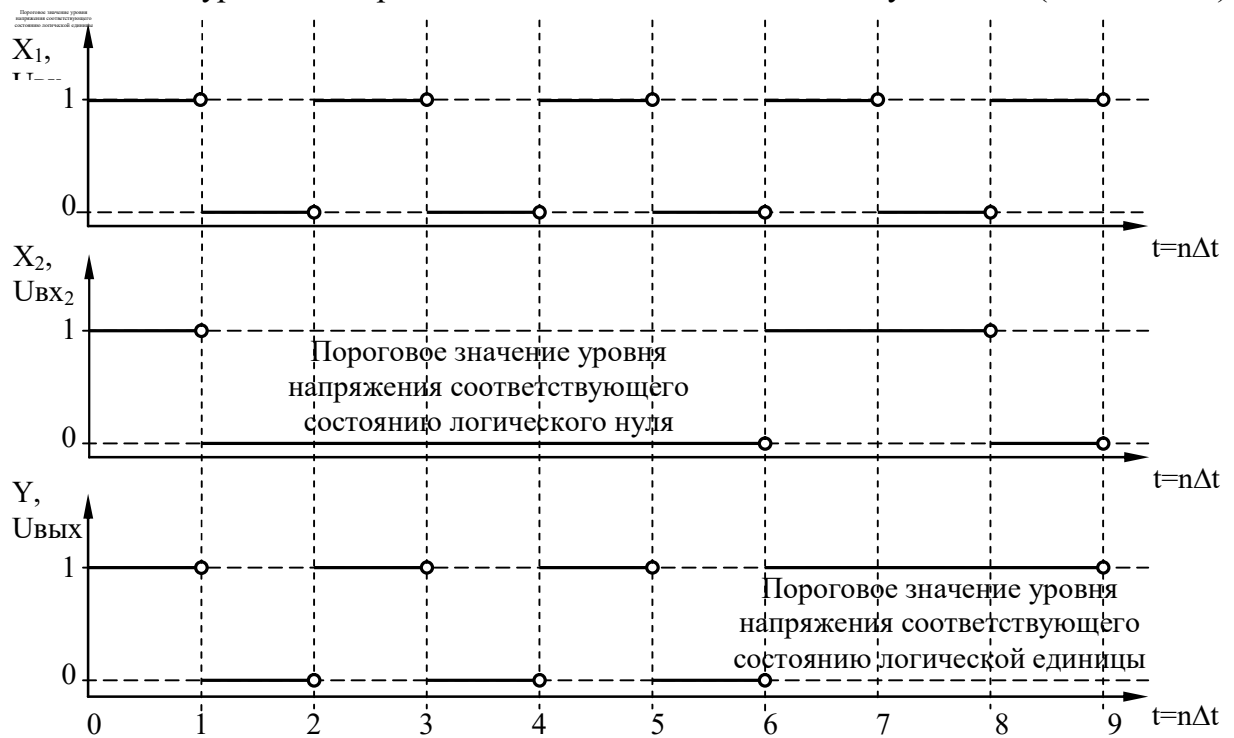


Рисунок 3.18. Временная диаграмма динамики изменения состояний на выходах дизъюнктора в зависимости от уровней сигналов на его входах

Прокомментируем полученную временную диаграмму состояний дизъюнкции:

- на первом и втором графиках задана динамика изменения состояний на входах (X_1, X_2);
- на третьем графике $Y(X_1, X_2)$ пороговое значение напряжения соответствующее состоянию логической единицы (1) достигается только в случае если достигнуто пороговое значение напряжения соответствующее логической единице по крайней мере на одном входе дизъюнктора (первый или второй операнд – X_1, X_2);
 - на интервале между нулевым и первым дискретными моментами времени ($t|_{n=0,1} = n\Delta t$) – пороговое значение напряжения на обоих входах дизъюнктора $X=\{X_1, X_2\}$ соответствует состоянию логической единицы (1), поэтому на единственном выходе операции логического сложения $Y= X_1 \vee X_2$ достигается состояние логической единицы (кружок на каждом графике означает исключение точки в момент переключения устройства при $n=0, 1, 2, \dots, n$);
 - на интервале между первым и вторым дискретными моментами времени ($t|_{n=1,2} = n\Delta t$) – пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y= X_1 \& X_2$ достигается состояние логического нуля;

- на интервале между вторым и третьим дискретными моментами времени ($t|_{n=2,3} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y = X_1 \vee X_2$ достигается состояние логической единицы;
- на интервале между третьим и четвертым дискретными моментами времени ($t|_{n=3,4} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y = X_1 \vee X_2$ достигается состояние логического нуля;
- на интервале между четвертым и пятым дискретными моментами времени ($t|_{n=4,5} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y = X_1 \vee X_2$ достигается состояние логической единицы;
- на интервале между пятым и шестым дискретными моментами времени ($t|_{n=5,6} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y = X_1 \vee X_2$ достигается состояние логического нуля;
- на интервале между шестым и седьмым дискретными моментами времени ($t|_{n=6,7} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе дизъюнктора X_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического сложения двух значений логических операндов $Y = X_1 \vee X_2$ достигается состояние логической единицы;
- на интервале между седьмым и восьмым дискретными моментами времени ($t|_{n=7,8} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе дизъюнктора X_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического сложения $Y = X_1 \vee X_2$ достигается состояние логической единицы;
- на интервале между восьмым и девятым дискретными моментами времени ($t|_{n=7,8} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y = X_1 \vee X_2$ достигается состояние логической единицы.

Таким образом, истинность результирующего значения операции логического умножения достигается только в случае истинности сразу двух значений операндов.

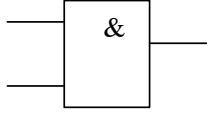
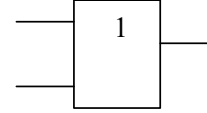
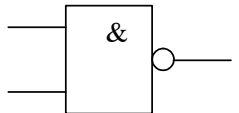
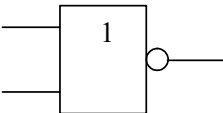
3.3.4. Гибридные логические операции И-НЕ и ИЛИ-НЕ

Все рассмотренные простые логические операции позволяют реализовать любую сложную или гибридную логическую операцию или функцию алгебры логики.

Существуют основные (инверсия, конъюнкция и дизъюнкция) и гибридные логические операции и функции, поэтому вводятся соответствующие дополнительные логические операции: конъюнкция и дизъюнкция инверсные по выходу (табл. 3.9).

Таблица 3.9

Условно-графические обозначения простых и гибридных логических операций И, ИЛИ, НЕ, И-НЕ, ИЛИ-НЕ

Логическая операция	Конъюнкция (И)	Дизъюнкция (ИЛИ)
Условно-графическое обозначение простой логической операции	 И	 ИЛИ
Условно-графическое обозначение гибридной логической операции: конъюнкция (И) и дизъюнкция (ИЛИ) инверсные по выходу	 И-НЕ	 ИЛИ-НЕ

Интегральные микросхемы характеризуются видом и степенью интеграции, а также служат аппаратной реализацией простых и сложных логических функций.

Уровень (степень) интеграции определяет плотность расположения элементов (компонентов), которые исполнены в объеме полупроводникового кристалла микросхемы:

- низкий – небольшое количество элементов соответствует микросхемам логики;
- средний – среднее количество электротехнических элементов в объеме полупроводникового кристалла, что характерно для реализации микросхем памяти;
- высокий – большая плотность распределения элементов для процессоров.

Пример 3.1. Представить логические элементы в основе микросхемы низкого уровня интеграции К155 ЛА3, которые реализуют определенные логические операции.

Микросхема К155 ЛА3 (К555 ЛА3) содержит 4 элемента И-НЕ (рис. 3.19).

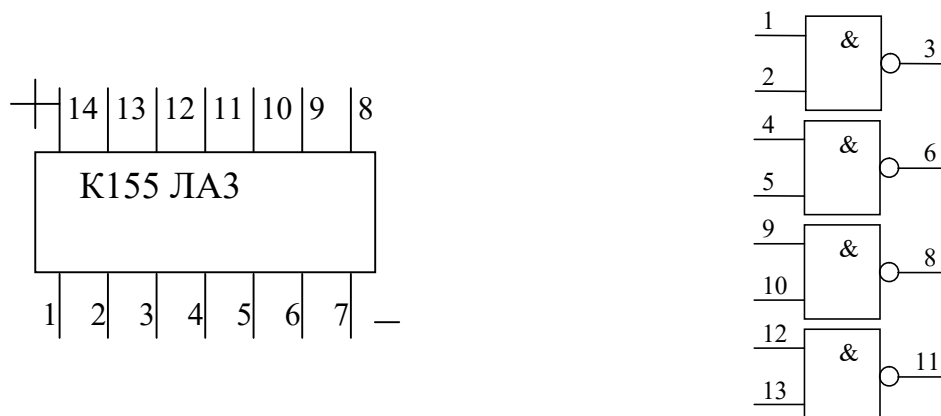


Рисунок 3.19. Топологическая и функциональная схема микросхемы К155 ЛА3

Рекомендуемая последовательность проектирования и сборки функциональных логических схем и электрических принципиальных схем для аппаратной реализации различных логических элементов и устройств представляется следующим образом:

- первый этап – определить количество входов (n) и выходов (m) логического устройства по аналитическому, табличному или графическому представлению (временная диаграмма состояний, функциональная логическая или электрическая принципиальная схема), что соответствует числу операндов на входе и достигнутых результирующих значений логической функции на выходе, а затем записать систему логических уравнений или функций алгебры логики;
- второй этап – если имеется аналитическое представление определенной логической функции или логического устройства, то необходимо максимально упростить имеющуюся формулу (логическое уравнение) посредством использования правил эквивалентного преобразования (законов алгебры логики);
- третий этап – на основе аналитического представления логического устройства посредством использования формулы требуется построить таблицу истинности, которая задает взаимно однозначное соответствие между достигнутыми результирующими значениями функции (выходы $Y = \{y_1, y_2, \dots, y_m\}$) и заданными значениями логических операндов (входы $X = \{x_1, x_2, \dots, x_n\}$);
- четвертый этап – на основе сформированной таблицы истинности необходимо разработать функциональную схему заданного логического устройства;
- пятый этап – построить временную диаграмму состояний на входах и выходах логического устройства на основе имеющейся функциональной схемы;
- шестой этап – необходимо осуществить разработку электрической принципиальной схемы на основе имеющейся функциональной схемы, временной диаграммы, таблицы истинности проектируемого логического устройства;
- седьмой этап – обеспечить аппаратную реализацию логического устройства на основе имеющейся функциональной схемы и имеющейся элементной базы;
- восьмой этап – необходимо провести стендовые испытания разработанного логического устройства для построения экспериментальных таблиц истинности, временных диаграмм состояний достигнутых на выходах в зависимости от заданных на входы логического устройства двоичных комбинаций;
- девятый этап – проверить соответствие между теоретическими (заданными) и экспериментальными таблицами истинности, временными диаграммами состояний с целью выявления имеющихся несоответствий в принципах функционирования разработанного логического устройства (статика и динамика);
- десятый этап – необходимо устранить выявленные несоответствия и ошибки между теоретическими и экспериментальными таблицами истинности, временными диаграммами состояний входов и выходов, а затем усовершенствовать аналитическое представление, таблицу истинности, функциональную и электрическую принципиальную схемы разработанного логического устройства.

Пример 3.2. Спроектировать функциональную схему логического устройства, которое осуществляет процедуру арифметического сложения двух одноразрядных двоичных чисел (BIN) и задается следующей системой логических уравнений:

$$\begin{cases} P = x_1 \& x_2 \\ S = (x_1 \vee x_2) \& (HE(x_1) \vee HE(x_2)) = (x_1 \vee x_2) \& HE(x_1 \& x_2) = (x_1 \vee x_2) \& HE(P) \end{cases}$$

Для целей эквивалентного преобразования с целью упрощения исходного логического выражения было использовано правило Де-Моргана (Шеннона).

Динамика функционирования логического устройства описывается следующей таблицей истинности с позиций булевой алгебры и двоичной арифметики (табл. 3.10).

Таблица 3.10

Таблица истинности полусумматора

Значение первого слагаемого (X ₁)	Значение второго слагаемого (X ₂)	Результат конъюнкции Y=(X ₁ ∨ X ₂) в DEC	Результат арифметического сложения в BIN	
			P	S
0	0	0	0	0
0	1	1	0	1
1	0	1	0	1
1	1	2	1	0

Вычислительная процедура отражается в виде определенной схемы (рис. 3.20).

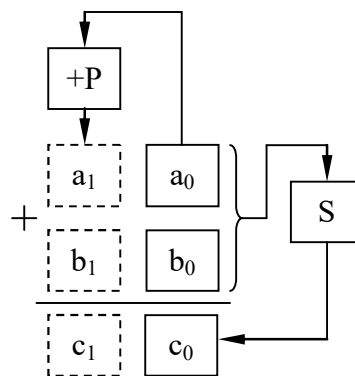


Рисунок 3.20. Вычислительная процедура расчета номинала младшего разряда и номинала переноса в старший разряд при суммировании одноразрядных чисел

Согласно представленной структурной схеме необходимо осуществить операцию сложения двух разрядов (a₀ и b₀) заданных чисел A={a_{n-1}, ..., a_i, ..., a₁, a₀} и B={b_{n-1}, ..., b_i, ..., b₁, b₀} с целью формирования результирующего номинального значения (S) в младшем разряде (c₀) и номинала для переноса (+P) в старший разряд (c₁).

Таким образом, для реализации алгоритма вычисления необходимо иметь:

- исходное номинальное значение нулевого разряда (a₀) в первом слагаемом;
- исходное номинальное значение нулевого разряда (b₀) во втором слагаемом;
- номинальное значение (S) для формирования младшего разряда (c₀) в сумме;
- номинальное значение (P) для переноса в старший разряд (c₁) результата.

Согласно аналитическому представлению логического устройства в виде заданной системы логических уравнений и таблицы истинности (табл. 3.10) проектируется его функциональная схема, которая включает ряд условно-графических обозначений логических операций, которые обеспечивают формирование результата:

- в верхней части – две конъюнкции $\&$, одна дизъюнкция 1, одна инверсия;
- в нижней части – полусумматор SH как эквивалентное представление.

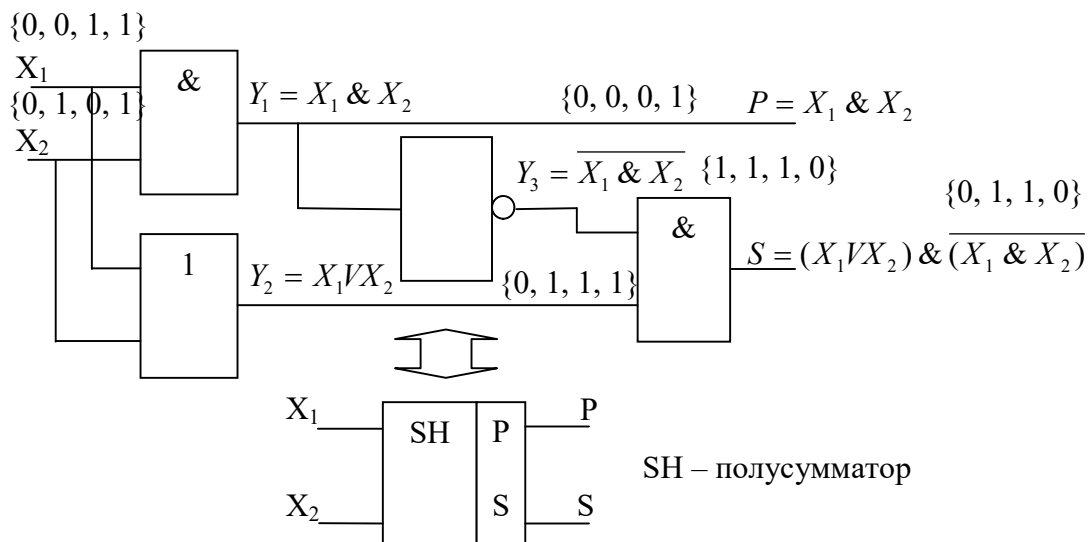


Рисунок 3.21. Функциональная схема полусумматора и его условно-графическое обозначение

Рассмотрим полученную функциональную схему логического устройства и зададим несколько значений двух логических операндов $X_1 = \{0, 0, 1, 1\}$, $X_2 = \{0, 1, 0, 1\}$:

- первый конъюнктор ($\&$) на представленной схеме имеет два входа и выход:
 - на первый вход последовательно задаются логические состояния согласно четырем значениям первого логического операнда ($X_1 = \{0, 0, 1, 1\}$);
 - на второй вход последовательно задаются логические состояния согласно четырем значениям второго логического операнда ($X_2 = \{0, 1, 0, 1\}$);
 - на выходе конъюнктора Y_1 последовательно достигаются следующие логические состояния $Y_1 = \{0, 0, 0, 1\}$ после заданных воздействий X_1, X_2 ;
- первый дизъюнктор (1) на представленной схеме имеет два входа и выход:
 - на первый вход последовательно задаются логические состояния согласно четырем значениям первого логического операнда ($X_1 = \{0, 0, 1, 1\}$);
 - на второй вход последовательно задаются логические состояния согласно четырем значениям второго логического операнда ($X_2 = \{0, 1, 0, 1\}$);
 - на выходе дизъюнктора Y_2 последовательно достигаются следующие логические состояния $Y_2 = \{0, 1, 1, 1\}$ после заданных воздействий X_1, X_2 ;
- инвертор на представленной схеме устройства имеет один вход и один выход:
 - на вход последовательно задаются логические состояния после операции конъюнкции значений первого (X_1) и второго (X_2) логических операндов;
 - на выходе Y_3 достигаются логические состояния $Y_3 = \overline{Y_1} = \overline{X_1 \& X_2} = \{1, 1, 1, 0\}$;

- второй конъюнктор (&) на представленной схеме имеет два входа и один выход:
 - на первый вход последовательно задаются логические состояния достигнутые в результате инверсии результирующих значений конъюнкции первого (X_1) и второго (X_2) логических операндов ($Y_3 = \overline{X_1 \& X_2} = \{1, 1, 1, 0\}$);
 - на второй вход последовательно задаются логические состояния достигнутые в результате дизъюнкции первого (X_1) и второго (X_2) операндов $\{0, 1, 1, 1\}$;
 - на выходе S последовательно достигаются логические состояния, которые достигнуты в результате конъюнкции первого (инвертированные значения конъюнкции первого X_1 и второго X_2 операндов) и второго множителей (результирующие значения дизъюнкции первого X_1 и второго X_2 операндов) $S = \overline{X_1 \& X_2} \& (X_1 \vee X_2) = \{0, 1, 1, 0\}$;
- в результате подачи входных воздействий (операнды $X_1 = \{0, 0, 1, 1\}$, $X_2 = \{0, 1, 0, 1\}$) на два входа представленного логического устройства на выходах достигаются:
 - на первом выходе – номинал для переноса в старший разряд ($P = \{0, 0, 0, 1\}$);
 - на втором выходе – номинальное значение младшего разряда ($S = \{0, 1, 1, 0\}$).

Определение

Полусумматор – это логическое устройство, которое формирует номинальное значение цифры (S) в младшем разряде (c_0) и номинал для переноса (P) в старший разряд (c_1) при арифметическом сложении двух двоичных одноразрядных чисел.

Выполним упрощение второго логического выражения и осуществим его подстановку в первое логическое выражение в заданной системе логических уравнений.

Исходная система логических уравнений представлена в следующем виде:

$$\begin{cases} P = x_1 \& x_2 \\ S = (x_1 \vee x_2) \& (HE(x_1) \vee HE(x_2)) = (x_1 \vee x_2) \& HE(x_1 \& x_2) = (x_1 \vee x_2) \& HE(P) \end{cases}$$

После подстановки значений первого $X_1 = \{0, 0, 1, 1\}$ и второго $X_2 = \{0, 1, 0, 1\}$ логических операндов в полученную систему логических уравнений, получим:

$$\begin{cases} P|_{\substack{X_1=\{0,0,1,1\} \\ X_2=\{0,1,0,1\}}} = \begin{cases} 0 \& 0 = 0 \\ 0 \& 1 = 0 \\ 1 \& 0 = 0 \\ 1 \& 1 = 1 \end{cases} \\ S|_{\substack{X_1=\{0,0,1,1\} \\ X_2=\{0,1,0,1\}}} = \begin{cases} (0 \vee 0) \& HE(0 \& 0) = (0 \vee 0) \& HE(0) = 0 \\ (0 \vee 1) \& HE(0 \& 1) = (0 \vee 1) \& HE(0) = 1 \\ (1 \vee 0) \& HE(1 \& 0) = (1 \vee 0) \& HE(0) = 1 \\ (1 \vee 1) \& HE(1 \& 1) = (1 \vee 1) \& HE(1) = 0 \end{cases} \end{cases}$$

Таким образом, сформированы результирующие значения номиналов в младшем разряде и номинальные значения для переноса в старший разряд полученного числа:

$$\begin{cases} P|_{\substack{X_1=\{0,0,1,1\} \\ X_2=\{0,1,0,1\}}} = 0,0,0,1 & \text{где P – номинал для переноса в старший разряд;} \\ S|_{\substack{X_1=\{0,0,1,1\} \\ X_2=\{0,1,0,1\}}} = 0,1,1,0 & \text{S – номинальное значение цифры в младшем разряде.} \end{cases}$$

Представленная функциональная схема логического устройства (рис. 3.21, сверху) широко используется на практике и называется полусумматором, который имеет определенное условно-графическое обозначение «полусумматор» (рис. 3.21, внизу).

Временная диаграмма динамики состояний на выходах полусумматора под влиянием изменения уровней напряжения на его входах представлена на рис. 3.22.

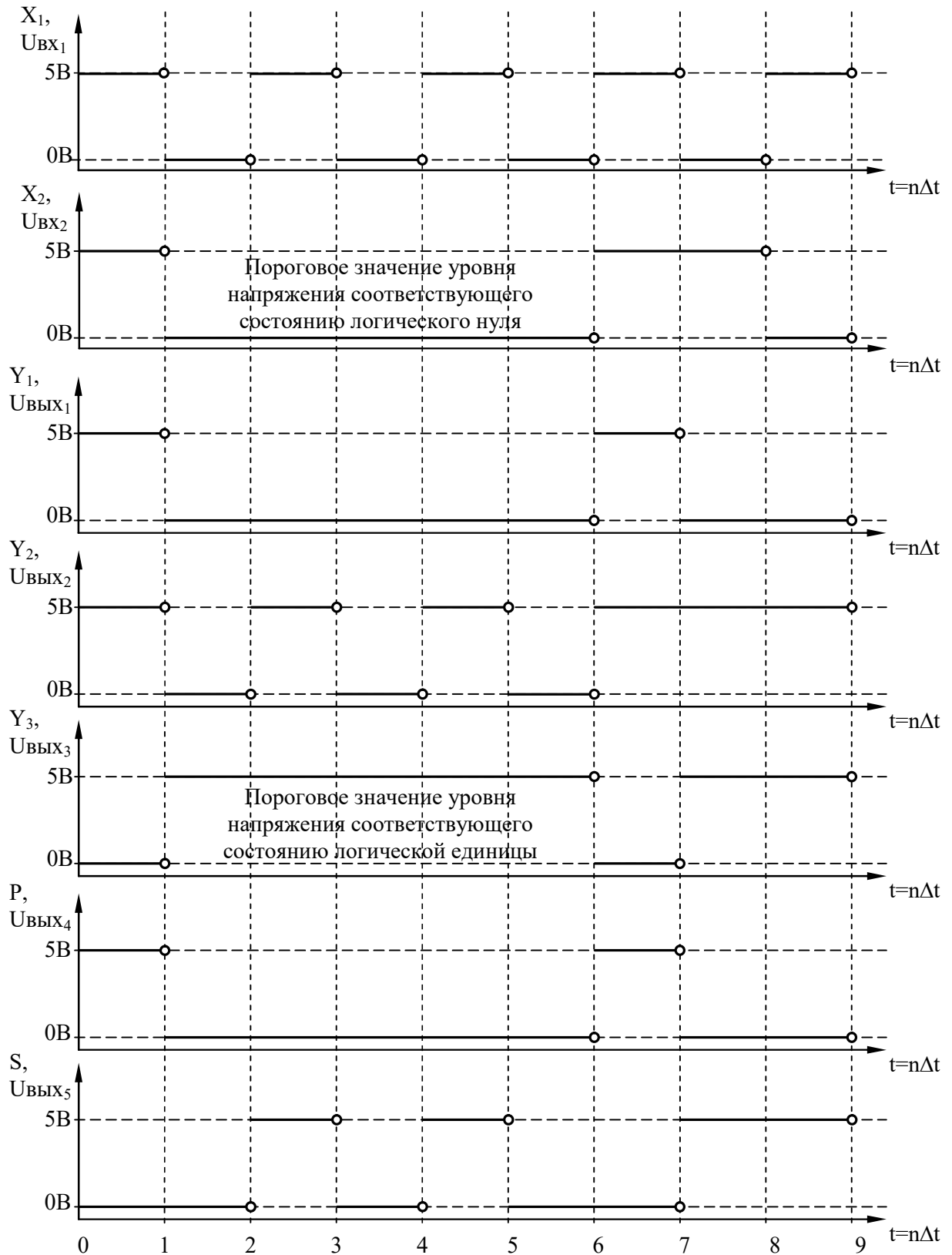


Рисунок 3.22. Временная диаграмма состояний полусумматора

Прокомментируем полученную временную диаграмму состояний дизъюнкции:

- на первом графике изображена динамика изменения состояний уровней напряжения на первом входе ($U_{вх1}$) представленного логического устройства (рис. 3.21), которые соответствуют значениям первого логического операнда (X_1);
- на втором графике изображена динамика изменения состояний уровней напряжения на втором входе ($U_{вх2}$) представленного логического устройства (рис. 3.21), которые соответствуют значениям второго логического операнда (X_2);
- на третьем графике изображены уровни напряжения на выходе ($U_{вых1}$) первого конъюнктора, которые соответствуют результату конъюнкции значений первого логического операнда (X_1) и второго логического операнда (X_2);
 - на интервале между нулевым и первым дискретными моментами времени ($t|_{n=0,1} = n\Delta t$) пороговое значение напряжения на обоих входах конъюнктора $X = \{X_1, X_2\}$ соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $Y_1 = X_1 \& X_2$ достигается состояние логической единицы (кружок на каждом графике означает исключение точки в момент переключения устройства при $n=0, 1, 2, \dots, n$);
 - на интервале между первым и вторым дискретными моментами времени ($t|_{n=1,2} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе конъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $Y_1 = X_1 \& X_2$ достигается состояние логического нуля;
 - на интервале между вторым и третьим дискретными моментами времени ($t|_{n=2,3} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе конъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $Y_1 = X_1 \& X_2$ достигается состояние логического нуля;
 - на интервале между третьим и четвертым дискретными моментами времени ($t|_{n=3,4} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе конъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $Y_1 = X_1 \& X_2$ достигается состояние логического нуля;
 - на интервале между четвертым и пятым дискретными моментами времени ($t|_{n=4,5} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе конъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $Y_1 = X_1 \& X_2$ достигается состояние логического нуля;
 - на интервале между пятым и шестым дискретными моментами времени ($t|_{n=5,6} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1

- соответствует состоянию логического нуля (0) и на втором входе конъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $Y_1 = X_1 \& X_2$ достигается состояние логического нуля;
- на интервале между шестым и седьмым дискретными моментами времени ($t|_{n=6,7} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе конъюнктора X_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $Y_1 = X_1 \& X_2$ достигается состояние логической единицы;
 - на интервале между седьмым и восьмым дискретными моментами времени ($t|_{n=7,8} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе конъюнктора X_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $Y_1 = X_1 \& X_2$ достигается состояние логического нуля;
 - на интервале между восьмым и девятым дискретными моментами времени ($t|_{n=8,9} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе конъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $Y_1 = X_1 \& X_2$ достигается состояние логического нуля.
- на четвертом графике изображены уровни напряжения на выходе ($U_{\text{вых}2}$) первого дизъюнктора, которые соответствуют результату дизъюнкции значений первого логического операнда (X_1) и второго логического операнда (X_2);
 - на интервале между нулевым и первым дискретными моментами времени ($t|_{n=0,1} = n\Delta t$) пороговое значение напряжения на обоих входах дизъюнктора $X = \{X_1, X_2\}$ соответствует состоянию логической единицы (1), поэтому на выходе операции логического сложения $Y_2 = X_1 \vee X_2$ достигается состояние логической единицы (кружок на каждом графике означает исключение точки в момент переключения устройства при $n=0, 1, 2, \dots, n$);
 - на интервале между первым и вторым дискретными моментами времени ($t|_{n=1,2} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y_2 = X_1 \vee X_2$ достигается состояние логического нуля;
 - на интервале между вторым и третьим дискретными моментами времени ($t|_{n=2,3} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения двух значений логических операндов $Y_2 = X_1 \vee X_2$ достигается состояние логической единицы (1);

- на интервале между третьим и четвертым дискретными моментами времени ($t|_{n=3,4} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y_2 = X_1 \vee X_2$ достигается состояние логического нуля;
- на интервале между четвертым и пятым дискретными моментами времени ($t|_{n=4,5} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y_2 = X_1 \vee X_2$ достигается состояние логической единицы;
- на интервале между пятым и шестым дискретными моментами времени ($t|_{n=5,6} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y_2 = X_1 \vee X_2$ достигается состояние логического нуля;
- на интервале между шестым и седьмым дискретными моментами времени ($t|_{n=6,7} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе дизъюнктора X_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического сложения $Y_2 = X_1 \vee X_2$ достигается состояние логической единицы;
- на интервале между седьмым и восьмым дискретными моментами времени ($t|_{n=7,8} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логического нуля (0) и на втором входе дизъюнктора X_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического сложения $Y_2 = X_1 \vee X_2$ достигается состояние логической единицы;
- на интервале между восьмым и девятым дискретными моментами времени ($t|_{n=8,9} = n\Delta t$) пороговое значение напряжения на первом входе дизъюнктора X_1 соответствует состоянию логической единицы (1) и на втором входе дизъюнктора X_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического сложения $Y_2 = X_1 \vee X_2$ достигается состояние логической единицы.
- на пятом графике изображены уровни напряжения на выходе ($Y_3, U_{\text{вых3}}$) инвертора, которые соответствуют инвертированному результату конъюнкции значений первого логического операнда (X_1) и второго логического операнда (X_2);
 - на интервале между нулевым и первым дискретными моментами времени ($t|_{n=0,1} = n\Delta t$) пороговое значение напряжения на входе инвертора соответствует состоянию логической единицы (1), при этом на выходе инвертора достигается взаимно инверсное состояние логического нуля (кружок на графике означает исключение точки в момент переключения устройства при $n=0, 1$);

- на интервале между первым и вторым дискретными моментами времени ($t|_{n=1,2} = n\Delta t$) пороговое значение напряжения на входе инвертора соответствует состоянию логического нуля (0), при этом на выходе инвертора сразу достигается взаимно инверсное состояние логической единицы (1);
- на интервале между вторым и третьим дискретными моментами времени ($t|_{n=2,3} = n\Delta t$) пороговое значение напряжения на входе инвертора соответствует состоянию логического нуля (0), при этом на выходе инвертора сразу достигается взаимно инверсное состояние логической единицы (1);
- на интервале между третьим и четвертым дискретными моментами времени ($t|_{n=3,4} = n\Delta t$) пороговое значение напряжения на входе инвертора соответствует состоянию логического нуля (0), при этом на выходе инвертора сразу достигается взаимно инверсное состояние логической единицы (1);
- на интервале между четвертым и пятым дискретными моментами времени ($t|_{n=4,5} = n\Delta t$) пороговое значение напряжения на входе инвертора соответствует состоянию логического нуля (0), при этом на выходе инвертора сразу достигается взаимно инверсное состояние логической единицы (1);
- на интервале между пятым и шестым дискретными моментами времени ($t|_{n=5,6} = n\Delta t$) пороговое значение напряжения на входе инвертора соответствует состоянию логического нуля (0), при этом на выходе инвертора сразу достигается взаимно инверсное состояние логической единицы (1);
- на интервале между шестым и седьмым дискретными моментами времени ($t|_{n=6,7} = n\Delta t$) пороговое значение напряжения на входе инвертора соответствует состоянию логической единицы (1), при этом на выходе инвертора сразу достигается взаимно инверсное состояние логического нуля (0);
- на интервале между седьмым и восьмым дискретными моментами времени ($t|_{n=7,8} = n\Delta t$) пороговое значение напряжения на входе инвертора соответствует состоянию логического нуля (0), при этом на выходе инвертора сразу достигается взаимно инверсное состояние логической единицы (1);
- на интервале между восьмым и девятым дискретными моментами времени ($t|_{n=8,9} = n\Delta t$) пороговое значение напряжения на входе инвертора соответствует состоянию логического нуля (0), при этом на выходе инвертора сразу достигается взаимно инверсное состояние логической единицы (1).
- на шестом графике изображены уровни напряжения на выходе ($Y_1=P$) первого конъюнктора, которые непосредственно соответствуют результату конъюнкции значений первого логического операнда (X_1) и второго логического операнда (X_2), что характеризует достигнутое номинальное значение (P) подлежащее переносу в старший разряд (c_1) полученного в результате числа C;

- на седьмом графике изображены уровни напряжения на выходе (S) второго конъюнктора, которые соответствуют результату конъюнкции значений первого логического операнда (Y_3 – результат дизъюнкции первого логического операнда и второго логического операнда) и второго логического операнда (Y_2 – результат инверсии конъюнкции первого и второго логических операндов), что соответствует номиналу цифры в младшем разряде (c_0) полученного числа С;
 - на интервале между нулевым и первым дискретными моментами времени ($t|_{n=0,1} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора Y_3 соответствует состоянию логического нуля (0) и на втором входе конъюнктора Y_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $S=Y_3\&Y_2$ достигается состояние логического нуля (кружок на каждом графике означает исключение точки в момент переключения устройства при $n=0, 1, 2, \dots, n$);
 - на интервале между первым и вторым дискретными моментами времени ($t|_{n=1,2} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора Y_3 соответствует состоянию логической единицы (1) и на втором входе конъюнктора Y_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $S=Y_3\&Y_2$ достигается состояние логического нуля;
 - на интервале между вторым и третьим дискретными моментами времени ($t|_{n=2,3} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора Y_3 соответствует состоянию логической единицы (1) и на втором входе конъюнктора Y_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $S=Y_3\&Y_2$ достигается состояние логической единицы;
 - на интервале между третьим и четвертым дискретными моментами времени ($t|_{n=3,4} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора Y_3 соответствует состоянию логической единицы (1) и на втором входе конъюнктора Y_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $S=Y_3\&Y_2$ достигается состояние логического нуля;
 - на интервале между четвертым и пятым дискретными моментами времени ($t|_{n=4,5} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора Y_3 соответствует состоянию логической единицы (1) и на втором входе конъюнктора Y_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $S=Y_3\&Y_2$ достигается состояние логической единицы;
 - на интервале между пятым и шестым дискретными моментами времени ($t|_{n=5,6} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора Y_3 соответствует состоянию логической единицы (1) и на втором входе конъюнктора Y_2 соответствует состоянию логического нуля (0), поэтому на выходе операции логического умножения $S=Y_3\&Y_2$ достигается состояние логического нуля;

- на интервале между шестым и седьмым дискретными моментами времени ($t|_{n=6,7} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора Y_3 соответствует состоянию логического нуля (0) и на втором входе конъюнктора Y_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $S=Y_3 \& Y_2$ достигается состояние логического нуля;
- на интервале между седьмым и восьмым дискретными моментами времени ($t|_{n=7,8} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора Y_3 соответствует состоянию логической единицы (1) и на втором входе конъюнктора Y_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $S=Y_3 \& Y_2$ достигается состояние логической единицы;
- на интервале между восьмым и девятым дискретными моментами времени ($t|_{n=8,9} = n\Delta t$) пороговое значение напряжения на первом входе конъюнктора Y_3 соответствует состоянию логической единицы (1) и на втором входе конъюнктора Y_2 соответствует состоянию логической единицы (1), поэтому на выходе операции логического умножения $S=Y_3 \& Y_2$ достигается состояние логической единицы.

Таким образом, в результате вычислительной процедуры формируются:

- номинальное значение цифры (S) в младшем разряде результирующего числа (c_0);
- номинальное значение (P) для переноса в старший разряд (c_1) итогового числа.

Аппаратная реализация арифметической операции полного поразрядного суммирования двух двоичных чисел обеспечивается посредством реализации каскадного включения последовательности полусумматоров на функциональной схеме (рис. 3.23).

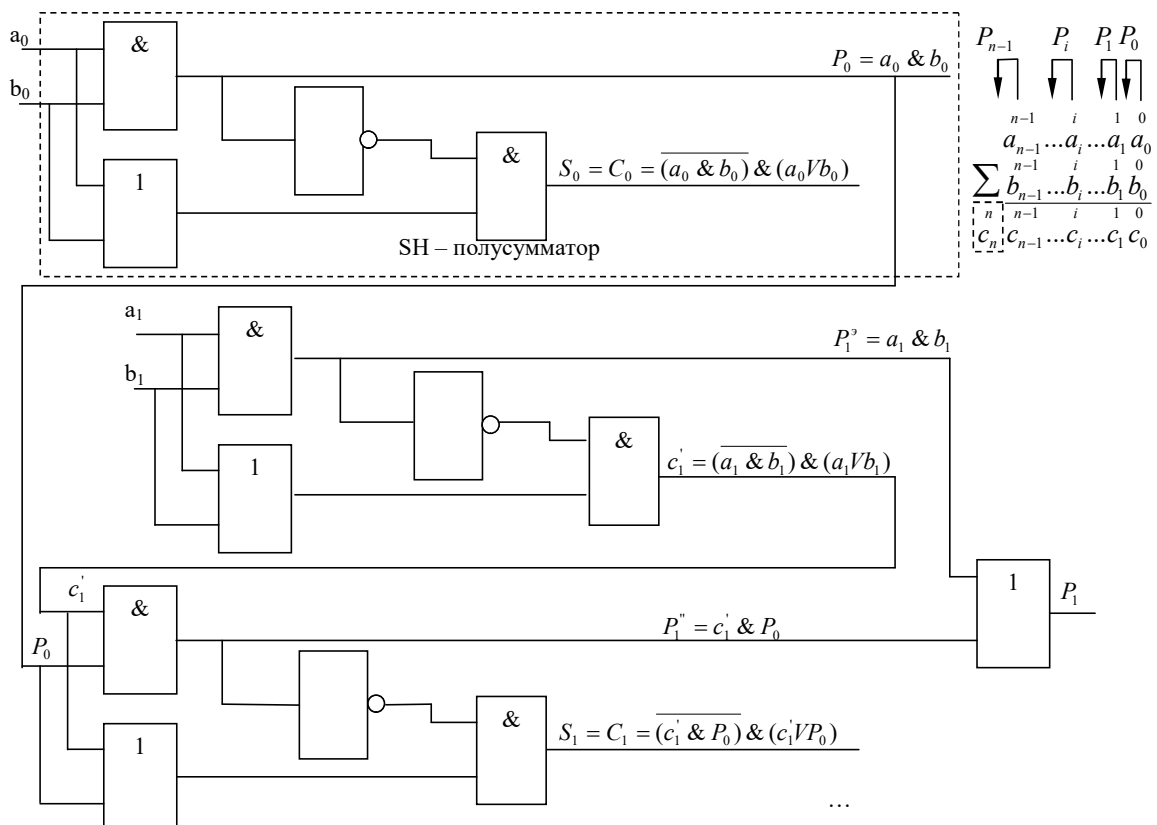


Рисунок 3.23. Функциональная схема полного сумматора на основе полусумматора

3.4. Основные законы булевой алгебры

Эволюция теории информации и появление новаций в области информационных технологий обусловили стремительное развитие алгоритмического обеспечения, которое инициировало повсеместное использование алгебры логики, выработку методов и создание технологий проектирования функциональных схем логических устройств различного уровня сложности для разных предметных областей:

В процессе формирования, преобразования и использования различных логических функций используются законы алгебры логики (Булевой алгебры логики).

Определение

Законы алгебры логики служат правилами эквивалентного преобразования логических выражений от сложного к простому в процессе их итеративного упрощения.

1. Переместительный (коммутативный) закон алгебры логики:

- для дизъюнкции, операции логического сложения значений двух операндов: $x_1 \vee x_2 = x_2 \vee x_1$ – перестановка значений операндов x_1 и x_2 при логическом сложении не влияет на номинальное значение полученного результата;
- для конъюнкции, операции логического умножения значений двух операндов: $x_1 \& x_2 = x_2 \& x_1$ – перестановка значений операндов x_1 и x_2 при логическом умножении не оказывает никакого влияния на номинальное значение результата.

2. Сочетательный (ассоциативный) закон булевой алгебры:

- для дизъюнкции значений двух операндов: $x_1 \vee (x_2 \vee x_3) = (x_1 \vee x_2) \vee x_3$ – при сохранении знака логического сложения значений различных операндов скобки позволяют изменить последовательность расчета конечного результата;
- для конъюнкции значений двух операндов: $x_1 \& (x_2 \& x_3) = (x_1 \& x_2) \& x_3$ – при сохранении знака логического умножения значений различных операндов скобки позволяют изменить последовательность расчета итогового результата.

3. Распределительный (дистрибутивный) закон алгебры логики:

- для операции логического умножения (И): $x_1 \& (x_2 \vee x_3) = x_1 \& x_2 \vee x_1 \& x_3$ – конъюнкция операнда x_1 с результатом дизъюнкции двух значений операндов x_2 и x_3 эквивалентна дизъюнкции результата двух независимых конъюнкций пар значений двух логических операндов $\{x_1 \text{ и } x_2\}$ и $\{x_1 \text{ и } x_3\}$;
- для операции логического сложения (ИЛИ): $x_1 \vee (x_2 \& x_3) = (x_1 \vee x_2) \& (x_1 \vee x_3)$ – дизъюнкция операнда x_1 с результатом конъюнкции двух значений операндов x_2 и x_3 эквивалентна конъюнкции результата двух независимых дизъюнкций пар значений двух логических операндов $\{x_1 \text{ и } x_2\}$ и $\{x_1 \text{ и } x_3\}$.

В процессе доказательства эквивалентности двух логических выражений с операндами, которые расположены слева и справа относительно скобок допустимо:

- подстановка значений логических операндов и формирование соответствующих таблиц истинности для дальнейшего сопоставления полученных результатов;
- создание функциональных схем определенных логических устройств и сопоставление временных диаграмм состояний выходов относительно входов.

4. Закон нулевого множества с нейтральным элементом по логическому сложению:
- для дизъюнкции, операции логического сложения значений двух операндов: $x_1 \vee 0 = x_1$ – ноль выступает нейтральным элементом по логическому сложению;
 - для конъюнкции, операции логического умножения значений 2х операндов: $x_1 \& 0 = 0$ – ноль выступает аннулирующим элементом по логическому умножению.
5. Закон универсального множества с нейтральным элементом по умножению:
- для дизъюнкции значений двух логических операндов: $x_1 \vee 1 = 1$ – единичное значение одного из логических операндов при логическом сложении позволяет говорить об истинности результирующего значения всего логического выражения независимо от заданного значения другого логического операнда;
 - для конъюнкции значений двух операндов: $x_1 \& 1 = x_1$ – единица выступает нейтральным элементом при логическом умножении значений двух операндов.
6. Закон повторения:
- при выполнении операции логического умножения (И) повторяющегося операнда: $x_1 \& x_1 = x_1$ – результирующее значение конъюнкции двух логических операндов при равенстве их значений равно значению логического операнда;
 - при выполнении операции логического сложения (ИЛИ) повторяющегося операнда: $x_1 \vee x_1 = x_1$ – результирующее значение дизъюнкции двух логических операндов при равенстве их значений равно значению логического операнда.
- Следствием закона повторения является закон исключения третьего операнда при осуществлении процедуры упрощения исходного сложного логического выражения:
- для конъюнкции значений двух или более операндов: $x_1 \& x_1 \& \dots \& x_1 = x_1$ – результирующее значение конъюнкции двух или более логических операндов при равенстве их значений равно значению логического операнда;
 - для дизъюнкции значений двух или более операндов: $x_1 \vee x_1 \vee \dots \vee x_1 = x_1$ – результирующее значение дизъюнкции двух или более логических операндов при равенстве их значений равно значению логического операнда.
7. Закон поглощения:
- при поглощении операции логического умножения (И) операцией логического сложения (ИЛИ): $x_1 \vee x_1 \& x_2 = x_1$ – результирующее значение дизъюнкции первого операнда с результатом конъюнкции значений первого и второго логических операндов зависит от значения первого операнда, поскольку происходит поглощение конъюнкции дизъюнкцией вследствие свойств этих двух логических операций и независимо от значения второго логического операнда результирующее значение определяется значением первого логического операнда;
 - при поглощении операции логического сложения (ИЛИ) операцией логического умножения (И): $x_1 \& (x_1 \vee x_2) = x_1$ – результирующее значение конъюнкции первого операнда с результатом дизъюнкции значений первого и второго логических операндов зависит от значения первого операнда, поскольку происходит поглощение дизъюнкции конъюнкцией вследствие свойств этих двух логических операций и независимо от значения второго логического операнда результирующее значение определяется значением первого логического операнда;

Для доказательства корректности седьмого закона используем законы (правила).

- $x_1 \vee x_1 \& x_2 = x_1 \& 1 \vee x_1 \& x_2 = x_1 \& (1 \vee x_2) = x_1 \& 1 = x_1$ – при поглощении операции логического умножения (И) операцией логического сложения (ИЛИ);
 - для выполнения операции конъюнкция двух логических операндов применим распределительный закон булевой алгебры: значение логического операнда x_1 умножается на 1 (нейтральный элемент по умножению) без изменения результата логического выражения, а затем получим $x_1 \& (1 \vee x_2)$;
 - для операции конъюнкция применим дистрибутивный закон и закон универсального множества для ИЛИ и И, а затем получим $x_1 \& (1 \vee x_2) = x_1 \& 1 = x_1$;
- $x_1 \& (x_1 \vee x_2) = x_1 \& x_1 \vee x_1 \& x_2 = x_1 \vee x_1 \& x_2 = x_1$ – при поглощении операции логического сложения (ИЛИ) операцией логического умножения (И):
 - для операции конъюнкция двух логических операндов применим распределительный закон булевой алгебры, а затем получим $x_1 \& x_1 \vee x_1 \& x_2$;
 - для операции конъюнкция применим закон повторения: $x_1 \& x_1 = x_1$;
 - для операции конъюнкция применим закон поглощения операции логического умножения (И) операцией логического сложения (ИЛИ): $x_1 \vee x_1 \& x_2 = x_1$.

8. Законы инверсии:

а) закон дополнения:

- для конъюнкции, операции логического умножения значений двух операндов: $x_1 \& \text{НЕ}(x_1) = 0$ – результирующее значение логического умножения значения логического операнда с его инвертированным значением равно нулю, поскольку в любом случае образуется аннулирующий элемент по умножению;
- для дизъюнкции, операции логического сложения значений двух операндов: $x_1 \vee \text{НЕ}(x_1) = 1$ – результирующее значение логического сложения значения логического операнда с его инвертированным значением равно единице, поскольку в любом случае образуется одна 1 слева или справа от знака операции;

б) закон склеивания:

- при поглощении операции логического сложения (ИЛИ) операцией логического умножения (И): $x_1 \& x_2 \vee x_1 \& \text{НЕ}(x_2) = x_1$ – вне зависимости от значения логического операнда x_2 в результате операции дизъюнкции первого слагаемого (результата конъюнкции логического операнда x_1 и логического операнда x_2) и второго слагаемого (результата конъюнкции логического операнда x_1 и инвертированного значения логического операнда x_2) достигается равенство значению логического операнда x_1 , поскольку в любом случае будут получены промежуточные и окончательные решения в первом или во втором слагаемом:
 - во-первых, аннулирующий элемент по умножению (0) – оператор x_2 ($\text{НЕ } x_2$);
 - во-вторых, нейтральный элемент по умножению (1) – оператор x_2 ($\text{НЕ } x_2$);
 - в-третьих, конъюнкция аннулирующего элемента по умножению (0) и значения логического операнда x_1 равна нулю в силу свойств конъюнкции;
 - в-четвертых, конъюнкция нейтрального элемента по умножению (1) со значением логического операнда x_1 равна значению логического операнда x_1 .

- при поглощении операции логического умножения (И) операцией логического сложения (ИЛИ): $(x_1 \vee x_2) \& (x_1 \vee \text{НЕ}(x_2)) = x_1$ – вне зависимости от значения логического операнда x_2 в результате операции конъюнкции первого множителя (результата дизъюнкции логического операнда x_1 и логического операнда x_2) и второго множителя (результата конъюнкции логического операнда x_1 и инвертированного значения логического операнда x_2) достигается равенство значению логического операнда x_1 , поскольку в любом случае будут получены промежуточные и окончательные решения в первом или во втором множителях:
 - во-первых, нейтральный элемент по сложению (0) – оператор x_2 ($\text{НЕ } x_2$);
 - во-вторых, логическая единица (1), которая обеспечивает истинность результирующего значения дизъюнкции одного из множителей – оператор x_2 ($\text{НЕ } x_2$);
 - в-третьих, согласно описанному ранее и используемому закону нулевого множества для операции логического сложения: дизъюнкция нейтрального элемента по сложению (0) и значения логического операнда x_1 равна x_1 ;
 - в-четвертых, согласно используемому закону универсального множества для операции логического сложения: дизъюнкция логической единицы (1) и значения логического операнда x_1 равна 1 в силу свойств дизъюнкции;
 - в-пятых, согласно закону универсального множества для операции логического умножения: конъюнкция нейтрального элемента по умножению (1) и значения логического операнда x_1 равна значению самого операнда x_1 .
- в) закон двойного отрицания:
- снятие двойного отрицания при использовании инверсии: $\text{НЕ}(\text{НЕ}(x_1)) = x_1$ – инверсия инвертированного значения логического операнда равна непосредственно значению этого логического операнда в любом логическом выражении;
- г) правило Де-Моргана:
- при выполнении операции логического умножения: $\text{НЕ}(x_1 \& x_2) = \text{НЕ}(x_1) \vee \text{НЕ}(x_2)$ – инвертированное результирующее значение конъюнкции первого логического операнда и второго логического операнда эквивалентно дизъюнкции инвертированных значений первого и второго логических операндов, при этом непосредственно происходит переход от операции логического умножения к операции логического сложения значений двух логических операндов;
 - при выполнении операции логического сложения: $\text{НЕ}(x_1 \vee x_2) = \text{НЕ}(x_1) \& \text{НЕ}(x_2)$ – инвертированное результирующее значение дизъюнкции первого логического операнда и второго логического операнда эквивалентно конъюнкции инвертированных значений первого и второго логических операндов, при этом непосредственно происходит переход от операции логического сложения к операции логического умножения значений двух логических операндов.
9. Преобразование гибридных логических операций к набору простых операций:
- для операции логического умножения (штрих Шеффера): $x_1 | x_2 = \text{НЕ}(x_1 \& x_2)$;
 - для операции логического сложения (стрелка Пирса): $x_1 \downarrow x_2 = \text{НЕ}(x_1 \vee x_2)$.

Рассмотрим несколько примеров использования рассмотренных ранее законов алгебры логики для реализации возможности упрощения логических выражений.

Пример 3.3. Задано аналитическое представление логического выражения:

$$\overline{HE}(X_1 \& X_2) \vee \overline{HE}(X_1 \& X_2) \vee \overline{HE}(X_1 \& X_2) \vee \overline{HE}(X_1 \& X_2).$$

Требуется упростить представленное логическое выражение.

Решение:

$$\overline{HE}(X_1 \& X_2) \vee \overline{HE}(X_1 \& X_2) \vee \overline{HE}(X_1 \& X_2) \vee \overline{HE}(X_1 \& X_2) =$$

на первом шаге применим правило Де-Моргана

$$= \overline{HE}(X_1) \vee \overline{HE}(X_2) \vee \overline{HE}(X_1) \vee \overline{HE}(X_2) \vee \overline{HE}(X_1) \vee \overline{HE}(X_2) \vee \overline{HE}(X_1) \vee \overline{HE}(X_2) =$$

на втором шаге упорядочим последовательность логических операндов

$$= \overline{HE}(X_1) \vee \overline{HE}(X_1) \vee \overline{HE}(X_1) \vee \overline{HE}(X_1) \vee \overline{HE}(X_2) \vee \overline{HE}(X_2) \vee \overline{HE}(X_2) \vee \overline{HE}(X_2) =$$

на третьем шаге применим закон повторения для дизъюнкции

$$= \overline{HE}(X_1) \vee \overline{HE}(X_2) =$$

на четвертом шаге применим вторично правило Де-Моргана

$$= \overline{HE}(X_1 \& X_2).$$

Пример 3.4. Задана функциональная схема определенного логического устройства.

Необходимо восстановить аналитическое представление логического выражения в виде элементарной логической формулы и затем упростить ее (рис. 3.24).

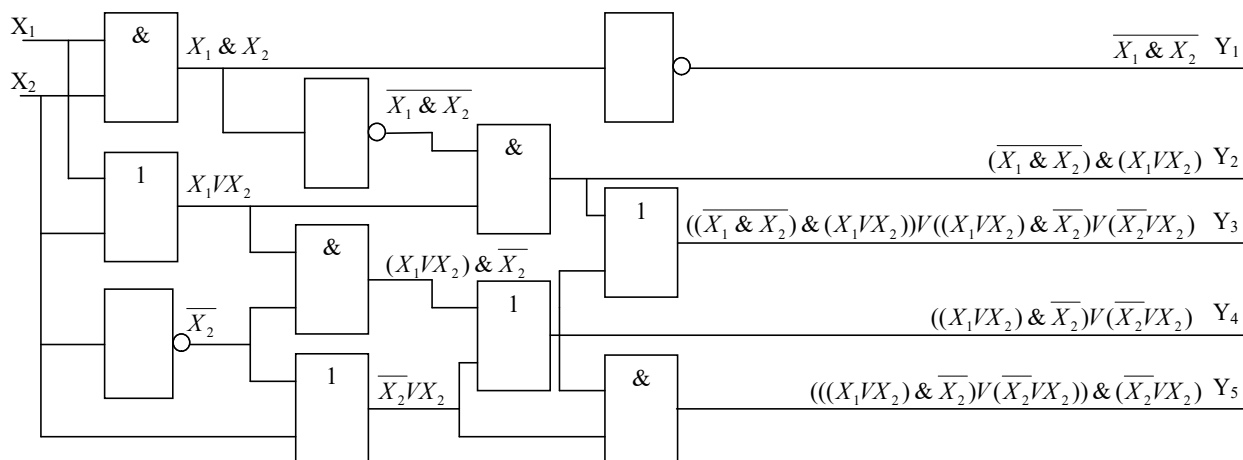


Рисунок 3.24. Функциональная схема логического устройства

В результате анализа функциональной схемы логического устройства необходимо записать систему логических уравнений, которые необходимо в дальнейшем упростить:

$$\begin{cases} y_1 = \overline{X_1 \& X_2}; \\ y_2 = (\overline{X_1 \& X_2}) \& (X_1 / X_2); \\ y_3 = ((\overline{X_1 \& X_2}) \& (X_1 / X_2)) / ((X_1 / X_2) \& \overline{X_2}) / (\overline{X_2} / X_2); \\ y_4 = ((X_1 / X_2) \& \overline{X_2}) / (\overline{X_2} / X_2); \\ y_5 = (((X_1 / X_2) \& \overline{X_2}) / (\overline{X_2} / X_2)) \& (\overline{X_2} / X_2) \end{cases}$$

Исходную систему логических уравнений попытаемся максимально упростить в рамках последовательности этапов, на которых она трансформируется.

На первом этапе происходит преобразование каждого логического уравнения:

- по отношению к первому уравнению применяется правило Де-Моргана: $\overline{X_1 \& X_2} = \overline{X_1} \vee \overline{X_2}$ (теоретически допустимо оставить его без изменений);
- по отношению ко второму уравнению применим распределительный (дистрибутивный) закон алгебры логики и далее выполним «раскрытие скобок»: $\overline{(X_1 \& X_2)} \& (X_1 \vee X_2) = X_1 \& \overline{X_1} \& X_2 \vee X_2 \& \overline{X_1} \& X_2$;
- по отношению к третьему уравнению также применим распределительный (дистрибутивный) закон алгебры логики и выполним «раскрытие скобок»: $((\overline{X_1} \& X_2) \& (X_1 \vee X_2)) \vee ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) =$
 $= (X_1 \& (\overline{X_1} \& X_2) \vee X_2 \& (\overline{X_1} \& X_2)) \vee ((X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2))$;
- по отношению к четвертому уравнению также применим распределительный (дистрибутивный) закон алгебры логики и выполним «раскрытие скобок»: $((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = (X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)$;
- по отношению к пятому уравнению также применим распределительный (дистрибутивный) закон алгебры логики и выполним «раскрытие скобок»: $((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) \& (\overline{X_2} \vee X_2) = ((X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2)$.

$$\left\{ \begin{array}{l} y_1 = \overline{X_1 \& X_2} = \overline{X_1} \vee \overline{X_2}; \\ y_2 = (\overline{X_1} \& X_2) \& (X_1 \vee X_2) = X_1 \& \overline{X_1} \& X_2 \vee X_2 \& \overline{X_1} \& X_2; \\ y_3 = ((\overline{X_1} \& X_2) \& (X_1 \vee X_2)) \vee ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = \\ = (X_1 \& (\overline{X_1} \& X_2) \vee X_2 \& (\overline{X_1} \& X_2)) \vee (X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2); \\ y_4 = ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = (X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2); \\ y_5 = (((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) = ((X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) \end{array} \right.$$

На втором этапе происходит повторное преобразование каждого логического уравнения:

- первое логическое уравнение остается без внесения каких-либо изменений;
- по отношению ко второму уравнению применяется правило Де-Моргана: $X_1 \& \overline{X_1} \& X_2 \vee X_2 \& \overline{X_1} \& X_2 = X_1 \& (\overline{X_1} \vee \overline{X_2}) \vee X_2 \& (\overline{X_1} \vee \overline{X_2})$;
- по отношению к третьему уравнению применим свойство операции дизъюнкция (истинность результирующего значения операции логического сложения значений двух логических операндов достигается в случае истинности значения одного из логических операндов), что позволяет преобразовать выражение $\overline{X_2} \vee X_2 = 1$ и затем записать полученный результат в следующем виде: $(X_1 \& (\overline{X_1} \& X_2) \vee X_2 \& (\overline{X_1} \& X_2)) \vee (X_1 \& \overline{X_2} \vee (X_2 \& \overline{X_2})) \vee (\overline{X_2} \vee X_2) =$
 $= (X_1 \& (\overline{X_1} \& X_2) \vee X_2 \& (\overline{X_1} \& X_2)) \vee (X_1 \& \overline{X_2} \vee 0) \vee 1$;
- по отношению к четвертому логическому уравнению также применим свойство операции дизъюнкция (истинность результирующего значения операции логического сложения значений двух логических операндов достигается в случае истинности значения хотя бы одного из логических операндов), что позволяет преобразовать выражение $\overline{X_2} \vee X_2 = 1$ и затем записать полученный после преобразования результат в следующем далее виде: $(X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = (X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee 1$;

- в пятом уравнении выделим скобками дизъюнкцию двух логических операндов $\overline{X_2} \vee X_2$ и конъюнкцию двух логических операндов $X_2 \& \overline{X_2}$ для целей дальнейшего эквивалентного преобразования и дальнейшего упрощения выражения:

$$\begin{aligned} & ((X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) = \\ & = ((X_1 \& \overline{X_2} \vee (X_2 \& \overline{X_2})) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) \end{aligned}$$

$$\left\{ \begin{aligned} y_1 &= X_1 \& X_2 = \overline{X_1} \vee \overline{X_2}; \\ y_2 &= (\overline{X_1 \& X_2}) \& (X_1 \vee X_2) = X_1 \& \overline{X_1 \& X_2} \vee X_2 \& \overline{X_1 \& X_2} = \\ &= X_1 \& (\overline{X_1} \vee \overline{X_2}) \vee X_2 \& (\overline{X_1} \vee \overline{X_2}) \\ y_3 &= ((\overline{X_1 \& X_2}) \& (X_1 \vee X_2)) \vee ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = \\ &= (X_1 \& (\overline{X_1 \& X_2}) \vee X_2 \& (\overline{X_1 \& X_2})) \vee (X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = \\ &= (X_1 \& (\overline{X_1 \& X_2}) \vee X_2 \& (\overline{X_1 \& X_2})) \vee (X_1 \& \overline{X_2} \vee 0) \vee 1; \\ y_4 &= ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = (X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = \\ &= (X_1 \& \overline{X_2} \vee (X_2 \& \overline{X_2})) \vee 1; \\ y_5 &= (((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) = ((X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) = \\ &= ((X_1 \& \overline{X_2} \vee (X_2 \& \overline{X_2})) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) \end{aligned} \right.$$

На третьем этапе происходит повторное преобразование каждого логического уравнения:

- первое логическое уравнение остается без внесения каких-либо изменений;
- по отношению ко второму уравнению применяется распределительный закон: $X_1 \& (\overline{X_1} \vee \overline{X_2}) \vee X_2 \& (\overline{X_1} \vee \overline{X_2}) = (X_1 \& \overline{X_1}) \vee X_1 \& \overline{X_2} \vee X_2 \& \overline{X_1} \vee X_2 \& \overline{X_2}$;

- по отношению к третьему уравнению повторно применим свойство операции дизъюнкция (истинность результирующего значения операции логического сложения значений двух логических операндов достигается в случае истинности значения одного из логических операндов), что позволяет преобразовать выражение $(X_1 \& \overline{X_2} \vee 1) \vee 1 = 1$ и записать результат в следующем виде:

$$\begin{aligned} & (X_1 \& (\overline{X_1 \& X_2}) \vee X_2 \& (\overline{X_1 \& X_2})) \vee ((X_1 \& \overline{X_2} \vee 0) \vee 1) = \\ & = (X_1 \& (\overline{X_1 \& X_2}) \vee X_2 \& (\overline{X_1 \& X_2})) \vee 1 \end{aligned}$$

- по отношению к четвертому уравнению также применим свойство операции конъюнкция (истинность результирующего значения операции логического умножения значений двух логических операндов достигается в случае истинности значений обоих логических операндов), что позволяет преобразовать выражение $X_2 \& \overline{X_2} = 0$ и записать полученный результат в следующем виде:

$$(X_1 \& \overline{X_2} \vee (X_2 \& \overline{X_2})) \vee 1 = (X_1 \& \overline{X_2} \vee 0) \vee 1;$$

- в пятом уравнении воспользуемся свойствами операции конъюнкция (истинность результирующего значения операции логического умножения значений двух логических операндов достигается в случае истинности значений обоих логических операндов) и операции дизъюнкция (истинность результирующего значения операции логического сложения значений двух логических операндов достигается в случае истинности значения хотя бы одного из логических операндов), что позволяет преобразовать заданное выражение

$X_2 \& \overline{X_2} = 0$ и $\overline{X_2} \vee X_2 = 1$, а затем записать результат в следующем виде:
 $((X_1 \& \overline{X_2} \vee (X_2 \& \overline{X_2})) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) = ((X_1 \& \overline{X_2} \vee 0) \vee 1) \& 1$.

$$\left\{ \begin{array}{l} y_1 = \overline{X_1} \& X_2 = \overline{X_1} \vee \overline{X_2}; \\ y_2 = (\overline{X_1} \& X_2) \& (X_1 \vee X_2) = X_1 \& \overline{X_1} \& X_2 \vee X_2 \& \overline{X_1} \& X_2 = \\ = X_1 \& (\overline{X_1} \vee \overline{X_2}) \vee X_2 \& (\overline{X_1} \vee \overline{X_2}) = \\ = (X_1 \& \overline{X_1}) \vee X_1 \& \overline{X_2} \vee X_2 \& \overline{X_1} \vee (X_2 \& \overline{X_2}) \\ y_3 = ((\overline{X_1} \& X_2) \& (X_1 \vee X_2)) \vee ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = \\ = (X_1 \& (\overline{X_1} \& X_2)) \vee X_2 \& (\overline{X_1} \& X_2) \vee ((X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)) = \\ = (X_1 \& (\overline{X_1} \& X_2)) \vee X_2 \& (\overline{X_1} \& X_2) \vee ((X_1 \& \overline{X_2} \vee 0) \vee 1) = \\ = (X_1 \& (\overline{X_1} \& X_2)) \vee X_2 \& (\overline{X_1} \& X_2) \vee 1; \\ y_4 = ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = (X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = \\ = (X_1 \& \overline{X_2} \vee (X_2 \& \overline{X_2})) \vee 1 = (X_1 \& \overline{X_2} \vee 0) \vee 1; \\ y_5 = ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) \& (\overline{X_2} \vee X_2) = ((X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) = \\ = ((X_1 \& \overline{X_2} \vee (X_2 \& \overline{X_2})) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) = ((X_1 \& \overline{X_2} \vee 0) \vee 1) \& 1 \end{array} \right.$$

На четвертом этапе происходит повторное преобразование каждого логического уравнения:

- первое логическое уравнение не преобразуется и остается без изменений;
- по отношению ко второму логическому уравнению применим свойство операции конъюнкция (истинность результирующего значения операции логического умножения значений двух логических операндов достигается в случае истинности значений двух логических операндов) ($X_1 \& \overline{X_1} = 0$) и используем правило Де-Моргана ($X_1 \& \overline{X_2} = \overline{\overline{X_1} \& X_2}$, $X_2 \& \overline{X_1} = \overline{\overline{X_2} \& X_1}$), что позволяет преобразовать исходное выражение и записать результат в следующем виде:
 $(X_1 \& \overline{X_1}) \vee X_1 \& \overline{X_2} \vee X_2 \& \overline{X_1} \vee (X_2 \& \overline{X_2}) = 0 \vee (\overline{\overline{X_1} \& X_2}) \vee (\overline{\overline{X_2} \& X_1}) \vee 0$;
- по отношению к третьему логическому уравнению применим закон универсального множества по отношению к операции дизъюнкция (логическая единица справа в выражении обуславливает истинность всего логического выражения), что позволяет преобразовать исходное выражение и записать полученный результат в следующем виде: $(X_1 \& (\overline{X_1} \& X_2)) \vee X_2 \& (\overline{X_1} \& X_2) \vee 1 = 1$;
- по отношению к четвертому уравнению применим закон нулевого множества по отношению к операции дизъюнкция (ноль является нейтральным элементом по логическому сложению), что позволяет преобразовать исходное выражение $X_1 \& \overline{X_2} \vee 0 = X_1 \& \overline{X_2}$ и записать результат в следующем виде:
 $(X_1 \& \overline{X_2} \vee 0) \vee 1 = (X_1 \& \overline{X_2}) \vee 1$;
- в пятом уравнении применим закон нулевого множества для операции дизъюнкция (ноль выступает нейтральным элементом по логическому сложению), что позволяет преобразовать логическое выражение $X_1 \& \overline{X_2} \vee 0 = X_1 \& \overline{X_2}$, а затем представить полученный результат в следующем далее виде:
 $((X_1 \& \overline{X_2} \vee 0) \vee 1) \& 1 = (X_1 \& \overline{X_2} \vee 1) \& 1$;

На пятом этапе происходит повторное преобразование каждого логического уравнения:

- первое логическое уравнение остается без внесения каких-либо изменений;
- по отношению ко второму уравнению применим правило Де-Моргана ($\overline{\overline{X_1 \& X_2}} = \overline{\overline{X_1}} \& \overline{\overline{X_2}}, X_2 \& X_1 = \overline{\overline{X_2}} \& \overline{\overline{X_1}}$), что позволяет преобразовать исходное логическое выражение и записать полученный результат в следующем виде:
 $0V\overline{\overline{X_1 \& X_2}}V\overline{\overline{X_2 \& X_1}}V0 = 0V\overline{\overline{X_1}}V\overline{\overline{X_2}}V\overline{\overline{X_2}}V\overline{\overline{X_1}}V0$;
- третье логическое уравнение остается без внесения каких-либо изменений;
- по отношению к четвертому логическому уравнению применим закон универсального множества для операции дизъюнкции и учтем свойства операции логического сложения значений двух логических операндов (логическая единица справа в выражении обуславливает истинность всего логического выражения), что позволяет преобразовать исходное логическое выражение и записать полученный результат в следующем далее виде:
 $(X_1 \& \overline{X_2})V1 = 1$;
- в пятом логическом уравнении применим закон универсального множества для операции дизъюнкции (логическая единица справа в выражении обуславливает истинность всего логического выражения), что позволяет эквивалентно преобразовать логическое выражение $X_1 \& \overline{X_2}V1 = 1$, а затем представить полученный результат в следующем виде: $(X_1 \& \overline{X_2}V1) \& 1 = 1 \& 1$.

$$\left. \begin{aligned}
 y_1 &= \overline{X_1 \& X_2} = \overline{X_1}V\overline{X_2}; \\
 y_2 &= (\overline{X_1 \& X_2}) \& (X_1VX_2) = X_1 \& \overline{\overline{X_1 \& X_2}}VX_2 \& \overline{\overline{X_1 \& X_2}} = X_1 \& \overline{\overline{X_1}}V\overline{\overline{X_2}}VX_2 \& \overline{\overline{X_1}}V\overline{\overline{X_2}} = \\
 &= (X_1 \& \overline{X_1})VX_1 \& \overline{X_2}VX_2 \& \overline{X_1}V(\overline{X_2} \& \overline{\overline{X_2}}) = 0V\overline{\overline{X_1 \& X_2}}V\overline{\overline{X_2 \& X_1}}V0 = \\
 &= 0V\overline{\overline{X_1}}V\overline{\overline{X_2}}V\overline{\overline{X_2}}V\overline{\overline{X_1}}V0 \\
 y_3 &= ((\overline{X_1} \& \overline{X_2}) \& (X_1VX_2))V((X_1VX_2) \& \overline{X_2})V(\overline{X_2}VX_2) = \\
 &= (X_1 \& (\overline{X_1} \& \overline{X_2})VX_2 \& (\overline{X_1} \& \overline{X_2}))V((X_1 \& \overline{X_2}VX_2 \& \overline{X_2})V(\overline{X_2}VX_2)) = \\
 &= (X_1 \& (\overline{X_1} \& \overline{X_2})VX_2 \& (\overline{X_1} \& \overline{X_2}))V((X_1 \& \overline{X_2}V0)V1) = \\
 &= (X_1 \& (\overline{X_1} \& \overline{X_2})VX_2 \& (\overline{X_1} \& \overline{X_2}))V1 = 1; \\
 y_4 &= ((X_1VX_2) \& \overline{X_2})V(\overline{X_2}VX_2) = (X_1 \& \overline{X_2}VX_2 \& \overline{X_2})V(\overline{X_2}VX_2) = \\
 &= (X_1 \& \overline{X_2}V(X_2 \& \overline{X_2}))V1 = (X_1 \& \overline{X_2}V0)V1 = (X_1 \& \overline{X_2})V1 = 1; \\
 y_5 &= (((X_1VX_2) \& \overline{X_2})V(\overline{X_2}VX_2)) \& (\overline{X_2}VX_2) = ((X_1 \& \overline{X_2}VX_2 \& \overline{X_2})V(\overline{X_2}VX_2)) \& (\overline{X_2}VX_2) = \\
 &= (((X_1 \& \overline{X_2}V(X_2 \& \overline{X_2}))V(\overline{X_2}VX_2)) \& (\overline{X_2}VX_2)) = \\
 &= ((X_1 \& \overline{X_2}V0)V1) \& 1 = (X_1 \& \overline{X_2}V1) \& 1 = 1 \& 1
 \end{aligned} \right\}$$

На шестом этапе происходит повторное преобразование каждого логического уравнения:

- первое логическое уравнение остается без внесения каких-либо изменений;
- по отношению ко второму уравнению применим закон инверсии для снятия двойного отрицания ($\overline{\overline{X_2}} = X_2, \overline{\overline{X_1}} = X_1$), что позволяет преобразовать исходное выражение и затем записать полученный результат в следующем виде: $0V(\overline{\overline{X_1}V\overline{\overline{X_2}}})V(\overline{\overline{X_2}V\overline{\overline{X_1}}})V0 = 0V(\overline{\overline{X_1}VX_2})V(\overline{\overline{X_2}VX_1})V0$;
- третье логическое уравнение остается без внесения каких-либо изменений;
- четвертое логическое уравнение остается без внесения каких-либо изменений;
- в пятом логическом уравнении осуществлено вычисление результирующего значения конъюнкции логической единицы и логического нуля (истинность результирующего значения конъюнкции двух логических операндов достигается в случае истинности обоих значений логических операндов), а затем получен конечный результат в следующем виде: $1 \& 1 = 1$.

$$\left\{ \begin{array}{l}
 y_1 = \overline{X_1} \& X_2 = \overline{X_1}V\overline{X_2}; \\
 y_2 = (\overline{X_1} \& X_2) \& (X_1VX_2) = X_1 \& \overline{\overline{X_1} \& X_2}VX_2 \& \overline{\overline{X_1} \& X_2} = X_1 \& (\overline{X_1}V\overline{X_2})VX_2 \& (\overline{X_1}V\overline{X_2}) = \\
 = (X_1 \& \overline{X_1})VX_1 \& \overline{X_2}VX_2 \& \overline{X_1}V(X_2 \& \overline{X_2}) = 0V(\overline{\overline{X_1} \& X_2})V(\overline{\overline{X_2} \& X_1})V0 = \\
 = 0V(\overline{\overline{X_1}V\overline{X_2}})V(\overline{\overline{X_2}V\overline{X_1}})V0 = 0V(\overline{\overline{X_1}VX_2})V(\overline{\overline{X_2}VX_1})V0 \\
 y_3 = ((\overline{X_1} \& X_2) \& (X_1VX_2))V((X_1VX_2) \& \overline{X_2})V(\overline{X_2}VX_2) = \\
 = (X_1 \& (\overline{X_1} \& X_2))VX_2 \& (\overline{X_1} \& X_2))V((X_1 \& \overline{X_2}VX_2 \& \overline{X_2})V(\overline{X_2}VX_2)) = \\
 = (X_1 \& (\overline{X_1} \& X_2))VX_2 \& (\overline{X_1} \& X_2))V((X_1 \& \overline{X_2}V0)V1) = \\
 = (X_1 \& (\overline{X_1} \& X_2))VX_2 \& (\overline{X_1} \& X_2))V1 = 1; \\
 y_4 = ((X_1VX_2) \& \overline{X_2})V(\overline{X_2}VX_2) = (X_1 \& \overline{X_2}VX_2 \& \overline{X_2})V(\overline{X_2}VX_2) = \\
 = (X_1 \& \overline{X_2}V(X_2 \& \overline{X_2}))V1 = (X_1 \& \overline{X_2}V0)V1 = (X_1 \& \overline{X_2})V1 = 1; \\
 y_5 = (((X_1VX_2) \& \overline{X_2})V(\overline{X_2}VX_2)) \& (\overline{X_2}VX_2) = ((X_1 \& \overline{X_2}VX_2 \& \overline{X_2})V(\overline{X_2}VX_2)) \& (\overline{X_2}VX_2) = \\
 = (((X_1 \& \overline{X_2}V(X_2 \& \overline{X_2}))V(\overline{X_2}VX_2)) \& (\overline{X_2}VX_2)) = \\
 = ((X_1 \& \overline{X_2}V0)V1) \& 1 = (X_1 \& \overline{X_2}V1) \& 1 = 1 \& 1 = 1
 \end{array} \right.$$

На седьмом этапе происходит повторное преобразование каждого логического уравнения:

- первое логическое уравнение остается без внесения каких-либо изменений;
- по отношению ко второму логическому уравнению применим закон нулевого множества для операции дизъюнкция (ноль выступает нейтральным элементом по логическому сложению) ($0V(\overline{\overline{X_1}VX_2}) = (\overline{\overline{X_1}VX_2}), (\overline{\overline{X_2}VX_1})V0 = (\overline{\overline{X_2}VX_1})$), что позволяет преобразовать исходное выражение и записать полученный результат в следующем виде: $0V(\overline{\overline{X_1}VX_2})V(\overline{\overline{X_2}VX_1})V0 = (\overline{\overline{X_1}VX_2})V(\overline{\overline{X_2}VX_1})$;
- третье логическое уравнение остается без внесения каких-либо изменений;
- четвертое логическое уравнение остается без внесения каких-либо изменений;
- пятое логическое уравнение остается без внесения каких-либо изменений.

$$\left\{ \begin{array}{l}
y_1 = \overline{X_1} \& X_2 = \overline{X_1} \vee \overline{X_2}; \\
y_2 = (\overline{X_1} \& X_2) \& (X_1 \vee X_2) = X_1 \& \overline{X_1} \& X_2 \vee X_2 \& \overline{X_1} \& X_2 = X_1 \& (\overline{X_1} \vee \overline{X_2}) \vee X_2 \& (\overline{X_1} \vee \overline{X_2}) = \\
= (X_1 \& \overline{X_1}) \vee X_1 \& \overline{X_2} \vee X_2 \& \overline{X_1} \vee (X_2 \& \overline{X_2}) = 0 \vee \overline{\overline{X_1} \& X_2} \vee \overline{\overline{X_2} \& X_1} \vee 0 = \\
= 0 \vee \overline{\overline{X_1} \vee \overline{X_2}} \vee \overline{\overline{X_2} \vee \overline{X_1}} \vee 0 = 0 \vee \overline{\overline{X_1} \vee X_2} \vee \overline{\overline{X_2} \vee X_1} \vee 0 = \overline{\overline{X_1} \vee X_2} \vee \overline{\overline{X_2} \vee X_1} \\
y_3 = ((\overline{X_1} \& X_2) \& (X_1 \vee X_2)) \vee ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = \\
= (X_1 \& (\overline{X_1} \& X_2)) \vee X_2 \& (\overline{X_1} \& X_2) \vee ((X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)) = \\
= (X_1 \& (\overline{X_1} \& X_2)) \vee X_2 \& (\overline{X_1} \& X_2) \vee ((X_1 \& \overline{X_2} \vee 0) \vee 1) = \\
= (X_1 \& (\overline{X_1} \& X_2)) \vee X_2 \& (\overline{X_1} \& X_2) \vee 1 = 1; \\
y_4 = ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = (X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = \\
= (X_1 \& \overline{X_2} \vee (X_2 \& \overline{X_2})) \vee 1 = (X_1 \& \overline{X_2} \vee 0) \vee 1 = (X_1 \& \overline{X_2}) \vee 1 = 1; \\
y_5 = ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) \& (\overline{X_2} \vee X_2) = ((X_1 \& \overline{X_2} \vee X_2 \& \overline{X_2}) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) = \\
= ((X_1 \& \overline{X_2} \vee (X_2 \& \overline{X_2})) \vee (\overline{X_2} \vee X_2)) \& (\overline{X_2} \vee X_2) = \\
= ((X_1 \& \overline{X_2} \vee 0) \vee 1) \& 1 = (X_1 \& \overline{X_2} \vee 1) \& 1 = 1 \& 1 = 1
\end{array} \right.$$

На восьмом этапе происходит повторное преобразование каждого логического уравнения:

- 1 и 3-5 логические уравнения остаются без внесения каких-либо изменений;
- по отношению ко второму логическому уравнению применим правило Де-Моргана (инверсия результирующего значения дизъюнкции двух логических операндов равна конъюнкции инвертированных значений двух логических операндов) $((\overline{\overline{X_1} \vee X_2}) \vee (\overline{\overline{X_2} \vee X_1})) = (\overline{\overline{X_1}}) \& \overline{X_2} \vee (\overline{\overline{X_2}}) \& \overline{X_1}$, а затем закон отрицания для снятия двойного отрицания (инверсия инвертированного значения логического операнда равна значению логического операнда) $(\overline{\overline{X_1}} = X_1$ и $\overline{\overline{X_2}} = X_2)$, что позволяет преобразовать исходное выражение и затем записать результат: $(\overline{\overline{X_1} \vee X_2}) \vee (\overline{\overline{X_2} \vee X_1}) = (\overline{\overline{X_1}}) \& \overline{X_2} \vee (\overline{\overline{X_2}}) \& \overline{X_1} = X_1 \& \overline{X_2} \vee X_2 \& \overline{X_1}$.

$$\left\{ \begin{array}{l}
y_1 = \overline{X_1} \& X_2 = \overline{X_1} \vee \overline{X_2}; \\
y_2 = (\overline{X_1} \& X_2) \& (X_1 \vee X_2) = X_1 \& \overline{X_2} \vee X_2 \& \overline{X_1} \\
y_3 = ((\overline{X_1} \& X_2) \& (X_1 \vee X_2)) \vee ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = 1; \\
y_4 = ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) = 1; \\
y_5 = ((X_1 \vee X_2) \& \overline{X_2}) \vee (\overline{X_2} \vee X_2) \& (\overline{X_2} \vee X_2) = 1
\end{array} \right.$$

Согласно исходному заданию, а также в результате использования набора правил эквивалентного преобразования (законов алгебры логики или Булевой алгебры) получена упрощенная система логических уравнений, в некоторых из которых получены конечные достигнутые логические состояния (ноль и единица).

Исходная система логических выражений может быть последовательно преобразована различными способами посредством использования вариативного набора законов алгебры логики, которые выступают правилами эквивалентного преобразования и структурной декомпозиции сложных логических выражений к простым.

Пример 3.7. Сформировать аналитическое представление логического устройства и вычислить значение полученного логического выражения с учетом подстановки предварительно заданных значений всех имеющихся логических операндов.

На рис. 3.26 представлена функциональная схема логического устройства.

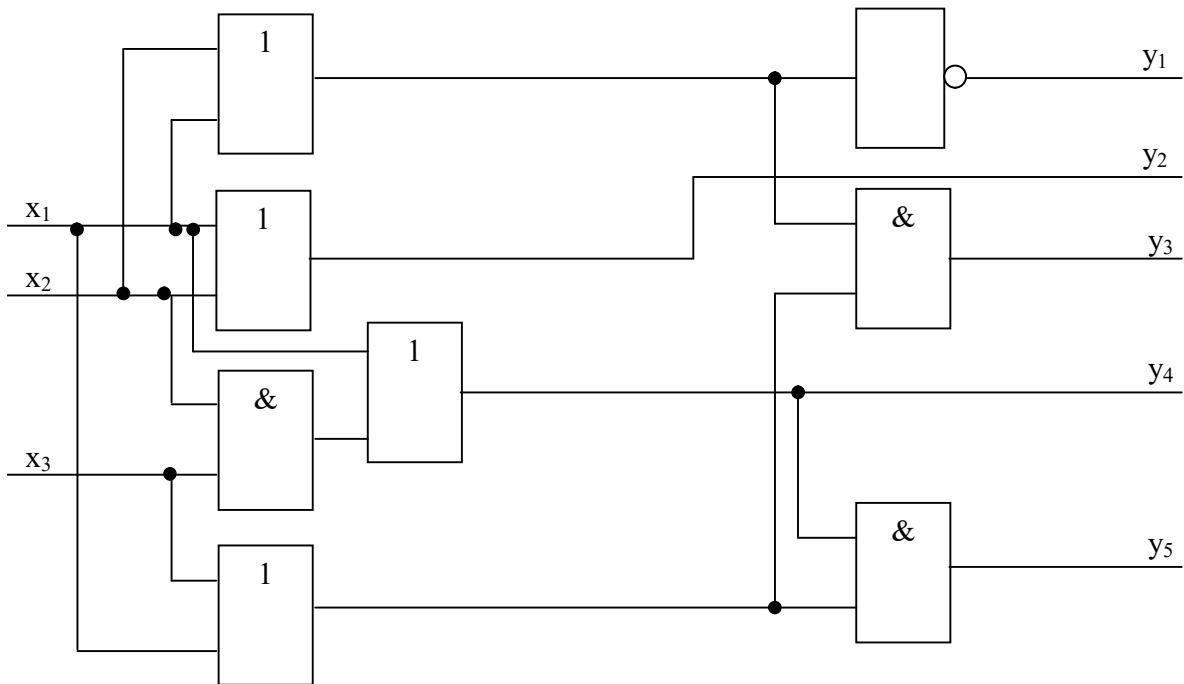


Рисунок 3.26. Функциональная схема логического устройства

Согласно представленной функциональной схеме логического устройства запишем систему логических уравнений, которые отражают уровни сигналов на выходах.

$$\begin{cases} y_1 = \overline{X_1 \vee X_2}; \\ y_2 = X_1 \vee X_2; \\ y_3 = (X_1 \vee X_2) \& (X_1 \vee X_3); \\ y_4 = X_1 \vee X_2 \& X_3; \\ y_5 = (X_1 \vee X_2 \& X_3) \& (X_1 \vee X_3) \end{cases}$$

Таблица 3.11

Пример расчета результирующих значений логических операций в таблице истинности на основе заданных бинарных комбинаций значений логических операндов

№	X ₁	X ₂	X ₃	X ₂ &X ₃	X ₁ ∨X ₂	X ₁ ∨X ₃	X ₁ ∨ X ₂ &X ₃	(X ₁ ∨X ₂)& &(X ₁ ∨X ₃)	Y ₁	Y ₂	Y ₃	Y ₄	Y ₅
0	0	0	0	0	0	0	0	0	1	0	0	0	0
1	0	0	1	0	0	1	0	0	1	0	0	0	0
2	0	1	0	0	1	0	0	0	0	1	0	0	0
3	0	1	1	1	1	1	1	1	0	1	1	1	1
4	1	0	0	0	1	1	1	1	0	1	1	1	1
5	1	0	1	0	1	1	1	1	0	1	1	1	1
6	1	1	0	0	1	1	1	1	0	1	1	1	1
7	1	1	1	1	1	1	1	1	0	1	1	1	1

Согласно представленной функциональной схеме логического устройства (рис. 3.26) и таблице истинности (табл. 3.11) построим графики на временной диаграмме состояний (рис. 3.27), которые отражают уровни промежуточных сигналов.

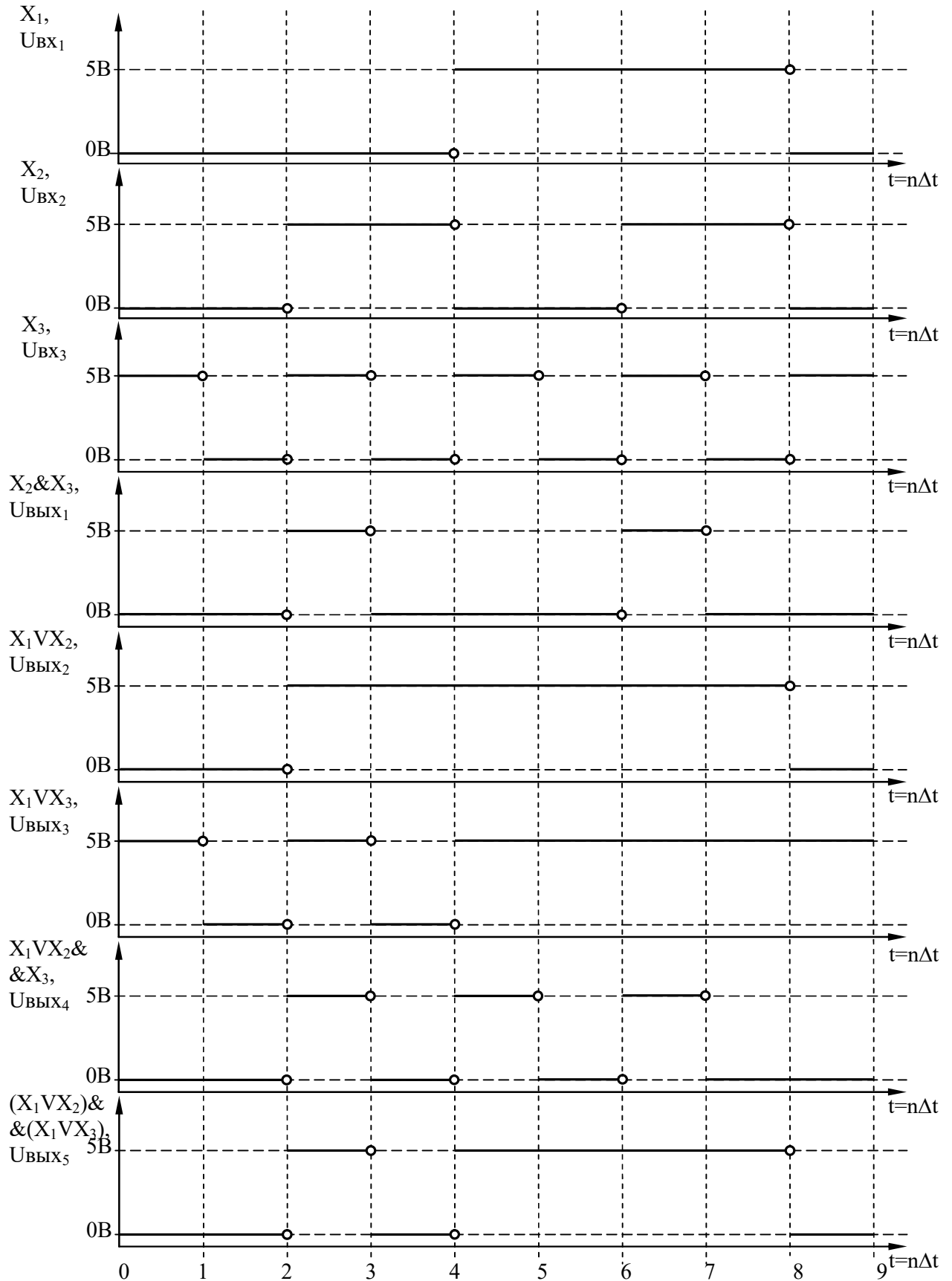


Рисунок 3.27. Временная диаграмма состояний на входе и промежуточных состояний

Продолжим построение графиков на временной диаграмме состояний (рис. 3.28), которые отражают уровни сигналов достигнутых на выходах логического устройства.

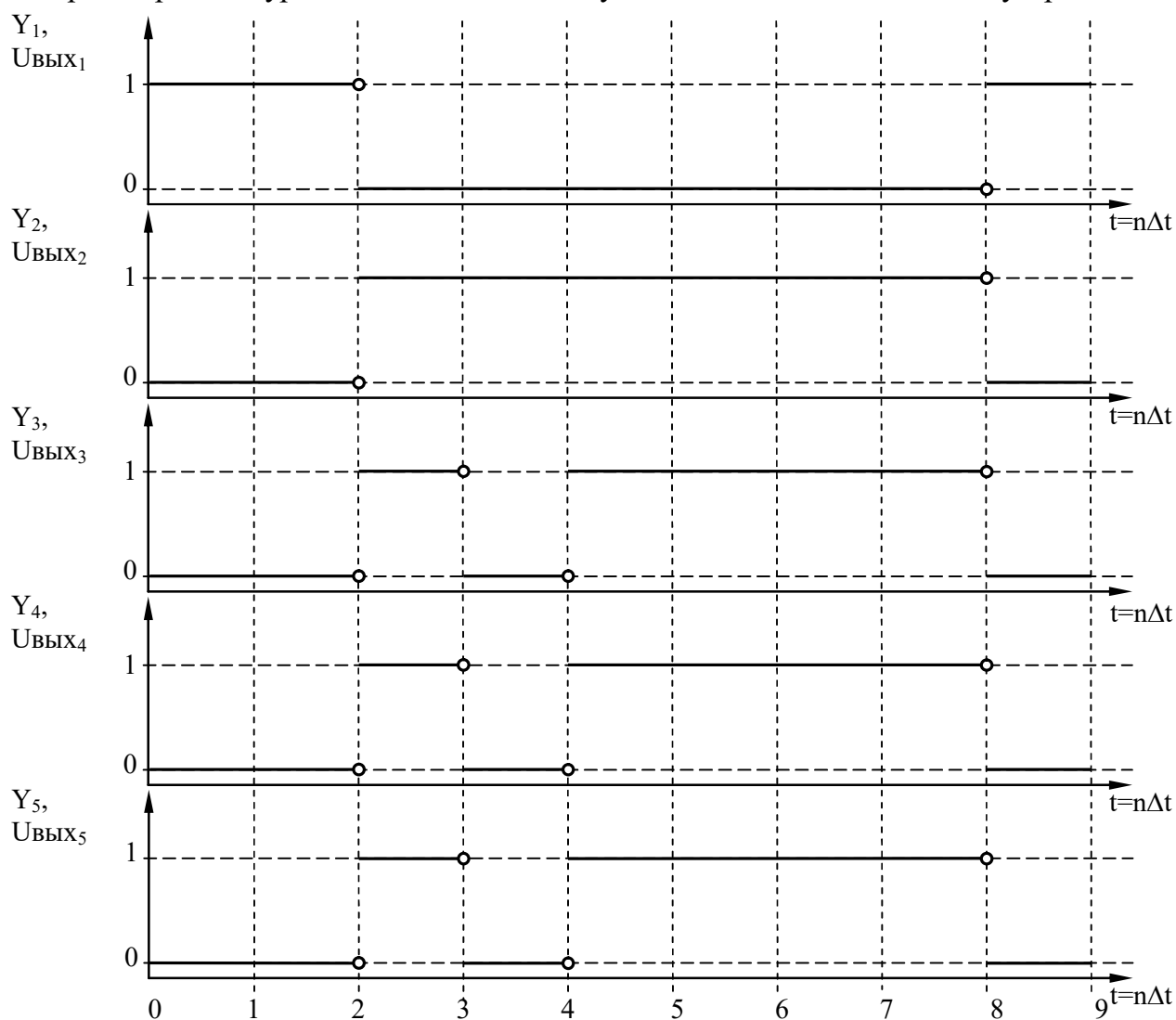


Рисунок 3.28. Временная диаграмма состояний на выходах логического устройства

Для интерпретации различных состояний логического устройства необходимо:

- сформировать таблицу истинности логического устройства (табл. 3.11) – определяет взаимно однозначное соответствие между промежуточными состояниями внутри логического устройства, достигнутыми состояниями на выходе логического устройства (Y_i) и заданными значениями логических операндов на входе логического устройства (память логических состояний не учитывается);
- построить вспомогательную временную диаграмму состояний логического устройства, которая отражает динамику изменения уровней напряжения на внутренних логических элементах функциональной схемы под влиянием изменения значений логических операндов (логических состояний на его входах);
- построить временную диаграмму состояний логического устройства, которая отражает динамику изменения уровней напряжения на выходах логического устройства под влиянием изменения уровней напряжения на его входах, соответствующих заданным значениям различных логических операндов.

3.5. Схемы комбинационной логики, обеспечивающие хранение информации

В процессе функционирования цифрового автомата осуществляется обработка дискретной информации, реализуются простейшие логические или арифметические операции, что обуславливает необходимость временного хранения промежуточных и результирующих значений различных операндов в период выполнения операции.

Определение

Схема называется **комбинаторной**, если в ней результирующее значение выходных параметров (вектор выхода) однозначно соответствует комбинации значений логических операндов заданных на входах только в данный момент времени.

Для обеспечения указанных возможностей используются схемы, которые обладают возможностью сохранять информацию в течение некоторого времени.

Определение

В **комбинационных схемах с памятью состояния** результирующее состояние после поступления некоторого входного воздействия изменяется и затем сохраняется до момента прихода новой входной комбинации, при этом реакция схемы на входное воздействие определяется еще и внутренним состоянием, которое характеризуется набором значений нескольких логических переменных (операндов).

Определение

Математической моделью комбинаторной схемы с памятью состояния является цифровой автомат, который описывается системой логических функций:

$$\begin{cases} Y_{i=(1,\dots,k)}^t = F_i(x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_m) \\ Z_{j=(1,\dots,p)}^{t+1} = F_j(x_1, x_2, \dots, x_n, z_1, z_2, \dots, z_m) \end{cases}, \text{ где}$$

$Y_{i=(1,\dots,k)}^t$ – уравнения для выходов логической схемы в момент времени t ;

$Z_{j=(1,\dots,p)}^{t+1}$ – уравнения для внутренних переменных, которые характеризуют

внутреннее состояние схемы к моменту следующего изменения ее состояния $(t+1)$;

n – разрядность входного слова, которая определяется количеством логических операндов (переменных) на входах $X = \{x_1, x_2, \dots, x_n\}$ логического устройства;

m – разрядность выходного слова, которая определяется количеством переменных характеризующих внутреннее состояние логического устройства $Z_{j=(1,\dots,p)}^{t+1}$;

i – индекс уравнения выхода, которое определяет состояние на момент времени t ;

j – индекс уравнения внутреннего состояния для момента времени $t+1$;

k – количество уравнений выхода, которые определяют достигнутое (установившееся) состояние на выходах произвольного логического устройства $Y_{i=(1,\dots,k)}^t$;

p – количество уравнений внутреннего состояния, которые определяют установившееся состояние внутренних переменных в предыдущий момент времени $(t+1)$.

Типовым примером комбинаторной схемы обладающей памятью состояния является триггер – бистабильная ячейка, которая способна хранить 1 бит информации.

Рассмотрим функциональную схему триггера (бистабильная ячейка), которая способна хранить 1 бит информации и обладает эффектом памяти состояния (рис. 3.29).

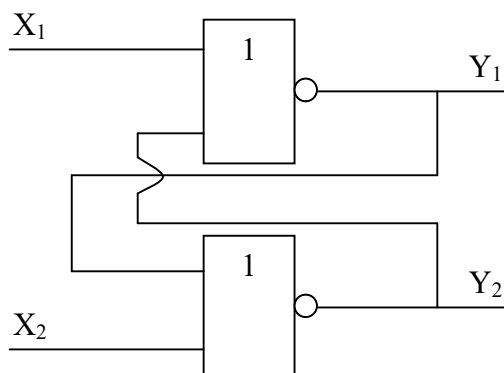


Рисунок 3.29. Функциональная схема триггера на основе двух элементов ИЛИ-НЕ

Функциональная схема триггера построена с использованием логических функций ИЛИ-НЕ и может быть описана следующей системой логических уравнений:

$$\begin{cases} Y_1 = \overline{X_1 \vee Y_2}; \\ Y_2 = \overline{X_2 \vee Y_1}; \\ Y_1 = \overline{Y_2}. \end{cases}$$

Проанализируем состояние на выходе триггера $Y=(Y_1, Y_2)$ при разных комбинациях входного слова, которое определяется значениями операндов $X=(X_1, X_2)$.

Первый этап

Поскольку триггер является элементом комбинаторной логики, который обладает памятью логического состояния, то необходимо учитывать предыдущее состояние на выходе триггера ($Y_0=(Y_{10}, Y_{20})$) и задать входное воздействие ($X=(X_1, X_2)$).

1) Пусть на входах задано $X_1=X_2=0$, а на выходах было достигнуто $Y_{10}=1, Y_{20}=0$.

Для определения значений вектора выхода (значения на выходах триггера) согласно приведенной выше системы уравнений (1 и 2 уравнения) зададим: начальное состояние входного воздействия ($X_{10}=0$ и $X_{20}=0$) с учетом предыдущих логических состояний достигнутых на двух выходах триггера ($Y_{10}=1$ и соответственно $Y_{20}=0$).

Выполним подстановку значений параметров, которые характеризуют предыдущее состояние достигнутое на выходе и начальное состояние входного воздействия.

Тогда исходная система логических уравнений выглядит следующим образом:

$$\begin{cases} Y_1 = \overline{X_1 \vee Y_2} = \overline{0 \vee 0} = 1; \\ Y_2 = \overline{X_2 \vee Y_1} = \overline{0 \vee 1} = 0; \\ Y_1 = \overline{Y_2} \Rightarrow \{Y_1 = 1 \Leftrightarrow Y_2 = 0\}. \end{cases} \quad \begin{array}{l} \text{При этом общее состояние триггера не меняется} \\ \text{поскольку } Y_{10}=1, Y_{20}=0 \text{ равно } Y_1=1, Y_2=0 \text{ с учетом} \\ \text{подачи входного воздействия } X_1=X_2=0. \end{array}$$

В результате обработки триггером входного воздействия $X=\{0,0\}$ в установившемся состоянии на выходах ($Y=\{1, 0\}$) логических противоречий не наблюдается.

При уровнях сигналов на обоих входах, которые соответствуют состоянию логического нуля ($X_1=0, X_2=0$) на выходе достигаются значения ($Y_1=1, Y_2=0$).

Зададим следующую комбинацию начальных значений параметров на входы.

2) Пусть на входах задано $X_1=0, X_2=1$, а на выходах было достигнуто $Y_{10}=1, Y_{20}=0$.

Для определения результирующих значений вектора выхода (значения на выходах триггера) согласно приведенной выше системы уравнений (1 и 2 уравнения) зададим: начальное состояние входного воздействия ($X_{10}=0$ и $X_{20}=0$) с учетом предыдущих логических состояний на выходах триггера ($Y_{10}=1$ и соответственно $Y_{20}=0$).

Выполним подстановку значений параметров, которые характеризуют предыдущее логическое состояние на выходе и начальное состояние входного воздействия.

Тогда исходная система логических уравнений выглядит следующим образом:

$$\text{по-} \begin{cases} Y_1 = \overline{X_1} \vee Y_2 = \overline{0 \vee 0} = 1; \\ Y_2 = \overline{X_2} \vee Y_1 = \overline{1 \vee 1} = 0; \\ \text{по-} \quad Y_1 = \overline{Y_2} \Rightarrow \{Y_1 = 1 \Leftrightarrow Y_2 = 0\}. \end{cases} \quad \begin{array}{l} \text{При этом общее состояние триггера не меняется} \\ \text{сколько } Y_{10}=1, Y_{20}=0 \text{ равно } Y_1=1, Y_2=0 \text{ с учетом} \\ \text{дачи входного воздействия } X_1=0, X_2=1. \end{array}$$

В результате отработки триггером входного воздействия $X=\{0,1\}$ в установившемся состоянии на выходах $Y=\{1, 0\}$ логических противоречий не наблюдается.

При уровнях сигналов на обоих входах триггера ($X_1=0, X_2=1$) на двух выходах этого логического устройства достигаются результирующие значения ($Y_1=1, Y_2=0$).

В первых двух случаях триггер не меняет и хранит предыдущее состояние.

Второй этап.

Зафиксируем начальное состояние на двух выходах триггера $Y_0=\{Y_{10}, Y_{20}\}$ в предыдущий момент времени и рассмотрим реакцию (установившееся логическое состояние триггера) на другие возможные входные воздействия $X=\{X_1, X_2\}$.

Изменяем только входное воздействие, внутреннее состояние равно $Y_0=\{0,1\}$.

1) Пусть на входах задано $X_1=1, X_2=0$, а на выходах было достигнуто $Y_{10}=0, Y_{20}=1$.

Для определения значений вектора выхода (значения на выходах триггера) согласно приведенной выше системы логических уравнений (1 и 2 уравнения) зададим: начальное состояние входного воздействия ($X_{10}=1$ и $X_{20}=0$) с учетом предыдущих логических состояний на выходах триггера ($Y_{10}=0$, соответственно $Y_{20}=1$).

Выполним подстановку значений параметров уравнений, характеризующих предыдущие состояния на выходах Y и начальные состояния входного воздействия X .

Тогда исходная система логических уравнений выглядит следующим образом:

$$\text{по-} \begin{cases} Y_1 = \overline{X_1} \vee Y_2 = \overline{1 \vee 1} = 0; \\ Y_2 = \overline{X_2} \vee Y_1 = \overline{0 \vee 0} = 1; \\ \text{по-} \quad Y_1 = \overline{Y_2} \Rightarrow \{Y_1 = 0 \Leftrightarrow Y_2 = 1\}. \end{cases} \quad \begin{array}{l} \text{При этом общее состояние триггера не меняется} \\ \text{сколько } Y_{10}=0, Y_{20}=1 \text{ равно } Y_1=0, Y_2=1 \text{ с учетом} \\ \text{дачи входного воздействия } X_1=1, X_2=0. \end{array}$$

В результате отработки триггером входного воздействия $X=\{1,0\}$ в установившемся состоянии на выходах $Y=\{0, 1\}$ логических противоречий не наблюдается.

При заданных уровнях сигналов на двух входах триггера ($X_1=1, X_2=0$) на выходах этого логического устройства достигаются взаимно инверсные значения ($Y_1=0, Y_2=1$).

Общее состояние триггера не изменяется: входное воздействие $X=\{1,0\}$ обуславливает изменение состояний на выходах триггера на $Y=\{0,1\}$ – установка триггера логическим нулем по прямому не инверсному выходу или сброс триггера.

Зададим следующие начальные значения параметров системы уравнений.

2) Пусть на входах задано $X_1=0, X_2=1$, а на выходах было достигнуто $Y_{10}=1, Y_{20}=0$.

Для определения значений вектора выхода (значения на выходах триггера) согласно приведенной выше системы логических уравнений (1 и 2 уравнения) зададим: начальное состояние входного воздействия ($X_{10}=0$ и $X_{20}=1$) с учетом предыдущего состояния достигнутого на выходах триггера ($Y_{10}=1$ и соответственно $Y_{20}=0$).

Выполним подстановку значений параметров, которые характеризуют предыдущее состояние на выходах Y и начальное состояние входного воздействия X .

Тогда исходная система логических уравнений выглядит следующим образом:

$$\text{по-} \begin{cases} Y_1 = \overline{X_1} Y_2 = \overline{0} \cdot 0 = 1; \\ Y_2 = \overline{X_2} Y_1 = \overline{1} \cdot 1 = 0; \\ Y_1 = \overline{Y_2} \Rightarrow \{Y_1 = 1 \Leftrightarrow Y_2 = 0\}. \end{cases} \quad \begin{array}{l} \text{При этом общее состояние триггера не меняется} \\ \text{сколько } Y_{10}=1, Y_{20}=0 \text{ равно } Y_1=1, Y_2=0 \text{ с учетом} \\ \text{дачи входного воздействия } X_1=0, X_2=1. \end{array}$$

В результате отработки триггером входного воздействия $X=\{0,1\}$ в установившемся состоянии на выходах ($Y=\{1, 0\}$) логических противоречий не наблюдается.

При уровнях сигналов на обоих входах триггера ($X_1=0, X_2=1$) на выходах этого логического устройства достигаются взаимно инверсные значения ($Y_1=1, Y_2=0$).

Общее состояние триггера не изменяется, входное воздействие $X=\{0,1\}$ обуславливает изменение состояний на двух выходах триггера на $Y=\{1,0\}$ – установка триггера логической единицей по прямому не инверсному выходу или установка триггера.

Зададим следующую комбинацию начальных значений параметров уравнения.

3) Пусть на входах задано $X_1=0, X_2=0$, а на выходах было достигнуто $Y_{10}=1, Y_{20}=0$.

Для определения значений выходного слова (значения на выходах триггера) согласно приведенной выше системы логических уравнений (1 и 2 уравнения) зададим: начальное состояние входного воздействия ($X_{10}=0$ и $X_{20}=0$) с учетом предыдущего логического состояния достигнутого на выходах триггера ($Y_{10}=1$, соответственно $Y_{20}=0$).

Выполним подстановку значений параметров, которые характеризуют предыдущее состояние на двух выходах Y и начальное состояние входного воздействия X .

Тогда исходная система логических уравнений выглядит следующим образом:

$$\text{по-} \begin{cases} Y_1 = \overline{X_1} Y_2 = \overline{0} \cdot 0 = 1; \\ Y_2 = \overline{X_2} Y_1 = \overline{0} \cdot 1 = 0; \\ Y_1 = \overline{Y_2} \Rightarrow \{Y_1 = 1 \Leftrightarrow Y_2 = 0\}. \end{cases} \quad \begin{array}{l} \text{При этом общее состояние триггера не меняется} \\ \text{сколько } Y_{10}=1, Y_{20}=0 \text{ равно } Y_1=1, Y_2=0 \text{ с учетом} \\ \text{дачи входного воздействия } X_1=X_2=0. \end{array}$$

В результате отработки триггером входного воздействия $X=\{0,0\}$ в установившемся состоянии на выходах ($Y=\{1, 0\}$) логических противоречий не наблюдается.

При уровнях сигналов на обоих входах, которые соответствуют состоянию логического нуля ($X_1=0, X_2=0$) на выходе логические состояния не меняются ($Y_1=1, Y_2=0$).

Таким образом, общее логическое состояние триггера не изменяется, поскольку входное воздействие $X=\{0,0\}$ не влияет на изменение состояний на выходах триггера.

Зададим следующую комбинацию начальных значений параметров уравнения.

Представленные расчеты и полученные результаты позволяют сделать ряд важных выводов, которые позволяют понять принцип функционирования триггера.

Сопоставим полученные результаты:

1. При входном воздействии $X=\{0, 0\}$ триггер продолжает хранить предыдущее состояние логического нуля или логической единицы по выходам $Y=\{Y_1, Y_2\}$:
 - первое возможное состояние при определенном входном воздействии: по прямому выходу $Y_1=0$ – триггер хранит состояние логического нуля, а по инверсному выходу $Y_2=1$ – триггер хранит состояние логической 1;
 - второе состояние: по прямому выходу $Y_1=1$, по инверсному выходу $Y_2=0$.
2. При входном воздействии $X=\{0, 1\}$ триггер сразу переключается и начинает хранить одно устойчивое логическое состояние по своим двум выходам $Y=\{Y_1, Y_2\}$: по прямому выходу $Y_1=1$ (единица), а по инверсному выходу $Y_2=0$ (ноль).
3. При входном воздействии $X=\{1, 0\}$ триггер сразу переключается и начинает хранить одно устойчивое логическое состояние по своим двум выходам $Y=\{Y_1, Y_2\}$: по прямому выходу $Y_1=0$ (ноль), по инверсному выходу $Y_2=1$ (единица).

Основное свойство в логике функционирования триггера – на прямом Y_1 и инверсном Y_2 выходах достигаются взаимно инверсные состояния $Y_1=\text{НЕ}(Y_2)$, что позволяет исходя из значения, которое достигнутого по первому выходу триггера непосредственно рассчитать значение по второму выходу этого логического элемента.

Для констатации факта изменения или сохранения общего логического состояния триггера необходимо учитывать три возможных логических состояния (табл. 3.12):

- бистабильная ячейка хранит 1 бит информации, общее состояние не меняется;
- триггер сброшен и хранит состояние логического нуля по прямому выходу – общее состояние потенциально меняется (запись № 2 и 3 в табл. 3.12);
- триггер установлен и хранит состояние логической единицы по прямому выходу – общее состояние потенциально меняется (запись № 4 и 5 в табл. 3.12);
- возникает состояние логической неопределенности, которое достигается по выходам – невозможно рассчитать значение на прямом и инверсном выходах.

Триггер называют бистабильной ячейкой поскольку по его прямому выходу достигается бинарное состояние: логический ноль (0) или логическая единица (1).

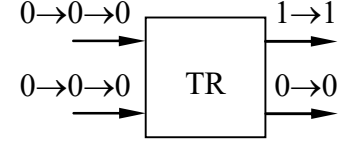
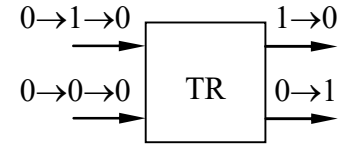
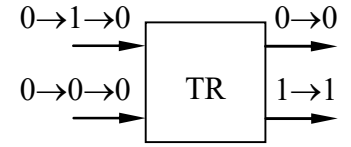
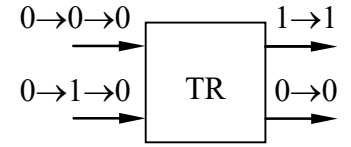
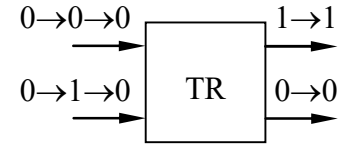
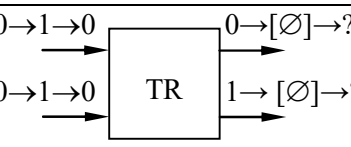
Результирующее установившееся состояние триггера ($Y=\{Y_1, Y_2\}$) после снятия входного воздействия ($X=\{X_1, X_2\}$) оценивается по его прямому выходу (Y_1):

- если по имеющемуся прямому выходу (Y_1) достигается состояние логического нуля, то триггер сброшен, при этом на инверсном выходе (Y_2) достигается установка состояние логической единицы, а сами выходы взаимно инверсны;
- если по прямому выходу (Y_1) достигается состояние логической единицы, то триггер установлен логической единицей, при этом на инверсном выходе (Y_2) достигается состояние логического нуля, а сами выходы взаимно инверсны.

При входном воздействии $X=\{1, 1\}$ ($X_1=1, X_2=1$) в триггере достигается состояние логической неопределенности по отношению к обоим его выходам $Y=\{Y_1, Y_2\}$ ($Y_1=0$ или 1, $Y_2=1$ или 0): невозможно аналитически определенно рассчитать значение логического состояния по прямому и инверсному выходу этого логического устройства.

Таблица 3.12

Динамика изменения состояний на выходах триггера в зависимости от уровней сигналов, которые подаются на его входы

	№	Исходное состояние по выходам	Входное воздействие	Состояние при подаче воздействия	Состояние после снятия воздействия	Последовательность переключения состояний
х р а н и т	1	$y_1 = 1$ $y_2 = 0$	$x_1 = 0$ $x_2 = 0$	$y_1 = 1$ $y_2 = 0$	$y_1 = 1$ $y_2 = 0$	 <p>состояние не изменилось (хранение информации)</p>
	2	$y_1 = 1$ $y_2 = 0$	$x_1 = 1$ $x_2 = 0$	$y_1 = 0$ $y_2 = 1$	$y_1 = 0$ $y_2 = 1$	 <p>состояние изменилось</p>
с б р о с	3	$y_1 = 0$ $y_2 = 1$	$x_1 = 1$ $x_2 = 0$	$y_1 = 0$ $y_2 = 1$	$y_1 = 0$ $y_2 = 1$	 <p>состояние не меняется</p>
	4	$y_1 = 1$ $y_2 = 0$	$x_1 = 0$ $x_2 = 1$	$y_1 = 1$ $y_2 = 0$	$y_1 = 1$ $y_2 = 0$	 <p>состояние не изменилось</p>
е д и н и ц а	5	$y_1 = 0$ $y_2 = 1$	$x_1 = 0$ $x_2 = 1$	$y_1 = 1$ $y_2 = 0$	$y_1 = 1$ $y_2 = 0$	 <p>состояние не меняется</p>
	[∅]	6	$y_1 = 1$ $y_2 = 0$	$x_1 = 1$ $x_2 = 1$	$y_1 = [∅]$ $y_2 = [∅]$	$y_1 = ?$ $y_2 = ?$
7		$y_1 = 0$ $y_2 = 1$	$x_1 = 1$ $x_2 = 1$	$y_1 = [∅]$ $y_2 = [∅]$	$y_1 = ?$ $y_2 = ?$	 <p>не имеет логического смысла !!!</p>

В результате анализа динамики процесса функционирования триггера построим таблицу (табл. 3.13), которая совпадает с теоретическим представлением.

Таблица 3.13

Возможные комбинации значений на входах триггера и достигнутые результирующие состояния на его выходах

Входная комбинация $X=\{X_1, X_2\}$	Достигнутое состояние на выходе
$X_1=0, X_2=0$	Сохраняет предыдущее состояние
$X_1=1, X_2=0$	Переключается в состояние «0» (сброс)
$X_1=0, X_2=1$	Переключается в состояние «1» (установка «1»)
$X_1=1, X_2=1$	Не имеет логического смысла

Временная диаграмма состояний входов и выходов триггера

Временная диаграмма, которая отражает динамику изменения состояний на выходах триггера в зависимости от входных воздействий представлена на рис. 3.30.

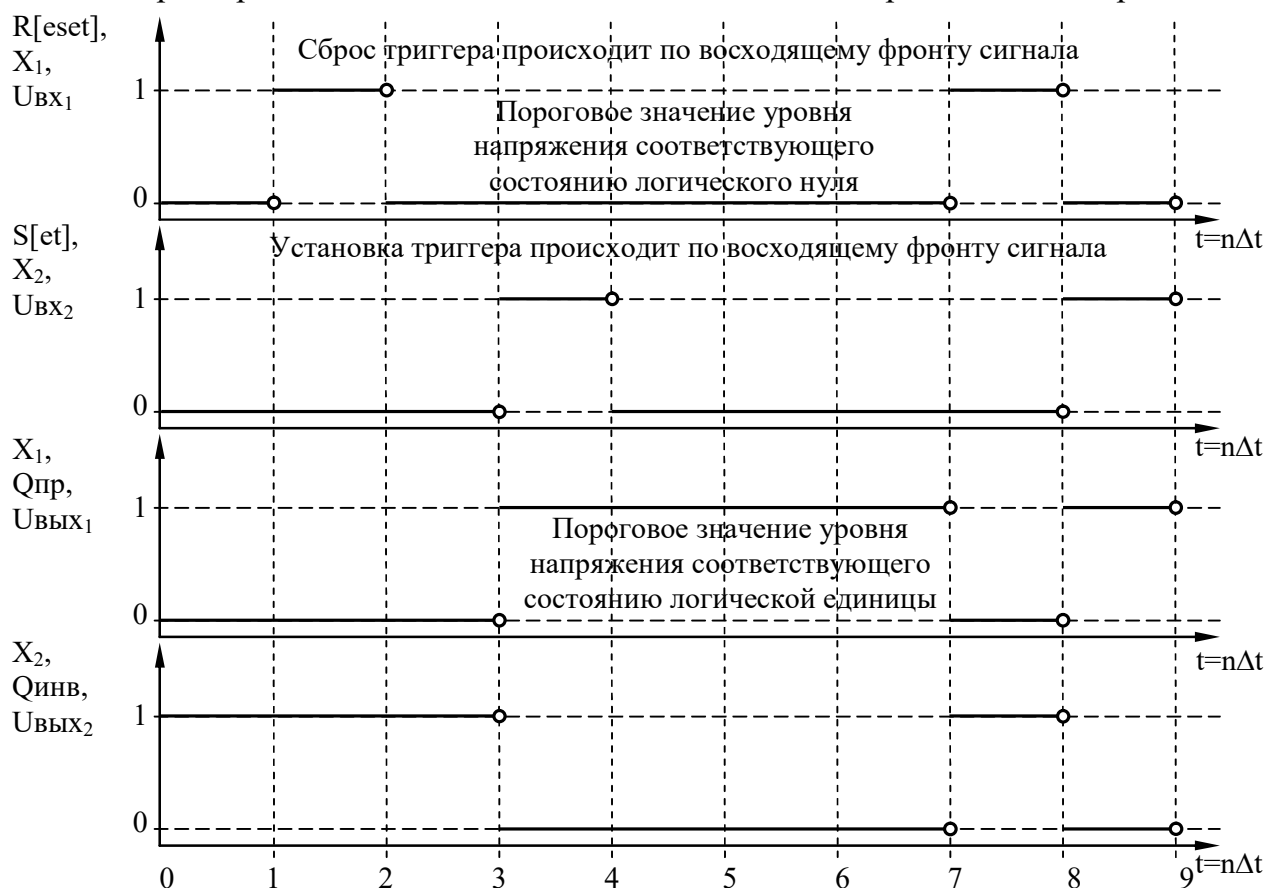


Рисунок 3.30. Временная диаграмма состояний выходов и входов триггера

Временная диаграмма переключения логических состояний на входах и выходах триггера позволяет отразить принцип функционирования этого устройства комбинаторной логики в динамике, а полученная ранее таблица истинности отражает его работу в статике – задает однозначное соответствие между достигнутыми состояниями на выходах триггера в зависимости от заданных входных комбинациях.

Рассмотрим динамику изменения уровней сигналов, которые характеризуют достигнутые логические состояния на двух выходах триггера под влиянием входных воздействий, а затем прокомментируем полученную временную диаграмму:

- на первом графике (рис. 3.30) изображена динамика изменения состояний уровней напряжения ($U_{вх1}$) на первом входе ($X_1, R[reset]$) представленного логического устройства, при этом достигается возможность обеспечения управления:
 - 0 – не влияет на изменение логического состояния на выходе триггера;
 - 1 – осуществляет сброс триггера и установку логическим нулем по прямому выходу, по инверсному выходу достигается состояние логической единицы;
- на втором графике изображена динамика изменения состояний уровней напряжения ($U_{вх2}$) на втором входе ($X_2, S[set]$) представленного логического устройства (рис. 3.30), при этом достигается возможность обеспечения управления:
 - 0 – не влияет на изменение логического состояния на выходе триггера;
 - 1 – осуществляет установку триггера логической единицей по прямому выходу, по инверсному выходу достигается состояние логического нуля;
- на третьем и четвертом графиках изображена динамика изменения уровней напряжения ($U_{вых1}, U_{вых2}$) на прямом (Y_1) и инверсном (Y_2) выходах триггера, которые соответствуют установившимся логическим состояниям и характеризуют внутреннее состояние этого логического устройства (для определенности положим, что начальное состояние на двух выходах триггера равно $Y=\{0, 1\}$):
 - на интервале между нулевым и первым дискретными моментами времени ($t|_{n=0,1} = n\Delta t$) пороговые значения напряжения ($U_{вх1}, U_{вх2}$) на обоих входах триггера $X=\{X_1, X_2\}$ соответствуют состоянию логического нуля (0), поэтому на его прямом выходе $Y_1 = \text{НЕ}(X_1 \vee Y_2)$ хранится предыдущее состояние – логический ноль; на его инверсном выходе $Y_2 = \text{НЕ}(Y_1) = \text{НЕ}(X_2 \vee Y_1)$ хранится предыдущее состояние – логическая единица (маркер в форме кружка на каждом графике означает исключение точки в момент переключения логического устройства при $n=0, 1, 2, \dots, k$);
 - на интервале между первым и вторым дискретными моментами времени ($t|_{n=1,2} = n\Delta t$) пороговое значение напряжения ($U_{вх1}, U_{вх2}$) на первом входе триггера X_1 соответствует состоянию логической единицы (инициирует безусловный сброс), на втором входе триггера X_2 соответствует состоянию логического нуля, поэтому на прямом выходе триггера $Y_1 = \text{НЕ}(X_1 \vee Y_2)$ достигается состояние логического нуля, а на инверсном выходе триггера $Y_2 = \text{НЕ}(Y_1) = \text{НЕ}(X_2 \vee Y_1)$ достигается состояние логической единицы (оба достигнутых логических состояния по двум выходам триггера продолжают сохраняться);
 - на интервале между вторым и третьим дискретными моментами времени ($t|_{n=2,3} = n\Delta t$) пороговое значение напряжения ($U_{вх1}, U_{вх2}$) на обоих входах триггера $X=\{X_1, X_2\}$ соответствуют состоянию логического нуля (0),

- поэтому на его прямом выходе $Y_1 = \text{НЕ}(X_1 \vee Y_2)$ хранится предыдущее состояние – логический ноль; на его инверсном выходе $Y_2 = \text{НЕ}(Y_1) = \text{НЕ}(X_2 \vee Y_1)$ хранится предыдущее состояние – логическая единица;
- на интервале между третьим и четвертым дискретными моментами времени ($t|_{n=3,4} = n\Delta t$) пороговое значение напряжения ($U_{вх1}, U_{вх2}$) на первом входе триггера X_1 соответствует состоянию логического нуля, на втором входе триггера X_2 соответствует состоянию логической единицы, поэтому на прямом выходе триггера $Y_1 = \text{НЕ}(X_1 \vee Y_2)$ достигается состояние логической единицы, а на инверсном выходе триггера $Y_2 = \text{НЕ}(Y_1) = \text{НЕ}(X_2 \vee Y_1)$ достигается состояние логического нуля (оба достигнутых логических состояния по двум выходам триггера меняются и начинают сохраняться до их принудительного изменения посредством следующего входного воздействия);
 - на интервале между четвертым и пятым дискретными моментами времени ($t|_{n=4,5} = n\Delta t$) пороговое значение напряжения ($U_{вх1}, U_{вх2}$) на обоих входах триггера $X = \{X_1, X_2\}$ соответствуют состоянию логического нуля (0), поэтому на его прямом выходе $Y_1 = \text{НЕ}(X_1 \vee Y_2)$ хранится предыдущее состояние – логическая единица; на его инверсном выходе $Y_2 = \text{НЕ}(Y_1) = \text{НЕ}(X_2 \vee Y_1)$ хранится предыдущее состояние – логический ноль;
 - на интервале между пятым и шестым дискретными моментами времени ($t|_{n=5,6} = n\Delta t$) пороговое значение напряжения ($U_{вх1}, U_{вх2}$) на обоих входах триггера $X = \{X_1, X_2\}$ соответствуют состоянию логического нуля (0), поэтому на его прямом выходе $Y_1 = \text{НЕ}(X_1 \vee Y_2)$ хранится предыдущее состояние – логическая единица; на его инверсном выходе $Y_2 = \text{НЕ}(Y_1) = \text{НЕ}(X_2 \vee Y_1)$ хранится предыдущее состояние – логический ноль;
 - на интервале между шестым и седьмым дискретными моментами времени ($t|_{n=6,7} = n\Delta t$) пороговое значение напряжения ($U_{вх1}, U_{вх2}$) на обоих входах триггера $X = \{X_1, X_2\}$ соответствуют состоянию логического нуля (0), поэтому на его прямом выходе $Y_1 = \text{НЕ}(X_1 \vee Y_2)$ хранится предыдущее состояние – логическая единица; на его инверсном выходе $Y_2 = \text{НЕ}(Y_1) = \text{НЕ}(X_2 \vee Y_1)$ хранится предыдущее состояние – логический ноль;
 - на интервале между седьмым и восьмым дискретными моментами времени ($t|_{n=7,8} = n\Delta t$) пороговое значение напряжения ($U_{вх1}, U_{вх2}$) на первом входе триггера X_1 соответствует состоянию логической единицы и на втором входе триггера X_2 соответствует состоянию логического нуля, поэтому на прямом выходе триггера $Y_1 = \text{НЕ}(X_1 \vee Y_2)$ достигается состояние логического нуля, а на инверсном выходе триггера $Y_2 = \text{НЕ}(Y_1) = \text{НЕ}(X_2 \vee Y_1)$ достигается состояние логической единицы (оба достигнутых логических состояния по двум выходам триггера меняются и начинают сохраняться до их принудительного изменения посредством входного воздействия);

- на интервале между восьмым и девятым дискретными моментами времени ($t|_{n=8,9} = n\Delta t$) пороговое значение напряжения ($U_{вх1}$, $U_{вх2}$) на первом входе триггера X_1 соответствует состоянию логического нуля и на втором входе триггера X_2 соответствует состоянию логической единицы, поэтому на прямом выходе триггера $Y_1 = \text{НЕ}(X_1 \vee Y_2)$ достигается состояние логической единицы, а на инверсном выходе триггера $Y_2 = \text{НЕ}(Y_1) = \text{НЕ}(X_2 \vee Y_1)$ достигается состояние логического нуля (оба достигнутых логических состояния по двум выходам триггера меняются и начинают сохраняться до их принудительного изменения посредством входного воздействия).

Таким образом, для реализации вычислительной процедуры на основе значений вектора входа и формирования результата на выходах триггера необходимо обеспечить выполнение последовательности условий и определенных действий:

- необходимо иметь аналитическое выражение (логическую функцию триггера), которое позволяет обеспечить расчет значений на двух выходах триггера;
- таблицу истинности триггера, которая задает однозначное соответствие между результирующими значениями на выходах триггера и комбинациями бинарных значений, которые задаются на его входы (операнды в функции);
- желательно построить функциональную схему логического устройства (триггера), которая позволяет спроектировать его аппаратную реализацию, а затем интерпретировать используемые условно графические обозначения, соответствующие набору простейших логических операций в основе триггера;
- временную диаграмму состояний, которые достигаются на выходах триггера при задании входных воздействий в виде уровней сигнала соответствующих определенным логическим состояниям (логическая единица или ноль);
- электрическую принципиальную схему аппаратной реализации логического устройства, которая позволяет вначале собрать и затем изучать принцип функционирования триггера, состоящего из набора условно графических обозначений полупроводниковых элементов и интегральных микросхем в его основе.

В результате сопоставления аналитического, табличного и графического представления триггера как логического элемента, включая его аппаратную реализацию, можно выявить несоответствия в процессе его функционирования и ошибки, которые были допущены на этапе проектирования и последующего подбора набора простейших логических операций и элементной базы для последующей сборки.

В заключение отметим, что данная схема служит основой класса комбинаторных логических схем, которые реализуют хранение информации и используются для записи и хранения одного бита информации, а триггер – это бистабильная ячейка, обладающая памятью 2^x логических состояний: логический ноль и логическая единица.

Познакомимся с несколькими типами различных триггеров, которые имеют ряд отличительных свойств и различий в процессе своего функционирования.

Вначале предлагается изучить принцип работы асинхронного RS-триггера.

3.5.1. Асинхронный RS-триггер

Представленный ранее триггер выступает элементарной бистабильной ячейкой, которая способна хранить 1 бит информации и является асинхронным RS-триггером.

Согласно представленной математической модели, в которой определены идентификаторы 2^x входов $X=\{X_1, X_2\}$ и 2^x выходов $Y=\{Y_1, Y_2\}$, потенциально достижимы:

- состояние хранения предыдущего достигнутого состояния по двум выходам $Y=\text{const}$ – при $X=\{0, 0\}$ на прямом и инверсном выходах триггера состояние не изменяется;
- установка логическим нулем (сброс) – при $X=\{1, 0\}$ на прямом и инверсном выходах достигаются соответственно состояния логического нуля и единицы ($Y=\{0, 1\}$);
- установка логической единицей (установка единицей) – при $X=\{0, 1\}$ на прямом и инверсном выходах принудительно и независимо достигаются соответственно состояния логического нуля и логической единицы $Y=\{1, 0\}$;
- состояние логической неопределенности – при $X=\{1, 1\}$ на прямом и инверсном выходах достигаются неопределенные состояния: быстрое попеременное переключение между логическим нулем и логической единицей $Y=\{1 \text{ или } 0, 0 \text{ или } 1\}$, а триггер не переходит в установившееся состояние.

На функциональных и электрических принципиальных схемах логических устройств триггер имеет определенное условное графическое обозначение, а его входы и выходы имеют идентификаторы (табл. 3.14), которые отличаются от формальных.

Таблица 3.14

Идентификаторы входов и выходов асинхронного RS-триггера

Идентификаторы входов и выходов на аналитическом представлении триггера	Идентификаторы входов и выходов триггера на функциональных и электрических принципиальных схемах
X_1	R[eset]
X_2	S[et]
Y_1	$Q_{пр}$
Y_2	$\overline{HE}(Q_{пр})=Q_{инв}$

Все рассматриваемые триггеры способны хранить 1 бит информации.

Состояние в асинхронном RS-триггере изменяется динамически и произвольно во времени посредством подачи входного воздействия аналогично ранее представленному аналитическому представлению (см. систему уравнений триггера):

- при задании входного воздействия равного логической единице на вход S происходит безусловное переключение прямого выхода триггера $Q_{пр}$ в состояние логической единицы, при этом изменяется уровень сигнала (напряжения) на его временной диаграмме, которая отражает динамику его работы;
- при задании входного воздействия равного логической единице на вход R происходит безусловное переключение прямого выхода триггера $Q_{пр}$ в состояние логического нуля, а инверсного выхода $Q_{инв}$ в состояние логической единицы, при этом изменяется уровень напряжения на его временной диаграмме.

Временная диаграмма состояний выходов и входов асинхронного триггера

Временная диаграмма состояний достигнутых на выходах асинхронного триггера ($U_{\text{вых1}}, U_{\text{вых2}}$) в зависимости от уровней сигналов (напряжения), которые задаются на его входы (R, S) представлена на рис. 3.31 и аналогична рассчитанной ранее (рис. 3.30).

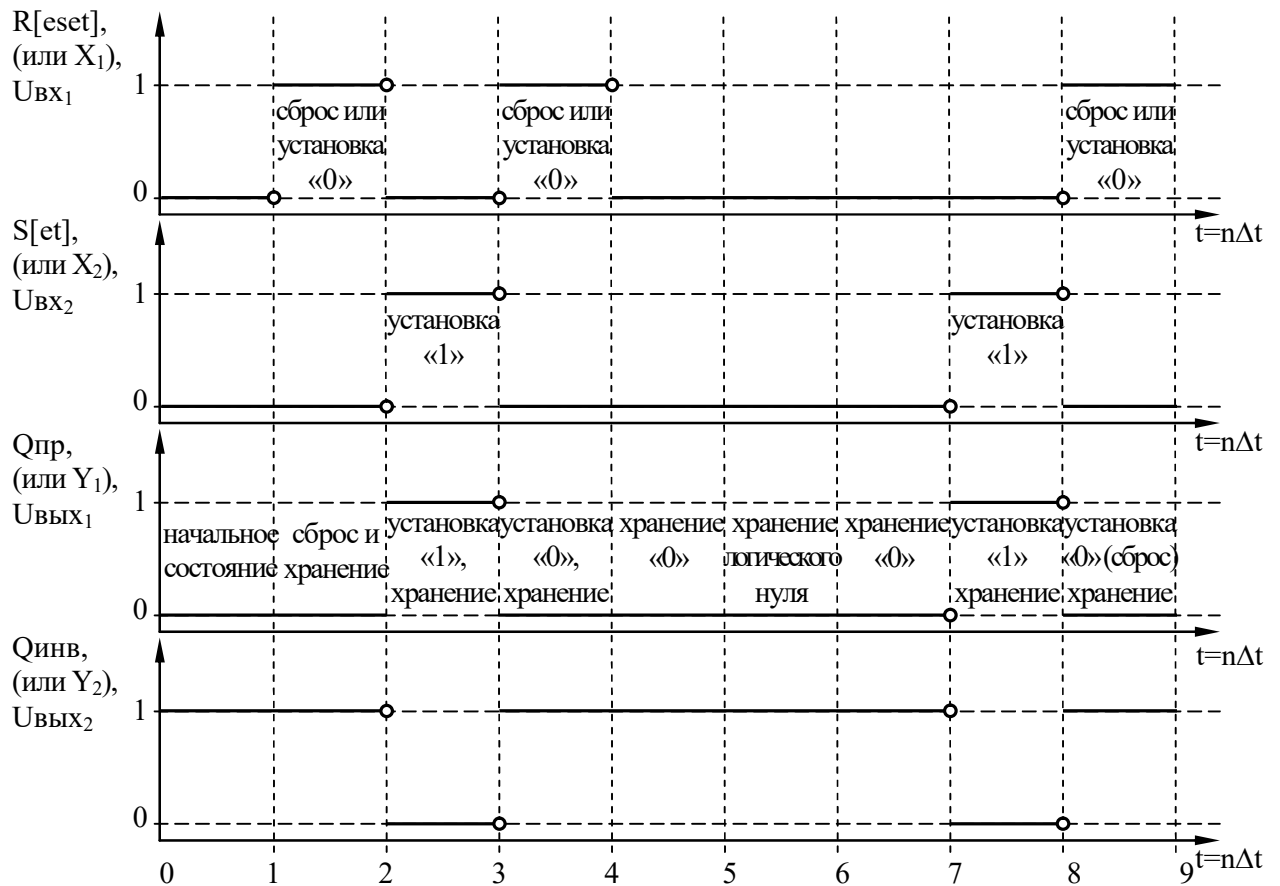


Рисунок 3.31. Временная диаграмма динамики изменения состояний на выходах RS-триггера в зависимости от уровней сигналов на его входах

Переключение асинхронного RS-триггера происходит в произвольный момент времени согласно входным воздействиям $X = \{0, 1; 1, 0\}$, но кроме $X = \{0, 0\}$ – не изменяет состояния, исключая $X = \{1, 1\}$ – комбинация не имеет логического смысла.

Таким образом, можно сформулировать определение асинхронного триггера.

Определение

Триггер называется **асинхронным**, если изменение его логических состояний на выходах $Y = \{Y_1, Y_2\}$ осуществляется в любой произвольный момент времени, но при создании необходимых условий на входах $R[\text{eset}]$ и $S[\text{et}]$: наличие входного воздействия $X = \{0, 1\}$ – установка «1» или $X = \{1, 0\}$ – установка «0» (сброс триггера).

Асинхронные триггеры применяются в основном в аналоговых системах.

Поскольку цифровые автоматы (компьютеры) являются дискретными по принципу своего функционирования, то в них применяются синхроимпульсы и тактовые генераторы для синхронизации в синхронных триггерах, которые переключаются только в определенные дискретные фиксированные моменты времени.

Устройства синхронизации конструируются и реализуются на основе схем комбинаторной логики, которые состоят из набора простых логических операций.

3.5.2. Синхронный RS-триггер

Иногда требуется, чтобы изменение состояний входов и выходов функциональных схем логических устройств и в частности триггеров происходило в фиксированные (дискретные) моменты времени, синхронно и согласованно во времени с некоторыми другими схемами комбинаторной логики в основе информационных систем (компьютеров).

Для обеспечения синхронного изменения логических состояний в цифровых автоматах (компьютерах) применяют специальные схемы синхронизации, которые обеспечивают генерацию синхронизирующих (стробирующих, тактирующих) импульсов.

Определение

Триггер называют **синхронным** (синхронизированным), если изменение уровней напряжения (логических состояний) на выходах $Q_{пр}$ и $Q_{инв}$ осуществляется под влиянием входных воздействий в дискретные или фиксированные моменты времени при использовании дополнительных устройств (схем синхронизации), которые обеспечивают синхронизацию: переключение логических состояний по выходу возможно только в случае, если уровни напряжения соответствуют логической единице одновременно по синхро-входу C и одному из управляющих входов R и S .

В качестве примера схем данного вида рассмотрим синхронный RS-триггер.

Функциональная схема синхронного RS-триггера

На представленной функциональной схеме синхронного триггера (рис. 3.32):

- слева штрихом выделен элемент синхронизации, который собран на основе двух конъюнкторов реализующих операцию логического сложения (И);
- справа штрихом выделен асинхронный RS-триггер, который собран на основе двух дизъюнкторов инверсных по выходу и реализующих операцию логического сложения (ИЛИ-НЕ) значений двух логических операндов.

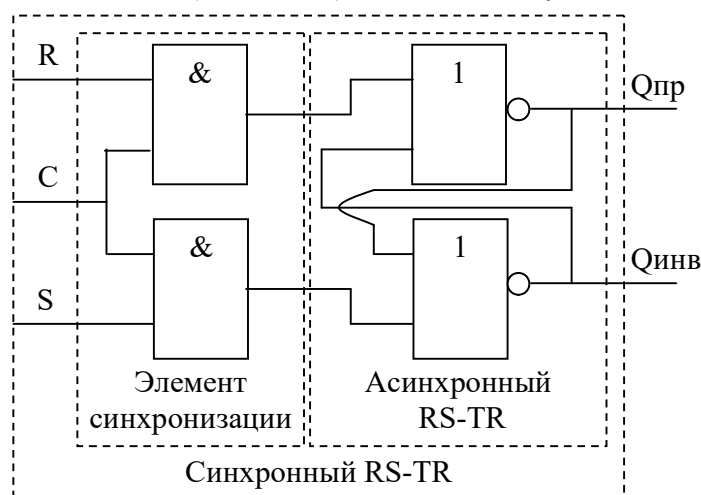


Рисунок 3.32. Функциональная схема синхронного RS-триггера

В данной схеме переключение синхронного триггера возможно при условиях:

- на дополнительный синхронизирующий-вход (C) синхронного триггера подан уровень напряжения соответствующий состоянию логической единицы;
- на одном из управляющих входов (R – для сброса или S – для установки «1») подан уровень напряжения соответствующий состоянию логической единицы.

Временная диаграмма

Временная диаграмма состояний (рис. 3.33), которые достигнуты на выходах синхронного триггера ($Q_{пр}$, $Q_{инв}$) в зависимости от уровней напряжения ($U_{вх1}$, $U_{вх2}$, $U_{вх3}$), поданным на три входа (R – сброс, S – установка «1», C – синхронизация).

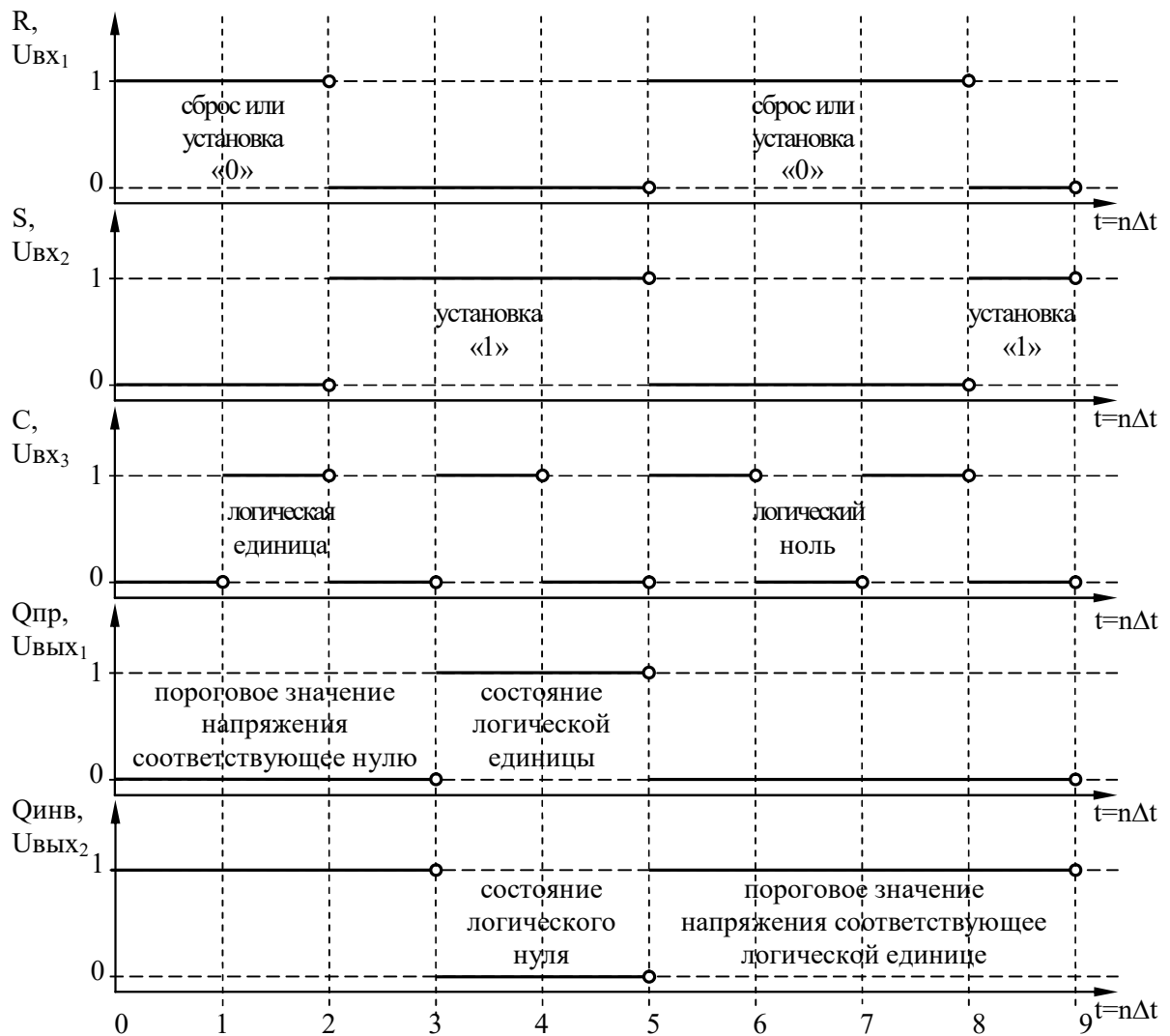


Рисунок 3.33. Временная диаграмма состояний на выходах синхронного триггера в зависимости от логических состояний на его входах

Принцип функционирования и временная диаграмма состояний синхронного триггера (рис. 3.33) аналогичны асинхронному триггеру (рис. 3.31), но имеется единственная особенность: переключение логических состояний на выходах $Q_{пр}$ и $Q_{инв}$ возможно только при наличии на синхронизирующем входе (C) уровня напряжения соответствующего состоянию логической единицы и достигается за счет подачи входных воздействий соответствующих состоянию логической единицы на управляющие входы (R или S). Переключение происходит в момент наличия сигналов по C и R или S .

В цифровых автоматах (ЭВМ) как правило реализуется генерация синхронизирующих импульсов посредством использования тактовых генераторов, которые обеспечивают синхронизацию переключения двух логических состояний (логический ноль и логическая единица), которые достигаются на выходах синхронного триггера ($Q_{пр}$, $Q_{инв}$). Тактовый генератор является компонентом архитектуры компьютера.

3.5.3. Тактируемый триггер

Сегодня существует большое количество разных способов синхронизации синхронного триггера, поэтому этот класс устройств имеет существенные особенности реализации: в зависимости от целей и задач использования, количества входов и выходов, логических операций на его функциональной схеме и элементной базы для его аппаратной реализации представленной на электрической принципиальной схеме.

Рассмотрим особенности реализации тактируемого триггера, который также обладает свойствами синхронизации, но имеет лишь один вход (Т) и два выхода ($Q_{пр}$, $Q_{инв}$).

Определение

Тактируемым триггером называют элемент комбинаторной логики, который попеременно изменяет свое состояние на прямом ($Q_{пр}$) и инверсном ($Q_{инв}$) выходах в момент изменения уровня порогового управляющего напряжения на синхронизирующем входе (Т) с 0В (логический ноль) на 5В (логическая единица).

Взаимно инверсное логическое состояние на обоих выходах тактируемого триггера переключается по восходящему фронту синхронизирующих импульсов.

В тактируемом триггере (Т-TR) используются тактирующие сигналы определенной формы: прямоугольная – «меандр», решетка – решетчатая функция.

Определение

Меандром (рис. 3.34) называют форму сигнала, у которого амплитуда (А) равна половине фазы изменения сигнала или периоду его квантования ($\Delta t = \text{const}$).

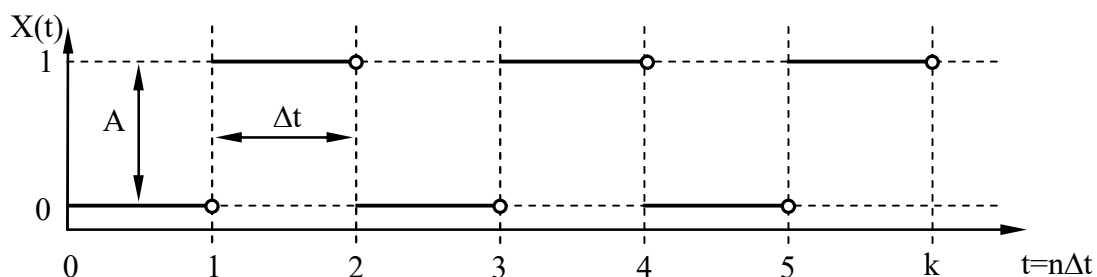


Рисунок 3.34. Сигнал прямоугольной формы типа «меандр»

Использование сигнала в форме меандра или решетчатой функции является оптимальным выбором для цифровых автоматов (компьютера), поскольку данная форма сигнала соответствует условиям и ограничениям аппаратной реализации компонентов, которые располагаются в основе архитектуры ЭВМ разного поколения.

При разработке компонентов архитектуры ЭВМ допускается использование стробирующих и синхронизирующих сигналов в форме решетчатой функции (рис. 3.35).

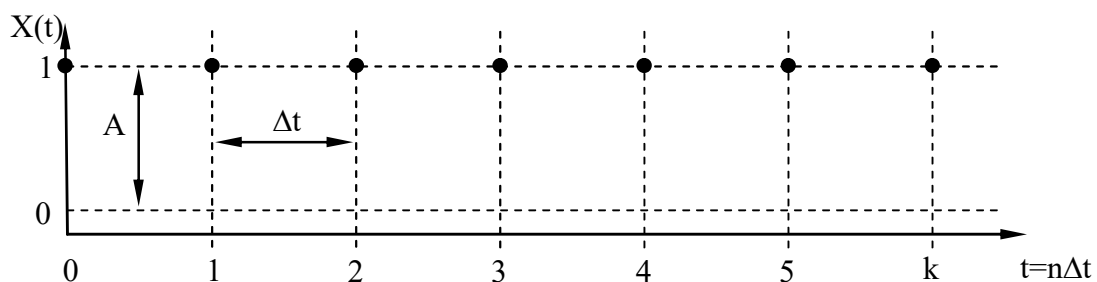


Рисунок 3.35. Дискретный сигнал типа «решетчатая функция»

Аналитическое представление тактируемого триггера

Принцип функционирования тактируемого триггера (Т-TR) описывается следующей системой логических уравнений, которые позволяют обеспечить расчет значений на его выходах ($Q_{\text{пр}}$, $Q_{\text{инв}}$) в зависимости от входного воздействия (Т):

$$\begin{cases} Q_{\text{пр}} = \overline{((Q_{\text{пр}} \& T) \vee Q_{\text{инв}})}; \\ Q_{\text{инв}} = \overline{(Q_{\text{пр}} \vee (T \& Q_{\text{инв}}))}; \\ Q_{\text{инв}} = \overline{Q_{\text{пр}}}. \end{cases}$$

Функциональная схема тактируемого триггера (Т-TR)

Тактируемый триггер (Т-TR) имеет всего один управляющий вход (Т), который обеспечивает синхронизацию попеременного переключения логических состояний (0 и 1) устанавливающихся на прямом ($Q_{\text{пр}}$) и инверсном ($Q_{\text{инв}}$) выходах.

Функциональная схема тактируемого триггера представлена на рис. 3.36.

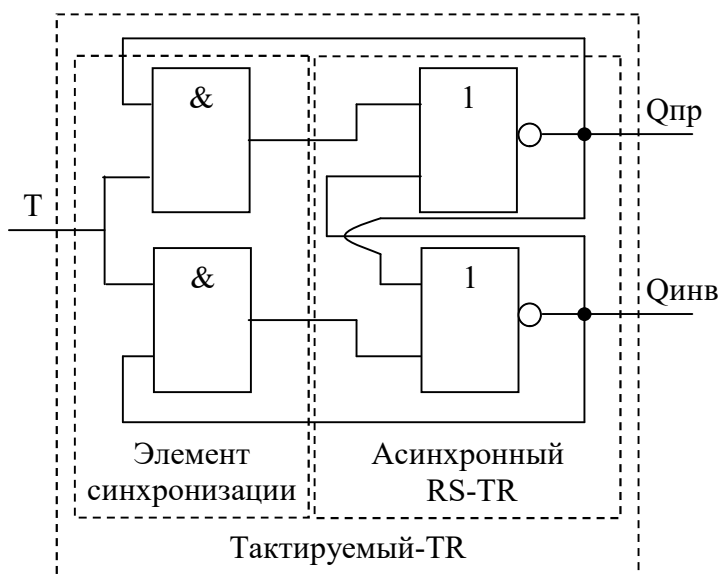


Рисунок 3.36. Функциональная схема тактируемого RS-триггера

Тактируемый триггер включает интегральную совокупность двух частей:

- элемент синхронизации – обеспечивает синхронность и синфазность попеременного переключения двух логических состояний (единица и ноль) на прямом ($Q_{\text{пр}}$) и инверсном ($Q_{\text{инв}}$) выходах асинхронного RS-триггера;
- асинхронный RS-триггер – бистабильная ячейка с памятью состояния на своих выходах ($Q_{\text{пр}}$ и $Q_{\text{инв}}$), собран посредством использования двух дизъюнкторов инверсных по выходу (логическая операция ИЛИ-НЕ).

Элемент синхронизации и асинхронный RS-триггер на функциональной схеме обозначены штриховой линией, а его реализация осуществлена с использованием простых логических операций и их стандартных обозначений для функциональных схем.

Переключение логических состояний тактируемого триггера (Т-TR) происходит попеременно по каждому восходящему фронту сигнала (уровню напряжения) типа меандр, который задается на управляющий вход (Т) в момент изменения порогового значения напряжения с 0В (логический ноль) на 5В (логическая единица).

Временная диаграмма состояний входов и выходов тактируемого триггера

На основе функциональной схемы и аналитического представления построена временная диаграмма динамики изменения состояний выходов и входов Т-триггера (рис. 3.37), которая позволяет говорить о переключении его выходов во взаимно инверсное состояние по каждому входному воздействию (Т – тактирующий вход).

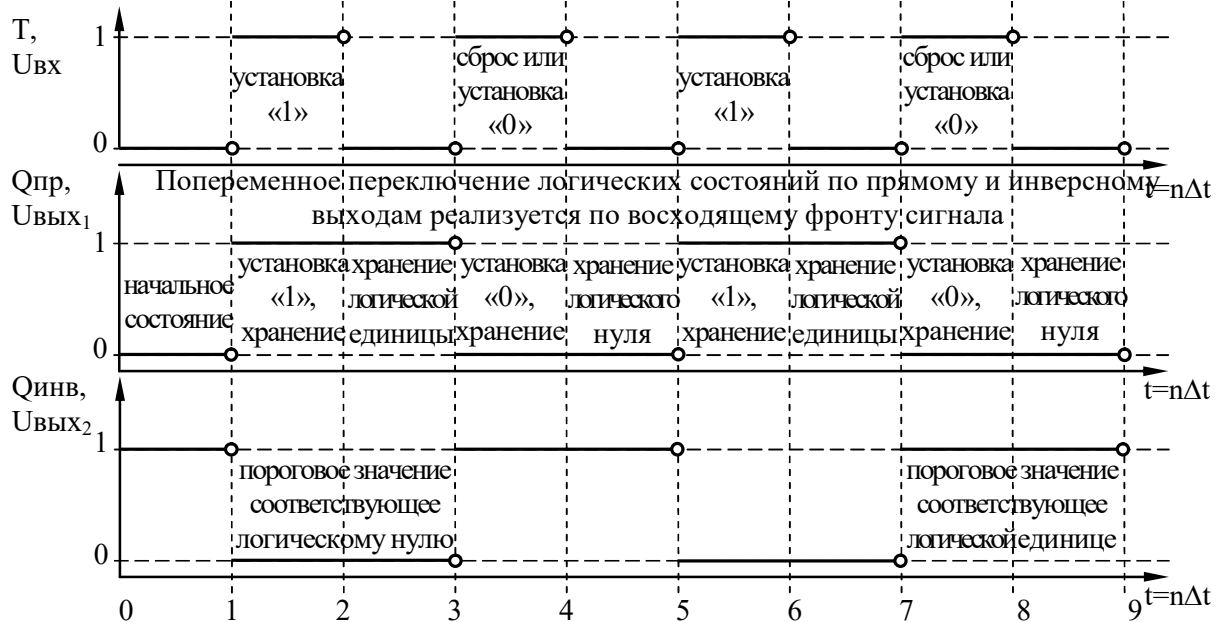


Рисунок 3.37. Временная диаграмма динамики изменения состояний

на выходах Т-триггера в зависимости от уровней сигнала на его входах

Рассмотрим динамику изменения уровней сигналов, которые характеризуют достигнутые логические состояния (0 и 1) на выходах Т-триггера под влиянием входного воздействия, а затем прокомментируем полученную временную диаграмму:

- на первом графике изображена динамика изменения состояния уровня напряжения ($U_{вх}$) на управляющем входе (Т) представленного логического устройства (рис. 3.36), при этом достигается возможность переключения логических состояний на прямом и инверсном выходах тактируемого триггера:
 - начальное логическое состояние на прямом выходе ($Q_{пр}$) триггера равно логическому нулю (пороговое значение уровня напряжения равно 0В), а на инверсном выходе ($Q_{инв}$) тактируемого триггера начальное состояние равно логической единице (пороговое значение уровня напряжения равно 5В);
 - переключение состояний на выходах ($Q_{пр}$ и $Q_{инв}$) осуществляется попеременно по восходящему фронту уровня напряжения на управляющем входе (Т):
 - осуществляется сброс триггера – установка логическим нулем по прямому выходу триггера (пороговое значение уровня напряжения равно 0В и соответствует состоянию логического нуля на прямом выходе триггера), а по инверсному выходу достигается состояние логической единицы;
 - осуществляется установка триггера единицей – установка логической единицей по прямому выходу (пороговое значение уровня напряжения равно 5В и соответствует состоянию логической единицы на прямом выходе триггера), а по инверсному выходу достигается состояние логического нуля.

- на втором и третьем графиках изображена динамика изменения уровней напряжения ($U_{\text{вых}1}$, $U_{\text{вых}2}$) на прямом ($Q_{\text{пр}}$) и инверсном ($Q_{\text{инв}}$) выходах тактируемого триггера, которые соответствуют установившимся логическим состояниям и также характеризуют внутреннее состояние этого устройства (для определенности положим, что начальное состояние на двух выходах данного логического устройства равно $Q_{\text{пр}0}=0$, $Q_{\text{инв}0}=1$, что позволяет в дальнейшем рассчитать установившиеся логические состояния на двух выходах тактируемого триггера после подачи соответствующего входного воздействия);
 - на интервале между нулевым и первым дискретными моментами времени ($t|_{n=0,1} = n\Delta t$) пороговое значение напряжения ($U_{\text{вх}}$) на входе триггера T соответствует состоянию логического нуля (0), поэтому на его прямом выходе $Q_{\text{пр}} = \overline{((Q_{\text{пр}0} \& T) \vee Q_{\text{инв}0})}$ хранится начальное состояние – логический ноль; на его инверсном выходе $Q_{\text{инв}} = \overline{(Q_{\text{пр}0} \vee (T \& Q_{\text{инв}0}))}$ хранится начальное состояние – логическая единица (маркер в форме кружка на каждом графике означает исключение точки соответствующей логическому состоянию в момент переключения логического устройства при $n=0, 1, 2, \dots, k$);
 - на интервале между первым и вторым дискретными моментами времени ($t|_{n=1,2} = n\Delta t$) пороговое значение напряжения ($U_{\text{вх}}$) на входе триггера T соответствует состоянию логической единицы (инициирует безусловное переключение логического состояния триггера на противоположное), поэтому на прямом выходе триггера $Q_{\text{пр}} = \overline{((Q_{\text{пр}0} \& T) \vee Q_{\text{инв}0})}$ достигается состояние логической единицы, а на инверсном выходе триггера $Q_{\text{инв}} = \overline{(Q_{\text{пр}0} \vee (T \& Q_{\text{инв}0}))}$ достигается состояние логического нуля (оба достигнутых логических состояния по двум выходам тактируемого триггера меняются посредством входного воздействия и начинают сохраняться);
 - на интервале между вторым и третьим дискретными моментами времени ($t|_{n=2,3} = n\Delta t$) пороговое значение напряжения ($U_{\text{вх}}$) на входе триггера T соответствует состоянию логического нуля (0), поэтому на его прямом выходе $Q_{\text{пр}} = \overline{((Q_{\text{пр}0} \& T) \vee Q_{\text{инв}0})}$ хранится предыдущее состояние – логическая единица; на его инверсном выходе $Q_{\text{инв}} = \overline{(Q_{\text{пр}0} \vee (T \& Q_{\text{инв}0}))}$ хранится предыдущее состояние – логический ноль (оба достигнутых логических состояния по двум выходам тактируемого триггера не меняются непосредственно после подачи входного воздействия и продолжают далее сохраняться);
 - на интервале между третьим и четвертым дискретными моментами времени ($t|_{n=3,4} = n\Delta t$) пороговое значение напряжения ($U_{\text{вх}}$) на входе триггера T соответствует состоянию логической единицы (1), поэтому на прямом выходе триггера $Q_{\text{пр}} = \overline{((Q_{\text{пр}0} \& T) \vee Q_{\text{инв}0})}$ достигается состояние логического

- нуля (0), а на инверсном выходе триггера $Q_{ИНВ} = \overline{(Q_{ПР_0} V (T \& Q_{ИНВ_0}))}$ достигается состояние логической единицы (оба достигнутых логических состояния по двум выходам тактируемого триггера меняются посредством поданного входного воздействия и начинают сохраняться до их последующего изменения);
- на интервале между четвертым и пятым дискретными моментами времени ($t|_{n=4,5} = n\Delta t$) пороговое значение напряжения ($U_{вх}$) на входе триггера Т соответствует состоянию логического нуля (0), поэтому на его прямом выходе $Q_{ПР} = \overline{((Q_{ПР_0} \& T) V Q_{ИНВ_0})}$ хранится предыдущее состояние – логический ноль; на его инверсном выходе $Q_{ИНВ} = \overline{(Q_{ПР_0} V (T \& Q_{ИНВ_0}))}$ хранится предыдущее состояние – логическая единица (оба достигнутых логических состояния по двум выходам тактируемого триггера не меняются после подачи входного воздействия и продолжают сохраняться до их изменения);
 - на интервале между пятым и шестым дискретными моментами времени ($t|_{n=5,6} = n\Delta t$) пороговое значение напряжения ($U_{вх}$) на входе триггера Т соответствует состоянию логической единицы (1), поэтому на его прямом выходе $Q_{ПР} = \overline{((Q_{ПР_0} \& T) V Q_{ИНВ_0})}$ устанавливается и начинает храниться состояние логической единицы; на его инверсном выходе $Q_{ИНВ} = \overline{(Q_{ПР_0} V (T \& Q_{ИНВ_0}))}$ устанавливается и начинает храниться состояние логического нуля (оба достигнутых логических состояния по двум выходам тактируемого триггера меняются посредством поданного входного воздействия и начинают сохраняться);
 - на интервале между шестым и седьмым дискретными моментами времени ($t|_{n=6,7} = n\Delta t$) пороговое значение напряжения ($U_{вх}$) на входе триггера Т соответствует состоянию логического нуля (0), поэтому на его прямом выходе $Q_{ПР} = \overline{((Q_{ПР_0} \& T) V Q_{ИНВ_0})}$ хранится предыдущее состояние – логическая единица; на его инверсном выходе $Q_{ИНВ} = \overline{(Q_{ПР_0} V (T \& Q_{ИНВ_0}))}$ хранится предыдущее состояние – логический ноль (оба достигнутых логических состояния по двум выходам тактируемого триггера не меняются после подачи входного воздействия на вход Т и продолжают сохраняться);
 - на интервале между седьмым и восьмым дискретными моментами времени ($t|_{n=7,8} = n\Delta t$) пороговое значение напряжения ($U_{вх}$) на входе триггера Т соответствует состоянию логической единицы, поэтому на прямом выходе триггера $Q_{ПР} = \overline{((Q_{ПР_0} \& T) V Q_{ИНВ_0})}$ достигается установка и хранение состояния логического нуля, а на инверсном выходе триггера $Q_{ИНВ} = \overline{(Q_{ПР_0} V (T \& Q_{ИНВ_0}))}$ достигается установка и хранение состояния логической единицы (оба достигнутых логических состояния по двум выходам тактируемого триггера меняются посредством входного воздействия и начинают сохраняться);

- на интервале между восьмым и девятым дискретными моментами времени ($t|_{n=8,9} = n\Delta t$) пороговое значение напряжения ($U_{вх}$) на входе триггера Т соответствует состоянию логического нуля, поэтому на прямом выходе триггера $Q_{пр} = ((Q_{пр_0} \& T) \vee Q_{инв_0})$ продолжает храниться состояние логического нуля, а на инверсном выходе триггера $Q_{инв} = (Q_{пр_0} \vee (T \& Q_{инв_0}))$ продолжает храниться состояние логической единицы (оба достигнутых логических состояния по двум выходам тактируемого триггера не меняются после подачи входного воздействия и продолжают далее сохраняться).

Таким образом, в результате анализа представленной вычислительной процедуры формирования результата на выходах тактируемого триггера ($Q_{пр}$ и $Q_{инв}$) на основе значений входных воздействий (Т), выделим несколько важных выводов:

- аналитическое выражение (система логических уравнений) позволяет обеспечить расчет установившихся значений достигнутых на выходах триггера ($Q_{пр}$ и $Q_{инв}$);
- на основе системы логических уравнений формируется таблица истинности, которая задает однозначное соответствие между бинарными результирующими значениями на выходах тактируемого триггера и последовательностью бинарных значений поступающих на его единственный управляющий вход;
- функциональная схема логического устройства (тактируемого триггера) позволяет спроектировать его аппаратную реализацию посредством определенной элементной базы (интегральные микросхемы, транзисторы и резисторы);
- аналитическое представление (формула), таблица истинности и функциональная схема тактируемого триггера позволяют построить временную диаграмму состояний, которая отражает динамику изменения состояний достигнутых на выходах этого логического устройства на основе значений входных воздействий представленных в виде уровней напряжения соответствующих определенным логическим состояниям: логическая единица или логический ноль;
- аналитическое выражение (система логических уравнений), таблица истинности, временная диаграмма состояний и функциональная схема тактируемого триггера позволяют спроектировать и реализовать электрическую принципиальную схему и аппаратную реализацию тактируемого триггера на основе полупроводниковой элементной базы, которая позволяет вначале собрать и затем изучать принцип функционирования данного логического устройства.

Сделаем важный вывод: для проектирования цифрового автомата, математическая модель которого описывается системой логических уравнений, необходимо иметь функциональные схемы всех логических устройств в его основе, а также полную электрическую принципиальную схему всего компьютера (ЭВМ) и топологию информационной системы с расположением всех узлов и компонентов (компьютеры разной архитектуры и конфигурации, периферийное оборудование, оконечное оборудование и аппаратура передачи данных, программное обеспечение).

4. Генезис и теоретические основы построения информационных вычислительных систем и компьютеров

Зарождение информационных систем связывают с появлением новаций в области информационных технологий (ИТ), созданием инновационных средств автоматизации нового поколения: аппаратного, программного и алгоритмического обеспечения.

Выделяют компоненты информационных систем (ИС) на базе традиционных и ИКТ:

- информация, данные и информационные ресурсы – БД, банки данных, хранилища;
- средства автоматизации (аппаратное, программное и алгоритмическое обеспечение) на основе коммуникационных и ИТ – реализуют сбор, получение, накопление, обработку, передачу информации разного вида и назначения;
- основные средства предприятия – здания, сооружения, машины и оборудование;
- персонал – штатные и привлеченные по договорам сотрудники, которые обеспечивают управление сбором, накоплением и обработкой информации;
- нематериальные активы – патенты, лицензии, свидетельства, фирменные ноу-хау.

Информация – это сведения, представленные в документах и массивах информации на машинных носителях. Информация ИС отвечает на вопрос «что?» и позволяет создавать информационные ресурсы. Информация выступает объектом преобразования: регистрация и сбор, обработка, хранение и передача данных. Информация позволяет обеспечить анализ состояния объекта в среде и системе управления, а затем выработать адекватные управляющие и корректирующие воздействия.

В современном постиндустриальном или информационном обществе информация выступает важнейшим стратегическим ресурсом, поэтому ИС различного назначения разрабатываются с учетом потребительских свойств информации:

- полнота предоставляемой информации, которая описывает вектор состояния объекта управления для быстрой выработки управляющих воздействий;
- точность и достоверность информации для отражения динамики ситуации;
- актуальность, своевременность и оперативность получения и передачи информации предназначенной для узкого и широкого круга потребителей.

В системах управления документооборотом документ является основным носителем информации и состоит из логически связанных структурных элементов.

Форма или макет документа определяет особенности его расположения и использования в ИС, а также формат представления значений различных реквизитов.

В частности, типовая форма определенного документа содержит ряд элементов:

- заголовочную – содержит различные грифы, идентификаторы принадлежности документа определенной организации, допускает наличие штрих-кода;
- содержательную – включает перечень разделов, подразделов и их описание;
- заключительную – имеет подписи лиц, которые согласовали или утвердили его.

Сеть Интернет – наиболее популярная глобальная сеть, объединяющая в себе многие глобальные, региональные и ЛВС, выступает коммуникационной средой для поиска и обмена информацией, электронной корреспонденцией, поддерживает распространение «заплаток» и программных модулей для модернизации ПО.

4.1. Понятие информационной системы

В современной теории информации понятие «система» широко распространено и имеет множество смысловых значений и различное наполнение, но в данной главе основной акцент делается на аппаратное и программное обеспечение ПЭВМ.

Чаще всего с понятием «техническая система» связывают аппаратное обеспечение в основе современной архитектуры компьютера – средства обработки данных.

Понятие «комплекс программ» подчеркивает структурное единство программных компонентов, которые обеспечивают решение широкого круга задач.

Прикладное программное обеспечение допускает включение совокупности программных модулей (компонентов), которые обеспечивают решение частных задач:

- автоматизация процесса документооборота на предприятии или в организации;
- реализация мониторинга за состоянием сложных технологических процессов (ТП);
- управление объектом управления на основе значений параметров состояния;
- автоматизированная математическая обработка апостериорных данных посредством использования совокупности методов статистического анализа;
- моделирование и выявление тенденций, зависимостей и закономерностей функционирования сложных динамических объектов, процессов и явлений;
- формирование информационных БД, хранилищ и банков данных, позволяющих накапливать и получать актуальную информацию по предметной области;
- реализация возможности доступа к распределенному хранилищу информации;
- автоматизированная систематизация, классификация и сохранение данных;
- обеспечение возможности поиска информации, адресное распределение данных;
- создание программных средств для анализа и обработки полученной информации.

Определение

Система – любой объект живой или неживой природы, который одновременно рассматривается и как единое целое, и как совокупность разнородных элементов, объединенная для достижения поставленной цели и выполнения набора задач.

Характерными признаками (свойствами) любой системы являются следующие:

- состоит из совокупности элементов выполняющих различные функции;
- элементы системы представляют собой функциональное единство со связями;
- вхождение каждого элемента и выполняемая им функция не является случайной.

Определение

Информационная система – взаимосвязанная совокупность средств и оборудования, методов и технологий, персонала, которые используются для получения, обработки, хранения, поиска и адресного распределения информации необходимой для достижения поставленной цели (производство товаров, выполнение работ, оказание услуг, в частности: обучение, консультирование, аудит, страхование).

Миссия ИС – производство необходимой для организации информации, обеспечение эффективного управления всеми ресурсами, создание информационной и технической среды для осуществления управления подразделениями организации.

Системы значительно отличаются между собой как по составу так и по главным целям, рассмотрим классификацию и особенности различных систем (табл. 4.1.):

Таблица 4.1

Классификация систем различного вида и назначения

Социализация (повышение уровня социальной организации)	Наименование системы	Элементы системы	Главная цель системы	Информатизация (повышение уровня аппаратных и программных средств)
	Фирма как организационная структура	Люди, машины и оборудование, материалы, здания и сооружения, др.	Производство товаров, выполнение работ, оказание услуг	
	Цифровой автомат (компьютер)	Электронные и электромеханические элементы и др.	Обработка данных	
	Телекоммуникационная система	Компьютеры, модемы, линии связи, сетевое программное обеспечение и др.	Передача информации	
	Информационная система	Все перечисленное	Производство и обработка информации	

Существует большое количество критериев классификации ИС, в частности: по субъектам, происхождению, предметным областям, особенностям использования, целям, задачам, принципам и алгоритмам функционирования, полученным результатам.

По биологическому принципу происхождения выделяют следующие типы систем и особенности реализации информационного взаимодействия в них (рис. 4.1).

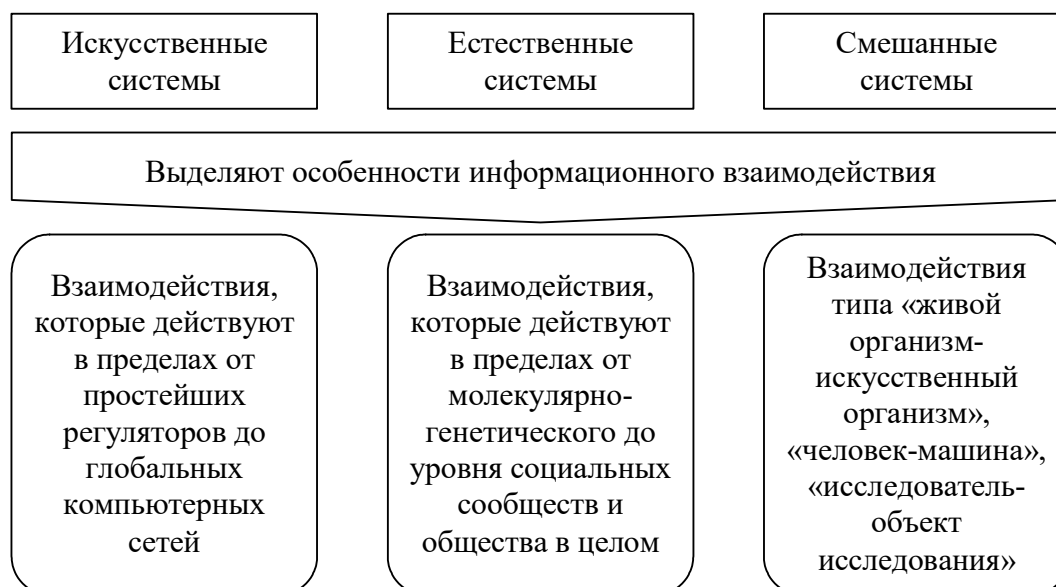


Рис. 4.1. Особенности информационного взаимодействия по типу систем
Информация о системе выступает сущностью, которая регистрируется при вычислимом изоморфизме объекта исследования, процесса или явления посредством использования технических средств наблюдения за динамикой его функционирования: датчики, аналогово-цифровые преобразователи, контроллеры и ЭВМ.

По вовлеченным субъектам ИС дифференцируются следующим образом:

- научные – системы для имитационного моделирования и исследования сложных детерминированных (определенных) и стохастических (случайных) объектов, процессов и явлений в разных средах: статические, динамические;
- астрономические – распределенные компьютерные системы слежения за состоянием подвижных и геостационарных космических объектов и явлений;
- физические – системы создания структурных формул и моделирования течения сложных химических реакций: создание структурных и электронных формул, анализ экзотермической реакции термоядерного синтеза в атомном реакторе;
- обучающие – системы автоматизированного и дистанционного обучения;
- бухгалтерские – системы комплексного бухгалтерского учета результатов финансово-хозяйственной деятельности и всех осуществленных операций;
- финансовые – системы автоматизации создания бизнес-планов и анализа;
- маркетинговые – системы разработки маркетинговых стратегий и планирования мероприятий, которые ориентированы на стимулирование сбыта (продаж);
- аудиторские – системы поддержки принятия решений и осуществления внешнего и внутреннего аудита сформированной финансовой и бухгалтерской отчетности для анализа эффективности финансово-хозяйственной деятельности предприятия (организации) и анализа корректности ведения учетной политики.

По мнению академика Д.А. Поспелова: «любая ИС и интеллектуальная система структурно декомпозируется на совокупность агентов, взаимодействующих в разных средах»:

По агентам как интеллектуальным носителям функционирующим в разных средах:

- последовательный агент – поведение основано на отработке предустановленной последовательности действий согласно линейному алгоритму;
- комбинационный агент – поведение основано на отработке разветвленной последовательности ядер продукции в основе сходящегося (прямой вывод) и расходящегося (обратный вывод) решающего дерева, при этом каждое продукционное ядро эквивалентно элементарному правилу, которое включает:
- антецедент – условие, которое основано на логической модели (Булева алгебра): исчисление предикатами и высказываниями первого и второго порядка;
- консеквент – содержит две процедуры (оператора): если условие выполняется, то выполняется первое (прямое) действие, а иначе альтернативное действие;
- целенаправленный агент – поведение основано на рекомбинации набора правил с целью достижения определенной цели в процессе выработки решения;
- целевыбирающий агент – поведение основано на обходе решающего дерева с целью подбора оптимального пути для достижения основной или одной из альтернативных целей в процессе выработки решения программной системой.

По предметным областям ИС подразделяются на несколько видов: статические, динамические, детерминированные, стохастические. В каждой предметной области между определенными объектами и субъектами выделяют разные уровни взаимодействия.

При рассмотрении взаимодействия в системах естественного происхождения:

- микроскопический уровень рассмотрения системы и всех ее элементов:
 - атомный – совокупность атомов одного или различных химических элементов определяет физические свойства вещества или материи: структурной единицей для анализа выступает атом, а атомы и группы атомов взаимно влияют друг на друга в основе химического элемента или соединения;
 - молекулярный – совокупность молекул определяют химические свойства вещества или материи: атомы или группы атомов упорядочиваются в последовательности согласно их валентности при вступлении в соединение;
- уровень рассмотрения разных веществ и соединений, а также их компонентов:
 - химический – химические свойства вещества определяют возможность вступления в соединения и потенциальный набор продуктов реакций;
 - физический – определяются физические, в частности аморфные свойства тела: цвет, вес, удельная плотность, теплоемкость, теплопроводность и прочие;
- уровень рассмотрения биологических организмов и социальных сообществ:
 - биологический – поведение простейших: инфузория туфелька, эвглена зеленая, амеба;
 - социальный – семья, школа, коллектив, сообщество, социальная группа;
 - общественный – отношения между членами общества, мирового сообщества;
- макроскопический уровень наблюдения за крупными объектами исследования:
 - гравитационный – любое тело, которое имеет массу вступает в гравитационное взаимодействие с другими телами обладающими инертными свойствами;
 - планетарный – планеты находятся в гравитационном взаимодействии между собой, вводится система координат и система взаимного расположения, в частности для Солнечной системы, принято считать, что все планеты нумеруются и вращаются на относительно стабильных орбитах вокруг Солнца.

При рассмотрении разных уровней взаимодействия в технических системах:

- физический – определяется внешними факторами и каналом передачи данных;
- канальный – определяется сетевыми концентраторами и топологией сети;
- сетевой – ограничивается параметрами используемого сетевого оборудования;
- протокольный (транспортный) – определяется особенностями используемого протокола передачи информации в сети с определенной топологией организации;
- сеансовый – реализует распределение сеансов взаимодействия между различными потребителями данных, характеризуется спецификаторами доступа к данным и идентификатором определяющим категорию пользователя;
- представительский – реализуется набором процедур осуществляющих регистрацию в системе, передачу данных между полями интерфейсной формы и источниками данных, а также используемыми методами доступа к БД;
- интерфейсный – определяется технологиями реализации интерфейса на уровне ПО, посредством которого пользователь решает набор разных задач.

Для изучения и уточнения особенностей определенного уровня взаимодействия необходимо опираться на специальные научные концепции, которые приняты в различных предметных областях, ковариантных предмету и объекту исследования.

4.2. Основные виды информационных систем

ИТ направлены на оптимизацию и повышение эффективности деятельности социальных субъектов в различных предметных областях посредством внедрения средств автоматизации разного назначения, которые обеспечивают создание, распределение и использование информационных ресурсов, продуктов и услуг.

Вид ИТ зависит от динамически изменяющихся потребностей дифференцированных потребителей, особенностей процесса информационного обмена между социальными субъектами и техническими средствами, а также обусловлен структурой потоков информации и документооборота сложившихся в предметной области.

Современные ИТ обеспечивают автоматизацию процессов сбора, систематизации, обработки, накопления, поиска и распространения информации разного вида.

Для получения необходимой информации потребитель должен обратиться к ее источникам с помощью ИС. Доступ к внешним источникам информации обеспечивается посредством использования программного обеспечения, которое реализует взаимодействие через локальные вычислительные сети, каналы и магистрали, а обращение к внутренним источникам реализуется при помощи собственных компонентов и ИС.

Решение того или иного круга задач (обработка данных, управление, поддержка принятия решения) требует активизации тех или иных компонентов информационной среды, которые обладают определенным уровнем автоматизации.

Основные виды современных ИТ, которые реализуют информационный обмен:

- ИТ получения данных – реализуют статическое (единовременное), квазидинамическое (систематическое) и динамическое (постоянное) измерение совокупности параметров характеризующих состояние объекта наблюдения посредством набора технических средств измерения (датчик, осциллограф, вольтметр);
- ИТ поиска информации – оптимизирует поиск по определенному шаблону или ключу и информационных ресурсов содержащих необходимую информацию, а также накопление, структурирование, систематизацию различных данных;
- ИТ обработки данных – аппаратно-программный комплекс реализующий извлечение (прием), преобразование, сохранение (передачу) информации;
- ИТ хранения данных – представляет собой совокупность БД, банков и хранилищ данных предназначенных для обеспечения сохранения и извлечения информации разного назначения ориентированной на конечного потребителя;
- ИТ управления и мониторинга – набор технических средств автоматизации реализующих выработку корректирующих воздействий и непрерывное наблюдение за динамикой изменения состояния объекта, процесса или явления;
- ИТ поддержки принятия решений – набор технических средств систематизации и накопления информации отражающей набор фактов и закономерностей (декларативная основа), позволяющих реализовать объективный вывод и поддержку принятия решений посредством набора ядер продукции (процедур).

Для решения глобальных (комплексных) и локальных (частных) задач разрабатываются ИС разного назначения, уровня интеграции, ориентированные на решение широкого круга задач в разных проблемных сферах и предметных областях.

Классификация ИС способствует выявлению их наиболее характерных черт, которые обеспечивают лучшее понимание их функционального назначения и особенностей.

По уровню интеграции и сложности решаемых задач цифровые автоматы делятся на:

- автономные – реализуют функции мониторинга и локального регулирования;
- малой интеграции – реализуют управление роботами и манипуляторами ТП;
- средней интеграции – автоматизированные системы управления технологическими процессами на уровне крупного многономенклатурного предприятия;
- высокой интеграции – обеспечивают решение стратегических задач, требующих больших вычислительных мощностей аппаратного обеспечения.

В нашем случае важнейшим классификационным признаком ИС является ее масштаб и интеграция на уровне используемых компонентов, узлов и интерфейсов:

- локальное рабочее место – программно-технический комплекс, который предназначен для реализации управленческих функций на отдельном компьютере, при этом он информационно и функционально не связан с другими;
- вычислительный комплекс – совокупность аппаратного, программного и алгоритмического обеспечения установленного на компьютере для реализации автоматизации выполнения ограниченного набора функций и задач разных пользователей, которые выполняют определенную работу на предприятии;
- вычислительная сеть – совокупность связанных между собой ЭВМ на единой информационной базе, позволяющей обеспечить интеграцию функций обмена информацией и управления в масштабе предприятия, его административно подчиненных подразделений решающих определенный круг задач;
- корпоративная система – совокупность аппаратного, программного и алгоритмического обеспечения, а также коммуникационного оборудования и средств связи на основе достижений в области ИКТ, которые реализуют полнофункциональное распределенное управление крупномасштабным предприятием посредством использования технологий ERP (Enterprise Resource Planning);
- система прогнозирования погодных явлений – реализует систематический сбор данных с метеорологических зондов и станций для прогноза погоды;
- система глобального позиционирования GPS (General Positioning System) – обслуживается и модернизируется NASA (National Aero-Space Agency), а также рядом научно-исследовательских институтов США, включает совокупность спутников расположенных на геостационарных орбитах планеты Земля и обеспечивает определение месторасположения подвижного объекта в географическом регионе посредством специальной мобильной приставки;
- геоинформационная система – аппаратно-программный комплекс, который обеспечивает решение стратегически важных задач министерства обороны и включает несколько вычислительных систем реализующих частные функции.

4.2.1. Информационная технология получения данных

ИТ сбора и получения данных (Data mining) обеспечивает оптимизацию поиска, систематизации и накопления информации по определенной предметной области.

В современном постиндустриальном или информационном обществе информация выступает важнейшим стратегическим ресурсом, поэтому ИС различного назначения разрабатываются с учетом всех потребительских свойств информации:

- полнота предоставляемой информации, которая описывает вектор состояния объекта управления для быстрой выработки управляющих воздействий;
- точность и достоверность информации для отражения динамики ситуации;
- актуальность, своевременность и оперативность получения и передачи информации предназначенной для узкого и широкого круга потребителей.

Существует несколько способов получения данных от объекта исследования:

- поиск необходимого первичного (объект исследования, эксперт) или вторичного (описание объекта исследования, его модель или книга) источника информации, которые позволяют быстро провести анализ выбранного объекта исследования;
- создание модели выбранного объекта исследования и проведение имитационного моделирования с использованием пакетов программного обеспечения.

При этом бланк для регистрации апостериорных данных измерения набора параметров является основным носителем информации в ИС, он состоит из логически связанных информационных полей, которые образуют стандартный шаблон или форму.

Форма, шаблон или макет бланка для регистрации апостериорных данных эксперимента определяет фиксированное расположение его элементов (полей) и формат представления значений измеренных параметров состояния объекта наблюдения.

Типовая форма бланка для регистрации апостериорных данных содержит:

- заголовочную часть – содержит наименование бланка и реквизиты организации, данные экспериментатора и испытуемого, а также допускает наличие штрих-кода для автоматизации процесса учета и обработки имеющихся данных;
- содержательную – включает перечень экзаменационных вопросов и варианты ответа (для традиционных тестов), которые сегментированы по разделам и подразделам, идентификаторы параметров оценки и единицы измерения (для автоматизированных тестов), описание и вспомогательную информацию;
- заключительную – включает перечень выводов и рекомендаций, подписи ответственных лиц и испытуемого, которые согласовали или утвердили его.

Немаловажным является организация процесса исследования параметров реального объекта или подготовка процедуры тестирования контингента испытуемых:

- подбор технических средств измерения и их установка на реальном объекте или настройка диагностического комплекса для тестирования испытуемых;
- разработка и тиражирование бланков для регистрации апостериорных данных;
- формирование групп испытуемых и подготовка средств измерения параметров;
- проведение процедуры исследования или тестирования и сбор апостериорных данных.

4.2.2. Информационная технология обработки данных

ИТ обработки данных предназначена для преобразования предварительно структурированных данных посредством использования аппаратного и программного обеспечения реализующего процесс обработки данных разного назначения.

Технологии этого рода повсеместно применяются в целях автоматизации рутинных и постоянно повторяющихся операций, которые выполняются персоналом.

На уровне операционной деятельности решаются следующие важные задачи:

- реализуется анализ особенностей имеющихся информационных потоков, выделяются источники информации и нормативные документы, которые отражают специфику документооборота в организационной структуре, выявляются основные закономерности и этапы выработки управленческих решений;
- формируются процессные диаграммы, отражающие потоки движения денежных средств, товарно-материальных ценностей, сырья и готовой продукции, а также решений вырабатываемых административно-управленческим и обслуживающим персоналом в организации, формализованные в виде документов;
- выделяется набор функций и задач, которые возлагаются на основной и вспомогательный персонал согласно их должностным инструкциям, разрабатываются стандартные шаблоны, которые позже используются в ОС для разграничения прав доступа к внутренним и внешним информационным ресурсам;
- подбирается стандартное и разрабатывается специализированное ПО, которое реализует автоматизацию операций выполняемых персоналом на ЭВМ расположенных в разных подразделениях организационной структуры предприятия;
- создается план, который обеспечивает внедрение, конфигурирование, модернизацию и использование различных компонентов информационной среды предприятия, включая топологию расположения компьютеров персонала в отделах;
- внедряется топология ЛВС организации, которая предполагает сетевое взаимодействие территориально распределенных ЭВМ персонала на нескольких иерархических уровнях обмена данными с разграничением прав доступа;
- устанавливается и модернизируется аппаратура передачи данных в сети: сетевые адаптеры, концентраторы, мосты, усилители, коммутаторы и прочее;
- закупка технологического оборудования разного назначения для оснащения и переоснащения сборочных линий, а также разработка плана поставки быстро изнашиваемых и взаимозаменяемых компонентов конвейера, которые полностью или частично изнашиваются на протяжении производственного цикла;
- подбор источников информации, установка датчиков и устройств регистрации информации для поддержки процесса сбора и обработки данных производства;
- обработка данных об операциях, производимых организационной структурой;
- создание периодических контрольных отчетов о динамике функционирования;
- выявление статистических зависимостей и оформление их в виде отчетов.

Постановка и осуществление бухгалтерского учета и документооборота на предприятии требует разработки и использования модели учета основанной на рабочем плане счетов, что инициирует внедрение средств автоматизации, которые реализуют оптимизацию регистрации осуществляемых финансово-хозяйственных операций посредством формирования набора проводок на основе документов-оснований.

Примером финансово-хозяйственных операций сопровождаемых бухгалтерскими проводками выступают: начисление заработной платы, оплата услуг поставщикам, а примером документов служат ежедневный отчет о поступлениях и выдачах наличных средств банком, формируемый в целях контроля баланса наличных средств, или же запрос к БД по кадрам, который позволит получить разные данные о требованиях, предъявляемых к кандидатам на занятие определенной должности.

Существует несколько особенностей, связанных с реализацией обработки данных, которые отличают рассмотренный вид ИТ среди всех представленных:

- анализ содержания процессных диаграмм отражающих последовательность выполнения различных операций контингентом административно-управленческого, основного и вспомогательного персонала, а также технических средств и оборудования поддерживающего процесс сборки узлов, элементов, устройств, агрегатов, изделий и конечных продуктов;
- структурирование декларативных данных отражающих множество состояний, которые потенциально достижимы определенным агрегатом входящим в сборочную линию или конвейер и разработка процедур обеспечивающих переключение между различными состояниями в процессе сборки изделия;
- разработка алгоритмов и программ для поддержки функционирования технологического оборудования автоматизированной сборочной линии: роботов-манипуляторов, агрегатов с числовым программным управлением;
- внедрение и конфигурирование ПО реализующего передачу информации между цехами и отделами предприятия, которая отражает состояние сборочной линии и номенклатурных единиц продукции относящихся к определенной ассортиментной группе на различных технологических заделах производства;
- мониторинг и обработка информации характеризующей состояние и количество изделий выпущенных без дефектов определенным агрегатом сборочной линии, выработка рекомендаций для реконфигурирования программы управления или ремонта различных механических исполнительных механизмов;
- расчет показателей характеризующих результативность (продуктивность) и уровень качества производства различных номенклатурных единиц продукции посредством использования определенной сборочной линии или конвейера;
- выявление «узких мест» сборочной линии, наличие которых характеризуется высокими показателями межоперационных пролеживаний и отбраковки;
- расчет степени автоматизации производства при выполнении основного объема работ в автоматическом режиме с минимальным участием человека.

4.2.3. Информационная технология хранения данных

В современном постиндустриальном обществе информация – это важнейший стратегический ресурс системы управления. Информационные системы разрабатываются с учетом заданных потребительских свойств информации: полнота информации для реализации управляющего воздействия, точность и достоверность информации, актуальность, своевременность и оперативность получения и передачи данных.

Хранение данных предполагает анализ типа информации, структурирование информационных полей и разработку инфологической схемы БД, обеспечивающей сохранение и извлечение данных используемых при работе разных пользователей.

Процедура взаимодействия с БД имеет несколько иерархических уровней:

- уровень интерфейса – представлен набором элементов навигации и управления между записями в БД, а также полей разного типа и назначения, которые реализуют ввод данных пользователем и отображение имеющихся данных;
- уровень приложения – представлен набором форм интерфейса позволяющих пользователю работать в различных режимах функционирования приложения, каждая из которых включает наборы различных элементов (поля, навигаторы);
- уровень представления данных – реализован за счет использования различных моделей представления данных обеспечивающих эффективное хранение;
- уровень процедур доступа к данным – представлен набором алгоритмов и процедур, которые реализуют отображение и сохранение данных пользователя содержащихся в информационных полях БД на машинном носителе;
- уровень хранения данных – обеспечивается посредством использования локальных источников данных (текстовые файлы, Dbase for Windows) и сетевых БД (MS Access, Fox Pro, Fire Fox, Oracle, Sybase, Informix, SQL Server).

При работе выделяют данные пользователя и служебные данные приложений.

Данные пользователя представляют собой перечень записей БД, каждая из которых отражает перечень значений информационных полей, содержание которых отображается посредством алгоритмов и доступно для редактирования пользователю.

Служебные данные приложений хранятся в реестре ОС, в файлах и БД, отражают расположение и параметры настройки элементов интерфейса приложения.

Создание отчетов и документов достигается за счет использования средств автоматизации документооборота, но возникает проблема хранения информации. Для ИТ обеспечивающих автоматизацию документооборота необходимо создавать документы для руководства и работников фирмы, а также для внешних контрагентов. При этом документы создаются периодически в конце каждого месяца, квартала или года согласно форме по запросу или могут сопровождать проведенную операцию.

В качестве основной формы организации хранения информации на машинных носителях используется система управления базами данных (СУБД), которая обеспечивает оптимальное сохранение и извлечение информации из нескольких территориально распределенных БД разного профиля и назначения, функционирующих как единая система посредством использования аппаратуры передачи данных.

В основу организации БД положен принцип распределенной архитектуры и реляционная модель, которая предполагает наличие совокупности связанных между собой таблиц содержащих набор информационных полей, некоторые из них выступают ключевыми при реализации различных типов связей между всеми таблицами:

- связь «один к одному» – предполагает взаимнооднозначное соответствие между информационным полем, которое содержится в одной таблице и полем расположенным в другой таблице, выбор записи в одной таблице обуславливает отображение только одной записи хранящейся в другой таблице;
- связь «один ко многим» – предполагает наличие соответствия между значением информационного поля в одной таблице и несколькими информационными полями в другой таблице, при этом выбор определенной записи в главной таблице инициирует отображение перечня записей в подчиненной таблице
- связь «многие ко многим» – предполагает наличие однозначного соответствия между несколькими значениями информационных полей в одной таблице и совокупностью значений информационных полей содержащихся в другой таблице, при этом выбор нескольких записей в одной таблице обуславливает отображение множества записей, значения которых хранятся в другой таблице БД.

Для наглядной интерпретации процесса конструирования и реализации БД используется инфологическая схема БД определенной структуры, которая зависит от предметной области, количества информационных полей заданного типа и размера.

БД выступает представлением структурированных данных многоцелевого назначения и использования, обеспечивает хранение данных, которые требуются для решения комплекса взаимосвязанных задач в определенной предметной области.

Для снижения транзакционной нагрузки и сетевого трафика между файловым сервером-носителем БД и удаленными ЭВМ пользователей используется распределенная архитектура БД, которая предполагает использование локально расположенных автономных или «изолированных» массивов информации. При этом сеансы взаимного обмена информацией между сервером и несколькими удаленными клиентами осуществляются систематически, направлен на обновление данных хранилища.

СУБД реализует обработку транзакций отражающих последовательность операций направленных на модификацию структуры данных, ввод, поиск, корректировку, удаление и резервирование значений полей, которые содержатся в БД.

БД может быть централизованной и храниться на одном компьютере или распределенной и храниться на нескольких компьютерах, которые объединены в ЛВС.

В настоящее время наибольшее распространение получили следующие СУБД:

- на уровне предприятия – DB2, Oracle, Informix, MS SQL-Server и ряд других;
- на уровне подразделений предприятия – выступают в роли локальных и сетевых БД – MS Access, dBase, Paradox, FoxPro, Clipper и файлы данных;
- на уровне локальных систем управления – специальные ресурсы и прикладные БД для автоматизированных систем управления технологическими процессами.

4.2.4. Информационная технология передачи данных

Вертикально интегрированная организационная структура предприятия предполагает распределенную архитектуру ИС, которая предполагает использование ЛВС реализующей обмен данными между подразделениями, отделами и службами.

Определение

Под **локальной вычислительной сетью (ЛВС)** в информационной системе понимают набор аппаратного обеспечения (оконечное оборудование передачи данных на стороне клиента и сервера), которое реализует процесс обмена информацией между несколькими компьютерами и автоматизированными рабочими местами посредством использования аппаратуры передачи данных (сетевой адаптер, концентратор, коммутатор, усилитель, АТС, точка доступа) через каналы передачи данных (спутниковый радиоканал, сотовый радиоканал, оптоволоконная среда, инфракрасный канал, витая пара, коаксиальный кабель, телефонный кабель, провод).

ЛВС позволяет существенно оптимизировать процесс информационного обмена между источниками и потребителями информации, а также существенно сократить временные и транзакционные издержки, которые неизбежно возникают при ее получении, систематизации, накоплении, поиске, отборе, обработке и отправке.

ИТ передачи данных накладывают определенные ограничения на информацию:

- информация представляется в форме данных, которые доступны для последующей обработки посредством использования разных средств автоматизации: специальное аппаратное, программное и алгоритмическое обеспечение;
- реализуется поиск, обработка и подготовка данных для последующей передачи;
- для хранения создается набор структур данных и инфологическая схема БД;
- разрабатывается комплекс программного обеспечения для отправки содержания используемой БД потребителям по каналам информационного обмена;
- формируется система разграничения прав доступа к системе управления БД, которая предусматривает первичную регистрацию и ввод личных данных, а также последующую аутентификацию пользователей с учетом уровня их доступа;
- реализация возможности сохранения перечня запросов к БД, использование панелей навигации и интуитивно понятного интерфейса взаимодействия между пользователем и системой управления БД, включая набор ссылок и пиктограмм;
- возможность частичного или полного ограничения доступа к БД посредством блокирования нежелательных сетевых соединений и учетных записей некоторых пользователей в случае возникновения соответствующей необходимости;
- использование механизмов архивирования и резервирования содержания информационной базы для исключения случайного и (или) намеренного повреждения данных пользователями или программными средствами (вирусами);
- использование различных механизмов восстановления информации из резервного хранилища, а также из территориально распределенных резервных БД;
- мониторинг и балансировка сетевой нагрузки для распределения потоков данных.

4.3. Зарождение архитектуры компьютеров

Предпосылками зарождения и развития архитектуры ЭВМ выступает становление системного и статистического анализа, алгебры логики (Булевой алгебры).

Алгебра логики основана на исчислении высказываниями и предикатами первого и второго порядка, которые позволяют сформировать простое или сложное логическое выражение: совокупность логических операндов и операций как связок между ними. Использование Булевой алгебры позволяет реализовать проектирование и моделирование принципов функционирования аппаратного обеспечения классических и современных компьютеров, а также разных информационных систем.

Развитие и использование алгебры логики позволяет говорить о бинарной основе функционирования любого компонента цифрового автомата (ЭВМ): логическое выражение включает последовательность логических операндов и операций, а об их истинности или ложности можно достоверно утверждать на основе расчета.

Существует однозначное соответствие между логическими состояниями логических операндов и операций на функциональной схеме логического устройства, а также уровнями сигналов, которые достигаются на входах и выходах аппаратной реализации логического устройства или сложной логической функции (операции).

Для аппаратной реализации компонентов и узлов компьютера необходима соответствующая элементная база электронных компонентов: вначале реле и электронные вакуумные лампы, позже резисторы, транзисторы, диоды и конденсаторы, еще позже интегральные микросхемы малой, средней и большой степени интеграции.

В 1946-1948 году коллектив исследователей под непосредственным руководством Джона фон Неймана (Принстонский университет, США) разработал классическую архитектуру компьютера, которая включала несколько компонентов и алгоритмы их функционирования, но на тот момент разработка не была аппаратно реализована ввиду отсутствия соответствующей элементной базы, тем не менее, основные идеи и принципы в ее основе используются до настоящего времени всеми производителями алгоритмического, программного и аппаратного обеспечения современных ЭВМ и микропроцессорной техники для различных потребителей.

Исторически проект получил название машины фон Неймана или Принстонской машины, а предложенная коллективом ученых архитектура названа классической.

При разработке аппаратной реализации архитектуры компьютеров разных поколений, их узлов и элементов, а также периферийного оборудования (внешних устройств) разработчики придерживаются определенной последовательности:

- разрабатывается функциональная схема и архитектура ЭВМ, которая включает взаимосвязанный набор различных компонентов, узлов и интерфейсов;
- каждому компоненту архитектуры ЭВМ вводятся в соответствие процедуры и алгоритмы, которые реализуют логику функционирования при решении задач;
- создается программное обеспечение, основанное на процедурах и алгоритмах;
- осуществляется аппаратная реализация компонентов архитектуры ЭВМ.

4.4. Развитие архитектуры электронных вычислительных машин

В соответствии с элементной базой и уровнем развития программных средств выделяют несколько поколений ЭВМ, а их краткая характеристика приведена в табл. 4.2.

Таблица 4.2

Развитие архитектуры ЭВМ и ее компонентов в рамках нескольких поколений

Параметры сравнения	Поколения электронных вычислительных машин			
	первое	второе	третье	четвертое
период времени	1946-1959	1960-1969	1970-1979	1980-2000
Элементная база (для устройства управления и АЛУ)	Электрические электронные вакуумные лампы	Полупроводники (транзисторы)	Интегральные схемы	Микросхемы большой степени интеграции
Размер ЭВМ	Большие	Большие	Малые (мини)	Микро
Основные устройства ввода	Пульт, перфокарточный и перфоленточный ввод	Клавиатура	Манипулятор	Сканер, клавиатура, цифровое перо
Основные устройства вывода	Алфавитно-цифровое табло и печатающее устройство, перфокарточный вывод	Алфавитно-цифровое печатающее устройство, перфоленточный вывод	Цветной графический дисплей, графопостроитель, матричный принтер	Жидкокристаллический дисплей, струйный и лазерный принтер
Внешняя память	Магнитные ленты, барабаны, перфоленты и перфокарты	Магнитный диск	Перфоленты, магнитный диск	Магнитные, оптические и электронные диски
Ключевые решения в программном обеспечении	Универсальные языки программирования, компиляторы	Локальные операционные системы, (ре)трансляторы, компиляторы	Интерактивные операционные системы, структурированные языки программирования	Интерфейсы, сетевые и многопользовательские операционные системы
Режим работы ЭВМ	Однопрограммный, последовательный ввод команд, командный интерфейс	Пакетный ввод команд, оболочки для операционных систем	Графический интерфейс, одно- и многопользовательский с разделением во времени	Многопользовательская работа и сетевая обработка данных
Цель использования ЭВМ	Научно-технические расчеты	Технические и экономические расчеты	Управление и экономические расчеты	Телекоммуникации, информационное обслуживание и управление

ЭВМ состоит из большого количества внутренних и внешних компонентов (устройств) в рамках архитектуры, а также выступает основой любой информационной системы, поскольку обеспечивает поддержку процессов обработки информации и структурированных данных в информационных хранилищах, банках и БД.

4.5. Основные принципы функционирования классических компьютеров

К основным принципам функционирования классической вычислительной машины разработанной под руководством Джона фон Неймана относят следующие:

- **принцип программного управления** – работа ЭВМ регламентируется программой, что позволяет, вводить программы для решения разных задач, а компьютер выступает универсальным устройством для решения разных задач:
 - любая программа состоит из последовательности операций над операндами, а также занимает определенный объем в регистровой и оперативной памяти;
 - операции в программе выполняются в строгой линейной последовательности;
 - каждая операция (команда) имеет определенную структуру: [код операции] [операнд₁][операнд₂]...[операнд_n], а количество операндов в определенной операции позволяет выделить: унарную операцию – один операнд; бинарная операция – два операнда; n-арная операция – n операндов;
 - определенный операнд обеспечивает непосредственное хранение значения, ссылку на значение в области памяти, ссылку на значение в другом операнде;
 - код операции взаимно однозначно определяет тип операции, которая выполняется над значениями операндов в арифметико-логическом устройстве;
- **принцип условного перехода** – линейную последовательность выполнения команд (операций) изменяет операция условного перехода, которая реализует ветвление разработанной программистом алгоритмической структуры (алгоритма);
- **принцип размещения программы в памяти** – в процессе работы ЭВМ программа предварительно размещается в памяти посредством использования устройства ввода или с помощью элементов внешней и внутренней памяти;
- **принцип иерархии памяти** – память компьютера организационно не является однородной по структуре и содержанию,- для часто используемых данных выделяется память меньшего объема, но большего быстродействия, а для редко используемых данных выделяется память большего объема, но меньшего быстродействия, при этом специалистами выделяют несколько классов:
 - регистровая память и кэш память – содержит данные, которые востребованы в течении нескольких смежных тактов операционного цикла процессора;
 - оперативная память – содержит данные, которые временно не востребованы и не обрабатываются в рамках операционного цикла процессора;
 - внешняя память – обеспечивает долговременное хранение данных, которые не используются продолжительное время в рамках данного сеанса работы.
- **принцип двоичной системы счисления** – требования и ограничения к элементной базе при осуществлении аппаратной реализации различных компонентов архитектуры компьютера обуславливают необходимость использования двоичной системы счисления для внутреннего представления данных и программ операционной системы и пользователя, которые располагаются во внутренней и внешней памяти в виде последовательностей нулей и единиц.

4.6. Структура и принцип функционирования классической (принстонской) электронной вычислительной машины

Структура компьютера разработанного в Принстонском университете США включает несколько основных элементов, которые представлены на рис. 4.2.

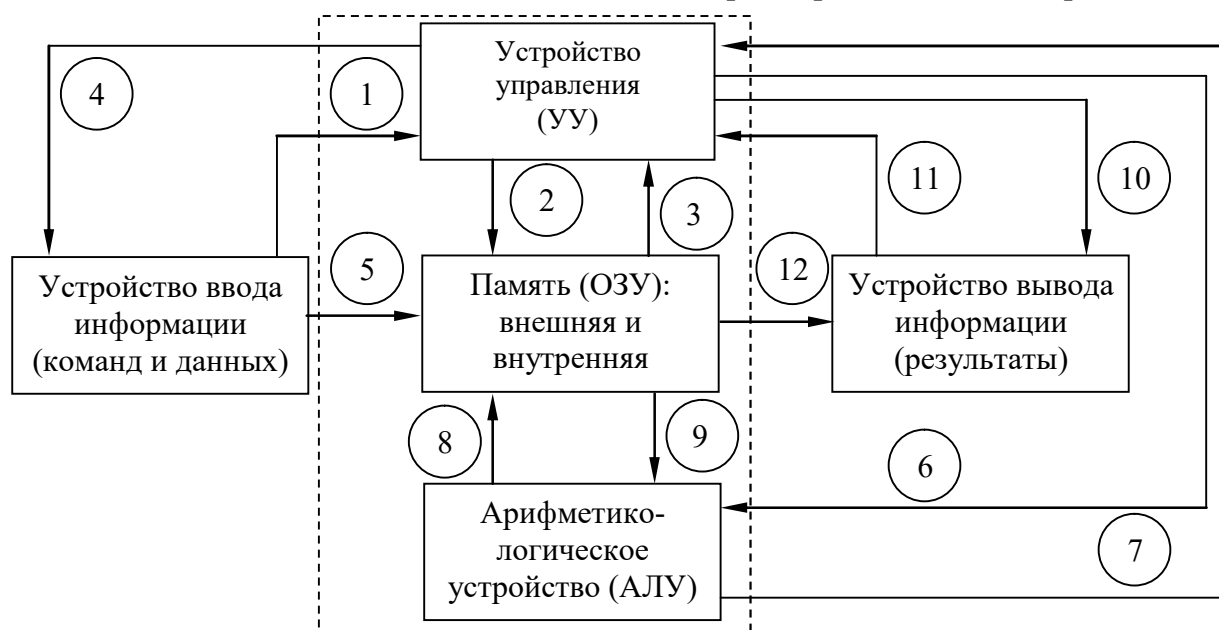


Рисунок 4.2. Схема классической (принстонской) архитектуры компьютера

На представленном рисунке присутствуют несколько элементов, которые взаимосвязаны между собой и реализуют выполнение различных функций и задач.

Определение

Устройство ввода информации (УВв) – реализует ввод команд и данных в ЭВМ.

Определение

Устройство вывода информации (УВыв) – обеспечивает вывод промежуточных и окончательных результатов выполнения команд и данных после их обработки.

Определение

Память (П) – предназначена для хранения промежуточных данных и служебной информации процессора, операционной системы, пользователя в процессе функционирования цифрового автомата при выполнении программы для решения задачи.

Определение

Арифметико-логическое устройство (АЛУ) – позволяет итеративно выполнять арифметические и логические операции над данными (значениями операндов).

Определение

Устройство управления (УУ) – обеспечивает согласованное функционирование всех элементов классической архитектуры ЭВМ в процессе выполнения различных операций, которые связаны с вводом исходных данных, вычислительными процедурами и алгоритмами обработки данных, а также выводом результирующих значений, которые были получены в ходе вычислительного процесса на компьютере.

Коллектив разработчиков классической архитектуры компьютера в Принстонском университете США предложили схематическое представление работы ЭВМ.

Принцип функционирования машины Джона фон Неймана следующий:

1. Любая попытка ручного или автоматизированного ввода данных в ЭВМ инициирует генерацию сигнала о начале ввода данных (операции и значения их операндов) из **Устройства ввода информации в Устройство управления по связи 1**:
 - ручной ввод информации заключается в нажатии пользователем (программистом) последовательности клавиш на клавиатуре, что инициирует возникновение соответствующих событий для последующей обработки компьютером;
 - автоматизированный ввод информации осуществляется за счет использования перфокарт или перфолент, в поверхности которых пробиваются отверстия для коммутации контактов в матрице устройства чтения перфокарт или перфолент;
 - автоматический ввод информации достигается за счет использования различных накопителей информации на разнородных носителях информации.
2. При регистрации факта ввода информации **Устройство управления** генерирует сигнал запроса наличия свободных ячеек памяти в блоке **Память по связи 2**.
3. Если имеются в наличии свободные ячейки памяти в блоке **Память**, то обеспечивается генерация сигнала, который свидетельствует о наличии свободных ячеек памяти и он передается из блока **Память в Устройство управления по связи 3**.
4. При наличии свободных ячеек памяти в блоке **Память Устройство управления** реализует генерацию сигнала о начале процесса перемещения данных (команды и их аргументы) из **Устройства ввода информации** в блок **Память по связи 4**, а данные перемещаются из **Устройства ввода информации в Память по связи 5**.
5. По факту завершения процесса передачи данных из **Устройства ввода информации** в блок **Память (связь 5)** генерируется сигнал окончания процесса перемещения данных из **Устройства ввода информации в Устройство управления по связи 1**, а по факту сохранения данных в ячейках памяти блока **Память** генерируется сигнал о завершении процесса сохранения данных (список операций и значения их операндов) из блока **Память в Устройство управления по связи 3**.
6. При условии наличия данных (сохранен список операций и значений их операндов) для выполнения операций в блоке **Память Устройство управления** генерирует сигнал запроса состояния **Арифметико-логического устройства** для начала выполнения перечня арифметических или логических операций по **связи 6**.
7. В случае свободного состояния **Арифметико-логического устройства** им генерируется сигнал о готовности выполнения последовательности арифметических и(или) логических операций в **Устройство управления по связи 7**.
8. По факту регистрации свободного состояния **Арифметико-логического устройства** **Устройство управления** отправляет в него сигнал о начале процесса выполнения определенной арифметической или логической операции по **связи 6**.
9. Для начала операционного цикла **Арифметико-логическое устройство** генерирует сигнал запроса данных операции (код операции, значение первого операнда, значение второго операнда, значение n-го операнда) в блок **Память по связи 8**.

10. Выбор данных определенной операции из блока **Память** и загрузка всех ее параметров непосредственно в регистры **Арифметико-логического устройства** реализуется на представленной схеме функционирования компьютера по **связи 9**.

11. По факту заполнения **Регистра операции** (регистр кода операции, регистр первого операнда, регистр второго операнда, регистр n-го операнда) **Арифметико-логическое устройство** производит классификацию типа команды и вида операции в зависимости от содержания регистра кода операции в данный момент:

- если операция является операцией ввода данных, то **Арифметико-логическое устройство** генерирует сигнал в **Устройство управления** с целью запроса состояния **Устройства ввода информации** по **связи 7**, а **Устройство управления** запрашивает состояние **Устройства ввода информации** по **связи 4** до получения положительного ответа о возможности ввода данных по **связи 1**, потом **Устройство ввода информации** сохраняет данные в блоке **Память** (**связь 5**) и генерирует ответ **Устройству управления** по **связи 1**, **Устройство управления** передает сигнал наличия данных в ячейках памяти блока **Память** в **Арифметико-логическое устройство** по **связи 6** (рис. 4.2);
- если операция является арифметической или логической, то **Арифметико-логическое устройство** выполняет соответствующую арифметическую или логическую операцию над значениями операндов, содержащимися в регистре первого операнда и регистре второго операнда, а затем размещает результирующее значение операции в регистре результата (в некоторых случаях для формирования результата допустимо использование регистра первого операнда), потом содержимое регистра операции выгружается в блок **Память** по **связи 8** и генерируется сигнал о завершении операции в **Устройство управления** по **связи 7**;
- если операция является операцией вывода информации, то **Арифметико-логическое устройство** генерирует сигнал в **Устройство управления** с запросом состояния **Устройства Вывода информации** по **связи 7**, а **Устройство управления** запрашивает состояние **Устройства вывода информации** по **связи 10** и рано или поздно получает положительный ответ о возможности вывода данных по **связи 11**, потом **Устройство вывода информации** загружает и выводит данные из блока **Память** по **связи 12** и генерирует ответ **Устройству управления** по **связи 11** о завершении вывода данных (результат операции).

12. Вычислительная процедура повторяется циклически до останова компьютера.

Для поддержания непрерывности осуществления вычислительной процедуры в компьютере необходимо обеспечить непрерывность поступления данных (операции и значения их операндов) на устройство ввода информации, гарантировать наличие процедур реализующих обработку различных арифметических и логических операций в арифметико-логическом устройстве, а также реализовать регулярность получения пользователем результирующих данных и возвращаемых значений после выполнения каждой операции посредством использования устройства вывода информации.

Необходимо отметить ряд важных выводов, которые следуют из рассмотрения принципов функционирования классических и современных компьютеров:

- устройство ввода данных функционирует в рамках нескольких режимов:
 - ручной ввод информации в ЭВМ – данные вводятся последовательно оператором, представляют собой значения операндов и операций между ними;
 - автоматический ввод информации – данные вводятся автоматически с внешнего носителя информации: перфокарта, перфолента, магнитная лента, магнитный диск, оптический диск, электронная карта памяти;
- арифметико-логическое устройство и устройство управления выступают прототипом центрального процессора в основе архитектуры современных ЭВМ;
- при ближайшем рассмотрении классической и современной архитектуры ЭВМ память выступает существенно неоднородным компонентом, выделяют:
 - регистровую память, которую использует арифметико-логическое устройство для реализации выполнения арифметических и логических операций, а также для регистрации текущего состояния центрального процессора;
 - кэш память (первого и второго уровней), которая предназначена для промежуточного хранения информации загружаемой из оперативной памяти ЭВМ или данных выгружаемых из регистровой памяти в течении операционного цикла центрального процессора цифрового автомата (ЭВМ);
 - оперативная память обеспечивает первичную загрузку выгружаемых и невыгружаемых компонентов операционной системы, а также текущих данных прикладного программного обеспечения, хранит промежуточные и окончательные результаты обработки данных и событий, которые инициированы пользователем или программным окружением компьютера;
 - внешняя память представляет собой совокупность накопителей информации, которые функционируют на основе носителей информации разного типа;
- устройство вывода данных из ЭВМ реализует несколько способов вывода:
 - визуальное отображение информации – используется алфавитно-цифровое табло, светодиодная панель, дисплей (монитор) или плазменная панель;
 - вывод на бумажный носитель – применяется самописец или устройство для автоматического пробивания поверхности перфокарты, а также принтер;
 - вывод на устройство внешней памяти – применяются: накопитель информации на перфокартах или перфолентах, накопитель информации на магнитных лентах, накопитель информации на гибких или жестких магнитных дисках, лазерный накопитель на оптических дисках, электронный накопитель информации, накопитель информации на электронных картах.

Соответственно классические основы функционирования компьютера претерпели существенное изменение в процессе развития аппаратного, программного и алгоритмического обеспечения, что обусловило появление новых методов, подходов и технологий создания средств автоматизации в различных областях деятельности.

4.7. Элементы неоклассической архитектуры электронной вычислительной машины

Элементы неоклассической архитектуры ЭВМ развивались на основе классической принстонской архитектуры компьютера, которая существенно изменилась.

Определение

Неоклассической архитектурой ЭВМ называют интегральную совокупность взаимосвязанных элементов, которые развивались на основе классической принстонской архитектуры компьютера включающей устройство ввода информации, память, арифметико-логическое устройство, устройство управления и устройство вывода информации, а также функционирующей на основе ряда основополагающих принципов: программное управление, расположение и программы в памяти, неоднородность и иерархичность памяти, условный переход, бинарная система представления данных.

Основные направления развития (нео)классической архитектуры компьютера:

- при изучении процесса эволюции элементов аппаратного обеспечения:
 - минимизация габаритных размеров цифровых автоматов (ЭВМ);
 - переход на новую элементную базу при аппаратной реализации компонентов архитектуры компьютера и используемого периферийного оборудования;
 - применение энергосберегающих технологий, которые позволяют достичь существенно снижения энергопотребления разных компонентов компьютера;
 - применение технологии многослойного печатного монтажа при изготовлении компонентов аппаратного обеспечения цифрового автомата (компьютера);
 - использование полупроводниковых элементов и интегральных микросхем малой, средней и большой степени интеграции в основе компонентов ЭВМ;
- при рассмотрении новаций в области компонентов программного обеспечения:
 - появление языков программирования низкого уровня для реализации микропрограмм в основе базовой системы ввода вывода (Basic Input Output System) на системной плате, прошивок видеокарт, модемов и контроллеров;
 - создание подходов к структурированию и моделированию представления данных;
 - внедрение языков программирования высокого уровня, а также методов и технологий визуальной разработки в программных средах нового поколения;
- при анализе новаций в имеющемся алгоритмическом обеспечении компьютера:
 - разработка представления алгоритмических структур посредством использования блок-схем с условными графическими обозначениями операций;
 - применение программных средств и сред для автоматизации процесса разработки элементов алгоритмического обеспечения компьютера (ЭВМ).

В ходе своего развития классическая (принстонская) архитектура ЭВМ совершенствовалась и усложнялась, что обусловило появление следующих элементов:

- инновационные элементы с возрастающей структурной и функциональной сложностью, а также появление новых технологий их аппаратной реализации;
- появление нового системного и прикладного программного обеспечения.

4.7.1. Структура и принцип действия устройства управления

Устройство управления является структурным компонентом центрального процессора компьютера, а разрядность шины адреса и шины данных выступают его важнейшими характеристиками, определяя разрядность передаваемых машинных команд.

Рассмотрим структуру устройства управления для бинарных команд (рис. 4.3), а также взаимодействие его элементов в процессе реализации операционного цикла.

см. 1, 4, 2, 3, 6, 7, 10, 11 на рис. 4.2.

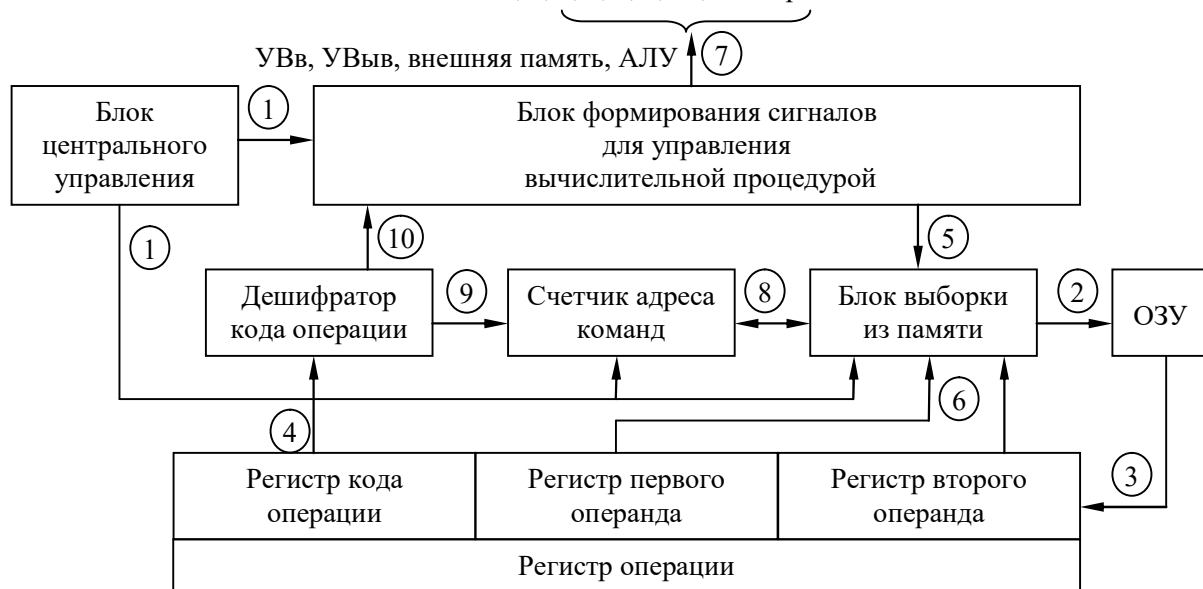


Рисунок 4.3. Структура устройства управления

На рис. 4.3 представлены несколько блоков, имеющих разное назначений:

- блок центрального управления – реализует синхронизацию загрузки и выгрузки информации во всех устройствах компьютера, а также, в частности, выполняет управление блоками арифметико-логического устройства, которые обеспечивают обработку данных и выполнение различных операций;
- блок формирования сигналов для управления вычислительной процедурой – позволяет управлять вычислительным процессом, который реализуется в арифметико-логическом устройстве в основе структуры цифрового автомата;
- дешифратор кода операции – позволяет по значению кода операции взаимно однозначно определить вид операции, которая выполняется в арифметико-логическом устройстве посредством использования встроенного алгоритма;
- счетчик адреса команд – содержит значение адреса операции (команды), которая подлежит выполнению на следующей итерации операционного цикла;
- блок выборки из памяти – обеспечивает непосредственную загрузку и сохранение исходных значений операции и результата выполнения операции;
- память (ОЗУ) – реализует хранение последовательности операций (команд) и значений их операндов, которые подлежат выполнению в компьютере;
- регистр операции позволяет обеспечить временное хранение значений параметров, которые необходимы для осуществления операции и включает: регистр кода операции, регистр первого операнда, регистр второго операнда.

Принцип функционирования устройства управления следующий (рис. 4.3):

1. **Блок центрального управления** генерирует управляющие сигналы к **Блоку формирования сигналов для управления вычислительной процедурой**, а также для синхронизации работы **Счетчика адреса команд** и **Блока выборки из памяти** по **связи 1**.
2. На предварительном этапе **Устройство управления** реализует первичную инициализацию устройств и значений, которые необходимы для выполнения арифметической или логической операции, поэтому анализируется ряд запросов:
 - при попытке ручного или автоматизированного ввода информации регистрируется сигнал о начале ввода данных через **Устройство ввода информации (связь 1 на рис. 4.2)**;
 - **Устройство управления** генерирует сигнал запроса о наличии свободных ячеек памяти для хранения данных операции в блоке **Память (связь 2)**, а затем получает сигнал-ответ о наличии свободных ячеек памяти в блоке **Память (связь 3)**;
 - **Устройство управления** генерирует сигнал о начале перемещения данных (операции и значения их операндов) (**связь 4 на рис. 4.2**) из **Устройства ввода информации** в блок **Память (связь 5 на рис. 4.2)**, а затем по факту завершения процесса передачи данных из **Устройства ввода информации** в блок **Память** и по факту завершения сохранения данных в ячейках памяти блока **Память (Устройство управления** в результате выполнения перемещения соответствующих данных получает два сигнала-ответа по **связям 1, 3 на рис. 4.2)**;
3. **Блок центрального управления** осуществляет обмен запросами с блоком **Память** и **Арифметико-логическим устройством** непосредственно для реализации арифметической или логической операции посредством использования ЭВМ:
 - **Блок центрального управления Устройства управления** генерирует запрос о состоянии **Арифметико-логического устройства** по **связи 6 (рис. 4.2)**, а в случае регистрации факта свободного состояния **Арифметико-логического устройства** он получает от него сигнал-ответ по **связи 7 (рис. 4.2)**;
 - **Блок центрального управления Устройства управления** отправляет в **Арифметико-логическое устройство** сигнал о начале выполнения арифметической или логической операции по **связи 6 (рис. 4.2)** согласно вычислительному алгоритму на основе классификации типа операции, которая реализуется посредством совместного использования **Дешифратора кода операции (связь 1 на рис. 4.3)**;
 - для реализации определенной арифметической или логической операции необходимо иметь в блоке **Память** значения первого и второго операндов;
 - результат операции над двумя значениями операндов непосредственно после отработки арифметико-логического устройства помещается в блок **Память**.
4. **Блок выборки из памяти** генерирует запрос блоку **Память (ОЗУ)** по **связи 2** для загрузки данных обеспечивающих выполнение операции: значение кода операции, значение первого операнда, значение второго операнда, значение n-го операнда.
5. Блок **Память** передает по **связи 3 (рис. 4.3)** данные операции в **Регистр операции** (регистр кода операции, регистр первого операнда, регистр второго операнда, регистр n-го операнда) **Устройства управления** и **Арифметико-логического устройства**.

6. Для реализации классификации типа команды и вида операции значение **Регистра кода операции** передается в **Дешифратор кода операции по связи 4** (рис. 4.3).
7. **Блок формирования сигналов для управления вычислительной процедурой** генерирует сигнал **Блоку выборки из памяти по связи 5** для проверки наличия в **Регистре операции** значения первого операнда, значения второго операнда, значения n-го операнда, которые поступают в него непосредственно по **связи 6**.
8. **Счетчик адреса команд** реализует приращение адреса команды в зависимости от ее типа (**связь 8**): **Дешифратор кода операции** позволяет определить тип операции (**связь 9**), а затем отправить результирующий код в **Блок формирования сигналов для управления вычислительной процедурой по связи 10**:
 - если операция является операцией ввода или вывода, то **Блок формирования сигналов для управления вычислительной процедурой** формирует сигнал **Устройству ввода информации (связь 4 на рис. 4.2)** или **Устройству вывода информации (связь 10 на рис. 4.2; связь 7 на рис. 4.3)** для инициализации ввода или вывода данных подлежащих обработке в компьютере, а счетчик адреса команд увеличивается на длину выполняемой команды в блоке **Память**;
 - если операция является арифметической или логической, то от **Блока формирования управляющих сигналов** поступает сигнал в **Блок выборки из памяти (связь 5)** с директивой «считать из блока **Память (ОЗУ)** значения операндов или их адреса» для последующего сохранения этих значений в **Регистре операции (связь 3)** с целью обеспечения работы **Арифметико-логического устройства**, а затем достигается формирование сигнала в **Арифметико-логическое устройство** на выполнение операции согласно коду операции (**связь 6 на рис. 4.2; связь 7 на рис. 4.3**), **Счетчик адреса команд** увеличивается на длину команды в блоке **Память**;
 - если операция условного перехода, то **Блок центрального управления** анализирует результат выполнения предыдущей операции, который находится в **Регистре операции** и сформирован **Арифметико-логическим устройством**, а затем непосредственно инициирует операцию условного перехода:
 - если знак результата выполнения предыдущей команды равен единице, то в **Счетчик адреса команд** записывается адрес из регистра первого операнда (на следующей итерации операционного цикла будет осуществлено выполнение операции по указанному адресу в блоке **Память** – прямое действие);
 - если знак результата выполнения предыдущей команды равен нулю, то в **Счетчик адреса команд** записывается адрес из регистра второго операнда (на следующей итерации операционного цикла будет осуществлено выполнение операции по указанному адресу в блоке **Память** – альтернативное действие).
 - если операция является операцией безусловного перехода, то в **Счетчик адреса команд** принудительно пересылается содержимое регистра первого операнда (на следующей итерации операционного цикла компьютера будет осуществлено выполнение очередной операции, которая непосредственно расположена по указанному адресу в блоке **Память**).
9. Выполнение операции завершается, а результат сохраняется в **Регистре операции**.

4.7.2. Структура и принцип действия арифметико-логического устройства

Арифметико-логическое устройство реализует выполнение различных операций в основе компьютера, а его структура и принцип действия представлены на рис. 4.4.

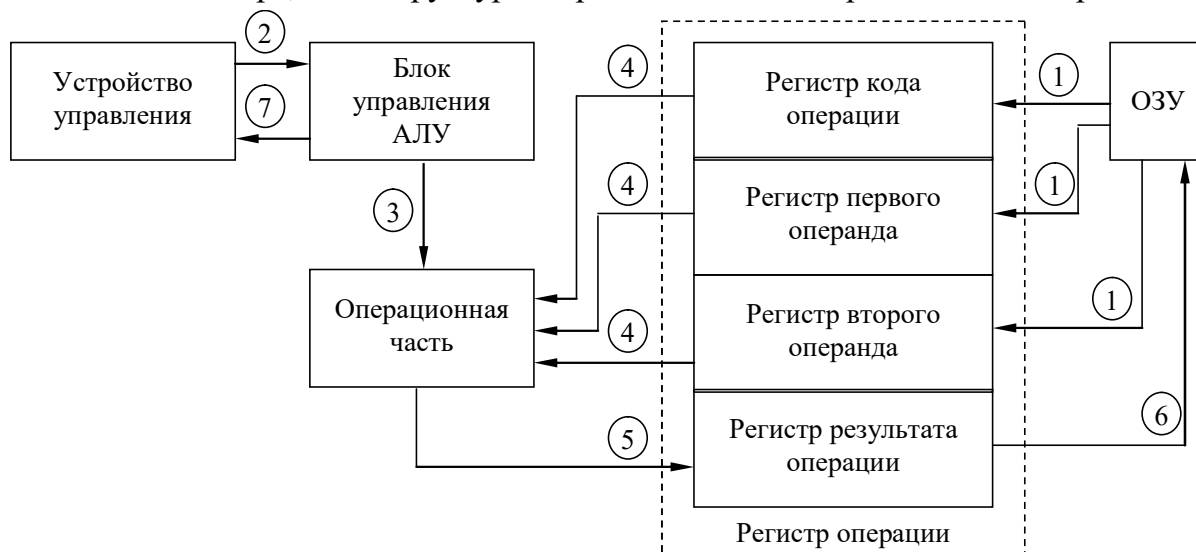


Рисунок 4.4. Структура арифметико-логического устройства

Последовательность выполнения операций представляется следующим образом:

1. Исходными данными для осуществления операционного цикла в **Арифметико-логическом устройстве** выступают операции и значения их операндов (значение кода операции, значение первого операнда, значение второго операнда), которые по командам **Устройства управления** (рис. 4.3) считываются из блока **Память (ОЗУ)** и записываются в **Регистр операции** (связь 1 на рис. 4.4): **Регистр кода операции**, **Регистр первого операнда** и **Регистр второго операнда**.
2. **Устройство управления** реализует синхронизацию выполнения каждой операции посредством формирования последовательности управляющих сигналов, которые направлены в **Блок управления арифметико-логическим устройством** (связь 2) и обеспечивают непосредственное управление всеми заданными фазами вычислительного процесса в **Операционной части** по связи 3 (рис. 4.4).
3. По факту завершения процесса передачи данных из **Устройства ввода информации** в блок **Память** (связь 5 на рис. 4.2) генерируется сигнал в **Устройство управления** по связи 1 (рис. 4.2), а по факту непосредственного сохранения данных в ячейках памяти блока **Память** генерируется сигнал в **Устройство управления** по связи 3 (рис. 4.2).
4. **Устройство управления** генерирует сигнал запроса состояния **Арифметико-логического устройства** для начала выполнения операции по связи 6 (рис. 4.2).
5. **Арифметико-логическое устройство** в случае свободного состояния (выполнение операции завершено) генерирует сигнал о готовности выполнения арифметической или логической операции в **Устройство управления** по связи 7 (рис. 4.2).
6. **Устройство управления** по связи 6 (рис. 4.2) отправляет в **Блок управления арифметико-логическим устройством** сигнал о начале выполнения текущей операции над операндами в **Регистре операции**, а **Блок управления Арифметико-логическим устройством** получает сигналы по связи 2 (рис. 4.4).

7. **Блок управления арифметико-логическим устройством** генерирует сигнал о начале вычислительного процесса в **Операционную часть** (связь 3 на рис. 4.4).
8. В **Операционной части** реализуется предустановленная вычислительная процедура, которая заключается в анализе кода операции в **Регистре кода операции** и последующем выполнении соответствующей операции над двумя значениями операндов, которые располагаются в **Регистре первого операнда** и **Регистре второго операнда** (связь 4), а результат выполнения операции сохраняется в **Регистре результата операции** для дальнейшего использования по связи 5:
 - если операция является операцией ввода данных, то **Арифметико-логическое устройство** генерирует сигнал в **Устройство управления** о запросе состояния **Устройства ввода информации** по связи 7 (рис. 4.4), а **Устройство управления** запрашивает состояние **Устройства ввода информации** по связи 4 (рис. 4.2) до получения положительного ответа о возможности ввода данных по связи 1 (рис. 4.2), потом **Устройство ввода информации** сохраняет данные в блоке **Память** и генерирует ответ **Устройству управления** по связи 1 (рис. 4.2), **Устройство управления** передает сигнал наличия необходимых данных для выполнения операции в ячейках памяти блока **Память** в **Арифметико-логическое устройство** по связи 6 (рис. 4.2);
 - если операция является арифметической или логической, то **Арифметико-логическое устройство** выполняет соответствующую арифметическую или логическую операцию над значениями операндов содержащихся в регистре первого операнда и регистре второго операнда, а затем размещает результирующее значение операции в регистре результата операции (в некоторых случаях для формирования результата допустимо использование регистра первого операнда), потом содержимое регистра операции выгружается в блок **Память** (связь 8 на рис. 4.2; связь 6 на рис. 4.4) и генерируется сигнал о завершении операции в **Устройство управления** по связи 7 (рис. 4.2, 4.4);
 - если операция является операцией вывода информации, то **Арифметико-логическое устройство** генерирует сигнал в **Устройство управления** с запросом состояния **Устройства Вывода информации** по связи 7 (рис. 4.4), а **Устройство управления** запрашивает состояние **Устройства вывода информации** по связи 10 (рис. 4.2) и рано или поздно получает положительный ответ о возможности вывода данных по связи 11, потом **Устройство вывода информации** загружает и выводит данные из блока **Память** по связи 12 и генерирует ответ **Устройству управления** по связи 11 о завершении вывода данных (результат операции).
9. Результат выполнения определенной операции из **Регистра результата операции** перемещается и сохраняется в блоке **Память** (ОЗУ) в основе структуры компьютера.

Немаловажное значение имеет блок **Память** в основе архитектуры классической ЭВМ, который в дальнейшем претерпел множество технологических изменений: четко выделяется внутренняя (ОЗУ) и внешняя (накопители информации) память.

4.7.3. Внутренняя память

Исходя из своего функционального назначения, быстродействия и местоположения в основе архитектуры классического и неоклассического компьютера (ЭВМ) память имеет несколько видов, которые ограничивают сферу ее использования:

- микросхемы памяти – обладают очень низкой скоростью записи информации, относительно низкой скоростью чтения информации, позволяют реализовать хранение микропрограмм базовой системы ввода вывода (Basic Input Output System) на материнской плате, видеокарте, сетевой карте архитектуры ЭВМ;
- модули памяти – обладают очень высокой скоростью записи и чтения информации, представляют собой совокупность микросхем памяти, которые располагаются на печатной плате с одним или двумя рядами контактов и реализуют оперативное запоминающее устройство цифрового автомата;
- элементы внешней памяти – обладают средней и высокой скоростью чтения-записи информации, представлены накопителями информации, которые выступают интегральной совокупностью устройства чтения-записи информации (дисковод или устройство чтения) и носителя информации (бумажная перфокарта или перфолента, магнитная лента, мягкий или жесткий магнитный диск, оптический диск, электронная карта или модуль памяти).

Определение

Внутренняя память предназначена для реализации хранения и оперативной обработки данных, которые поступают с устройств внешней памяти и сетевого адаптера посредством использования центрального процессора компьютера, а также отличается повышенным быстродействием и наличием дополнительных контроллеров для оптимизации чтения и записи, хранения и обработки информации.

При аппаратной реализации компонентов архитектуры компьютера выделяют несколько видов памяти, которые отличаются по механизму функционирования:

- основная память – в современной архитектуре ЭВМ выступает в роли ОЗУ;
- кэш память (рис. 4.5) – реализует буферизацию чтения и записи информации между основной (оперативной) памятью и внешними устройствами хранения информации, функционирует с использованием двух независимых механизмов:
 - FIFO – First In First Out (первый вошел первый вышел),- используется при реализации конвейерного механизма функционирования кэш-памяти;
 - LIFO – Last In First Out (последний вошел первый вышел, «стакан»),- используется при конвейерном механизме работы кэш памяти компьютера.

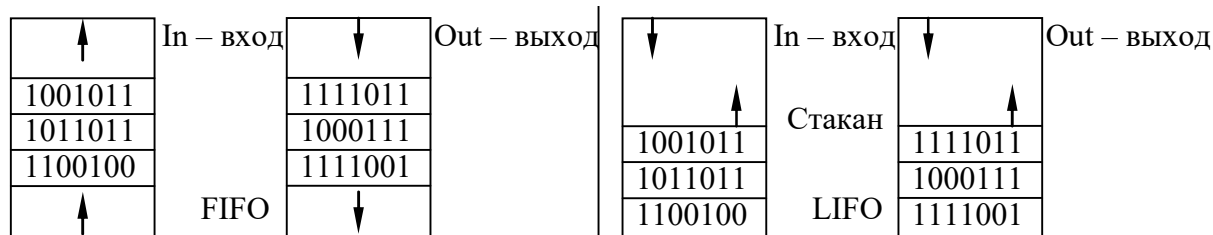


Рисунок 4.5. Принцип действия кэш памяти на основе метода FIFO и LIFO

Определение

Кэш память является самой быстрой и относительно небольшой по объему (от англ. cache – потайной кошелек), выступает буфером между основной памятью и процессором или оперативной памятью и процессором периферийного устройства, что позволяет увеличить скорость выполнения последовательности операций.

Кэш память предназначена для промежуточного хранения данных, которые подлежат обработке центральным процессором компьютера или процессором периферийного (внешнего) устройства в течение ближайших смежных тактов его работы.

Определение

Основная память является частью внутренней памяти компьютера и обеспечивает оперативное сохранение и извлечение основного объема данных и программ, которые временно не используются в течение ближайших смежных тактов работы центрального процессора, но возможно будут использованы через некоторое время, поэтому их выгрузка в устройство внешней памяти нецелесообразна.

Непосредственно после включения компьютера происходит загрузка программного обеспечения в определенной последовательности выполнения компонентов:

- первичная инициализация всех компонентов аппаратного обеспечения ЭВМ;
- выбор активного загрузочного раздела системного логического диска, который создан на одном из физических накопителей информации компьютера;
- обработка главной загрузочной записи и выявление типа файловой системы;
- выполнение загрузчика операционной системы и начало загрузки системы;
- загрузка невыгружаемых и выгружаемых программных компонентов ядра операционной системы, которые необходимы и достаточны для ее работы;
- запуск графического ядра пользователя операционной системы для его работы;
- запуск режима аутентификации пользователя в данной операционной системе.

Память является дискретной и неоднородной, а минимальной единицей хранения информации является 1 бит – бистабильная ячейка информации, которая может сохранять два устойчивых логических состояния: 0 – ноль или 1 – единица.

При работе компьютера используются кодовые комбинации разной длины, которые передаются в виде информационных сообщений (пакетов) между компонентами ЭВМ: 1 бит, 1 байт = 8 бит, 1 Кбайт = 1024 байта = $8 \cdot 1024$ бит = 8192 бит, 1 Мбайт = 1024 Кбайта = $1024 \cdot 1024$ байта = $8 \cdot 1024 \cdot 1024$ бит = 8388608 бит, 1 Гбайт = 1024 Мб = $1024 \cdot 1024$ Кбайт = $1024 \cdot 1024 \cdot 1024$ байта = $8 \cdot 1024 \cdot 1024 \cdot 1024$ бита = 8589934592 бит.

Два информационных элемента по 8 бит образуют машинное слово длиной 16 бит, а два информационных элемента по 16 бит образуют параграф данных в памяти.

Для обращения к элементам памяти применяются разные механизмы адресации.

Минимально возможный адрес в памяти компьютера начинается с нуля.

Максимальный адрес в оперативной памяти компьютера определяется техническими возможностями аппаратного обеспечения, которое реализует память.

Структура основной (оперативной) памяти для компьютеров класса IBM PC с указанием адресов отдельных различных областей представлена в табл. 4.3.

Таблица 4.3

Непосредственно адресуемая память операционной системы				Расширенная память XMA ¹	
Стандартная память CMA ² (640К)		Верхняя память UMA ³ (384К)		высокая память HMA ⁴ (64К)	остальная память
ОЗУ ⁵		ПЗУ			
область служебных программ и данных операционной системы (64К)	область программ и данных пользователя (576К)	область видеопамяти и служебных программ (256К)	область программы начальной загрузки операционной системы и других программ BIOS ⁶ (128К)		
0	64К	640К	896К	1024К	1088К MAX

В силу особенностей программной реализации операционной системы MS DOS (Microsoft Disc Operating System) непосредственно адресуются только первые 1024К оперативной памяти, а доступ к остальным адресам оперативной памяти осуществляется посредством использования драйверов, загрузка которых реализуется при помощи переменной окружения device в файле конфигурации config.sys.

Таблица распределения адресов и состояний сегментов оперативной памяти компьютера формируется операционной системой автоматически посредством использования директивы mem в режиме командной строки (однозадачная операционная система) или в режиме эмуляции MS DOS (многозадачная операционная система).

Определение

Драйвером называется специальная резидентная компьютерная программа, обеспечивающая управление и обработку событий инициированных внешним устройством (периферийный компонент компьютера) или пользователем (нажатие на определенную клавишу клавиатуры), загрузка которой производится непосредственно на первых этапах загрузки операционной системы или после ее полного запуска.

Определение

Резидентность – это свойство компьютерной программы сколько угодно долго находиться в оперативном запоминающем устройстве и реагировать на запросы внешних потребителей: операционная система, программа для ЭВМ, периферийный компонент компьютера, а также разные события инициированные пользователем.

¹ XMA – eXtended Memory Area

² CMA – Conventional Memory Area

³ UMA - Upper Memory Area

⁴ HMA – High Memory Area

⁵ ОЗУ – оперативное запоминающее устройство, ПЗУ – постоянное запоминающее устройство

⁶ BIOS – Basic Input-Output System – базовая система ввода-вывода – компонент ОС MS DOS

Определение

Многозадачность – свойство операционной системы, которое позволяет выполнять в условно параллельном режиме несколько задач или процессов и обеспечивать переключение между ними в процессе функционирования компьютера.

Определение

Операционной системой называется компьютерная программа, которая создает среду программного окружения в период исполнения программного обеспечения пользователя (системные, прикладные и диагностические программы), а также реализует мониторинг состояния и управление всеми аппаратными и программными компонентами, поддерживающими выполнение разных задач пользователя на ЭВМ.

Однозадачная операционная система поддерживает командный интерфейс взаимодействия с пользователем и реализует выполнение только одной задачи без разделения во времени (MS DOS), допускает установку нескольких драйверов устройств и оболочку, которая позволяет оптимизировать работу с файловой системой.

Многозадачная операционная система имеет графический интерфейс, поддерживает возможность выполнения нескольких задач пользователя в многопоточном режиме с разделением во времени (MS Windows), а также допускает запуск большого количества драйверов внутренних (внутренний модем, внутренний аудио адаптер) и внешних периферийных устройств (внешний модем, принтер, сканер, дигитайзер).

Оперативная память компьютера рассматривается в рамках двух точек зрения:

- как аппаратный компонент архитектуры ЭВМ – представляет собой печатную плату, которая содержит несколько микросхем памяти и подключается к материнской плате посредством использования слота SIMM, DIMM или RIMM;
- как программный компонент – выступает адресуемым пространством для хранения информации, которая представляет собой программные компоненты, данные операционной системы, а также файлы и папки пользователя.

Для расширения объема оперативной памяти определенного компьютера необходимо подобрать некоторое количество печатных плат памяти определенного объема хранимой информации, которые содержат несколько микросхем памяти.

Операционная система и прикладное программное обеспечение используют различные методы адресации при работе с оперативной памятью компьютера (ЭВМ):

- прямая (непосредственная) – используется физический или логический адрес;
- косвенная (ссылочная) – применяют ссылку на физический или логический адрес;
- относительная (сегментная) – полный адрес формируется из базы и смещения.

В простейшем случае для образования сегментного адреса все адресное пространство делится на блоки – сегменты – определенного размера 8КБ(512МБ), которые нумеруются шестнадцатеричными числами от 0000 (00000000) до FFFF (FFFFFFF). Адрес в памяти формируется посредством указания базы (базового адреса) и относительного смещения (смещения относительно базового адреса) – пара: базовый адрес : смещение. Например: 1FA4 : 0010 (10000001 : 0000FAAB).

4.7.4. Внешняя память

Элементы внешней памяти выступают в качестве накопителей информации разного рода, а также реализуют возможность долговременного хранения информации, которая по каким-либо причинам временно не используется операционной системой.

Определение

Внешняя память представлена в основном накопителями информации на основе ряда носителей информации: перфолента, перфокарта, магнитная лента, магнитный диск, оптический диск и электронная карта памяти или электронный носитель.

Определение

Накопитель информации на магнитных и оптических дисках структурно включает устройство чтения/записи информации (дисковод) и съемный носитель информации определенного типа и размера (магнитный или оптический диск).

Выделяют несколько видов накопителей на магнитных дисках и лентах:

- накопитель на магнитных лентах, Streamer (STR) – представляет собой два компонента: устройство чтения-записи (дисковод) и магнитную ленту (катушка).
- накопитель на жестких магнитных дисках (винчестер), Hard Disc Drive (HDD) – устройство управления чтением-записью и шпиндель с магнитными дисками, которые конструктивно изготовлены в едином неразборном корпусе;
- накопитель на гибких магнитных дисках, Floppy Disk Drive (FDD) – представляет собой два независимых компонента: дисковод (устройство чтения-записи) и дискету (магнитный диск в специальном пластиковом корпусе).

Соответственно, также выделяют несколько видов носителей информации:

- магнитная лента – используется в стримерах, которые обеспечивают ее протягивание относительно головок чтения-записи информации (создание архивов);
- магнитные диски – являются элементами накопителей информации на гибких (дискета как съемный магнитный диск) и жестких магнитных дисках (неразборные);
- электронные диски – представляют собой электронную карту памяти с устройством чтения-записи карт памяти (раздельно) или набор микросхем памяти с электронным устройством чтения-записи информации в едином компактном корпусе.

Определение

Магнитный диск – это пластиковый (для гибких дисков), либо алюминиевый или керамический (для жестких дисков) диск с чувствительным покрытием на его поверхностях, который позволяет регистрировать два магнитных состояния соответствующие двум логическим состояниям: логический ноль и логическая единица.

Накопитель информации на гибких магнитных дисках содержит всего один магнитный диск с двумя чувствительными поверхностями к магнитному полю.

Накопитель информации на жестких магнитных дисках конструктивно является неразборным устройством и включает несколько магнитных дисков на оси (шпинделе).

В процессе поиска, чтения и записи информации магнитный диск FDD и HDD раскручивается электрическим приводом, который конструктивно расположен у его оси.

5. Базовые элементы и устройства архитектуры современных персональных электронных вычислительных машин

В теории информации как отрасли занимающейся разработкой разных технических средств обработки информации понятие архитектуры может применяться по отношению к аппаратному, программному или алгоритмическому обеспечению.

Архитектура аппаратного обеспечения рассматривается как совокупность аппаратных компонентов разного рода, которые объединены в определенную иерархию по ряду классических и современных принципов (расположение программы в памяти, микропрограммное управление, условный переход, открытая архитектура) или критериев (энергопотребление, энергосбережение, быстродействие, надежность, точность), а также в процессе своего функционирования решают задачи обработки информации.

Архитектура программного обеспечения представляет собой совокупность программных средств (операционные системы, пакеты прикладных программ, утилиты, диагностические программы, оболочки, программные среды для разработки программ), которые позволяют реализовать управление потоком обработки данных.

Архитектура алгоритмического обеспечения представляет собой совокупность функциональных схем и логических выражений, которые отражают принцип функционирования распределенных вычислительных систем включающих большое количество процедур и алгоритмов, а также временных диаграмм состояний.

На основе алгоритмического обеспечения разрабатывается аппаратное и программное обеспечение, а затем все виды обеспечения верифицируются, при этом осуществляется поиск несоответствий и ошибок, реализуется усовершенствование и внедрение полученных достижений в различные области деятельности специалистов.

В данном разделе основной акцент делается на особенности архитектуры аппаратного обеспечения информационно-вычислительных систем и компьютеров.

Разработка и производство элементов аппаратного обеспечения выступает весьма динамичной отраслью, а продукция востребована на информационном рынке.

Развитие аппаратного обеспечения обусловлено появлением новаций в области технологий материального производства (физика, микроэлектроника, теория материалов, технологии производства топологий интегральных микросхем), а также в области информационных технологий (параллельные вычисления, структуры данных, БД).

Определение

Под **архитектурой** аппаратного обеспечения в информатике понимают совокупность основных компонентов и оборудования, блоков и узлов, а также управляющих модулей, которые рассматриваются как единое целое и реализуют решение различных вычислительных задач на основе программного обеспечения.

При разработке архитектуры компьютера производителем предусматривается возможность улучшения технических характеристик аппаратного обеспечения, а при этом пользователь имеет возможность подключать такой набор стандартизованных компонентов и устройств, которые реализуют решение конечного набора задач.

5.1. Классификация современных компьютеров

Современный этап развития вычислительной техники характеризуется широким спектром разных видов ЭВМ, которые обеспечивают решение задач разного рода.

Существуют различные критерии классификации современных компьютеров.

1. **По функциональному назначению** цифровые автоматы дифференцируются на:

- универсальные ЭВМ (персональные) – ориентированы на решение широкого круга задач различных категорий пользователей и характеризуются возможностью обработки данных разного вида (числовые, символьные, графические и т.п.), наращиваемым объемом внутренней и внешней памяти, а также большим количеством периферийных устройств для ввода и вывода информации;
- проблемно ориентированные ЭВМ (серверы и терминалы) – решают более узкий круг задач, например, обеспечивают управление технологическими процессами, регистрацию и анализ непрерывно поступающих данных, а также обладают ограниченным набором аппаратного и программного обеспечения;
- специализированные ЭВМ (адаптеры и контроллеры) – обеспечивают решение узкого круга задач, имеют узко специализированную архитектуру, которая включает ограниченный набор аппаратных средств и программных компонентов, а также обладают низким энергопотреблением, высокой производительностью, надежностью (отказоустойчивостью) и невысокой стоимостью.

2. **По размерам и техническим возможностям** выделяют несколько классов ЭВМ:

- супер ЭВМ (стратегического назначения) – являются мощными многопроцессорными вычислительными системами, которые обладают сверхвысоким быстродействием и позволяют реализовать параллельные вычисления при решении нескольких стратегических задач (моделирование, прогнозирование, диагностика, мониторинг, системный анализ, управление, интерпретация);
- большие ЭВМ (mainframes) – позволяют в многопользовательском режиме (несколько тысяч пользователей одновременно) решать научно-технические и практические задачи, обрабатывать запросы к базам и банкам данных, обеспечивать мониторинг и балансировку сетевой нагрузки в сложных компьютерных сетях при передаче различной информации (обеспечение возможности проведения видеоконференции в реальном времени по нескольким цифровым каналам, проходящим через глобальные сети передачи данных);
- малые ЭВМ (mini) – используются как терминалы в контурах управления сложными технологическими объектами и процессами, которые реализованы по принципам супервизорного и локального управления, а также применяются в многопользовательских системах проектирования и моделирования;
- микро ЭВМ (micro) – ноутбуки и наладонные компьютеры, которые выступают универсальными и специализированными по отношению к классам решаемых задач, а также при этом обладают относительно высоким быстродействием, низким энергопотреблением и высокой компактностью, а также надежностью.

3. **По количеству одновременно работающих пользователей** выделяют:

- однопользовательские операционные системы и компьютеры – технические особенности компьютера предусматривают взаимодействие с одним пользователем посредством использования командного интерфейса операционной системы;
- многопользовательские операционные системы и компьютеры (локальный режим) – предусматривают работу нескольких пользователей посредством использования командного или диалогового интерфейса в основе операционной системы с разделением во времени за счет применения разных профилей;
- многопользовательские (сетевой режим) – обеспечивают параллельную работу нескольких пользователей за счет создания и использования локальной вычислительной сети, которая позволяет предоставлять открытый доступ с целью совместного использования разных ресурсов предприятия (организации).

4. **По конструктивным особенностям исполнения системного блока выделяют:**

- настольные вертикальные (tower case) – системный блок компьютера расположен вертикально и при этом он устанавливается как правило на полу;
- настольные горизонтальные (desktop case) – системный блок компьютера расположен горизонтально и непосредственно устанавливается как правило на столе;
- портативные (ноутбуки) – выполняются в виде основания с клавиатурой и крышки с интегрированным дисплеем, обладают автономным источником питания повышенной емкости (аккумуляторная батарея элементов питания);
- наладонные (handle) – изготавливаются с использованием современных достижений в области микроэлектроники, что позволяет обеспечить относительно высокую компактность и энергосбережение, а также очень низкий вес;
- электронные переводчики и записные книжки – разновидность наладонных микро компьютеров, которые имеют аппаратное и программное обеспечение узко специализированного назначения для выполнения узкого круга задач;
- инженерные калькуляторы с интегрированным графопостроителем – реализуют поддержку аналитически-численных расчетов при работе инженеров.

5. **По обеспечению режима реального времени в период исполнения программ:**

- системы без поддержки режима реального времени – предназначены для использования широким кругом потребителей и не реализуют привязки к реальному времени в период исполнения разного программного обеспечения;
- системы мягкого реального времени – операционная система предусматривает наличие приоритетов и привилегий на уровне ядра и файловой системы;
- системы реального времени – операционная система имеет специальный диспетчер контроля периода исполнения всех процессов и потоков данных с расширенной системой прерываний, а также распределением приоритетов и привилегий;
- системы жесткого реального времени – отличаются минимальным значением времени отклика на все запросы, которые инициированы программным окружением операционной системы и действиями пользователя при решении задач.

5.2. Основные принципы построения архитектуры современных компьютеров

Принципы построения архитектуры современных ЭВМ основаны на классических принципах в основе прынстонской архитектуры компьютера, которая разработана коллективом ученых и исследователей под руководством Джона фон Неймана:

- архитектура современных компьютеров развивается на основе классической:
 - устройство ввода информации – клавиатура, манипулятор (мышь, тачпад, трекбол, джойстик, midi-клавиатура, микрофон, сканер, цифровое перо);
 - арифметико-логическое устройство, устройство управления, микропроцессорная память входят в основу центрального процессора (CPU – Central Processing Unit) как элемента современной архитектуры компьютера;
 - память – на современном этапе развития ЭВМ дифференцируется на внутреннюю память, которая обладает низким объемом и сверхвысоким быстродействием, а также оперативную память среднего быстродействия и объема, внешнюю память большого объема и низкого быстродействия;
 - устройство вывода данных – дисплей (электронно-лучевой монитор, жидкокристаллический монитор, плазменная панель), принтер, цифровое табло;
- принцип микропрограммного управления – последовательность действий и вычислительных процедур задается программой, которая состоит из последовательности директив, операндов и операций над ними, а каждая директива и операция включает: [код операции][операнд 1], [операнд 2], ..., [операнд n] (если используется один операнд – операция унарная, если два операнда – операция бинарная, а если задействовано n операндов – операция n-арная);
- принцип условного перехода – программа состоит из линейной последовательности команд и операций с операндами, при этом оператор условного перехода реализует ветвление алгоритмической структуры, появляется правило: Если (условие) То (прямое действие), Иначе (альтернативное действие), - при выполнении условия в antecedенте выполняется прямое действие в консеквенте, а в противном случае реализуется альтернативное действие;
- принцип размещения программы в памяти – программа находится в памяти, а для смены решаемой задачи посредством ЭВМ требуется загрузить новую программу;
- принцип сегментации и стандартизации адресации всех ячеек памяти компьютера;
- принцип иерархичности – архитектура ЭВМ содержит в своей основе совокупность разных компонентов, которые взаимодействуют между собой для решения набора актуальных задач пользователя на основе имеющихся программ. На современном этапе развития добавляют принцип открытой архитектуры.

Определение

Принцип открытой архитектуры – обеспечивает возможность взаимной совместимости аппаратного обеспечения и переносимости программного обеспечения между принципиально разными платформами и операционными системами.

5.3. Структура персонального компьютера

В структуре компьютера принято выделять набор нескольких групп компонентов:

- основные внешние компоненты персонального компьютера (ЭВМ) следующие:
 - системный блок – металлический корпус, который обеспечивает экранирование электромагнитного излучения и содержит все внутренние элементы ЭВМ;
 - монитор – выступает основным устройством вывода графической информации;
 - клавиатура – является основным устройством ввода символьной информации и управления текстовым курсором в процессе ввода данных пользователем;
 - манипулятор (мышь, трекбол, тачпад) – вспомогательное устройство ввода информации и управления курсором манипулятора определенного вида при работе с графическим интерфейсом пользователя в операционной системе;
- основные внутренние компоненты, которые расположены внутри системного блока:
 - материнская плата – реализует коммутацию всех устройств компьютера;
 - центральный процессор – выступает основным устройством управления;
 - оперативное запоминающее устройство – реализует сохранение данных;
 - накопитель информации на гибких магнитных дисках – дискета и дисковод;
 - накопитель информации на жестких магнитных дисках – «винчестер»;
 - лазерный накопитель информации на оптических дисках – диск и дисковод;
 - электронный накопитель информации – контроллер и микросхемы памяти;
 - платы расширения и интерфейсные модули – устанавливаются в свободные слоты;
- дополнительные (периферийные) компоненты персонального компьютера (ЭВМ):
 - внешний электронный накопитель информации – подключается к слоту USB;
 - принтер – периферийное устройство вывода информации на печатный носитель;
 - сканер – устройство для реализации ввода растровой графической информации;
 - модем – устройство передачи данных по коммутируемому, кабельному, радиочастотному или спутниковому каналу доступа на основе разных протоколов;
 - видеокамера – позволяет записать динамическое видео-изображение;
 - звуковая карта – позволяет реализовать воспроизведение аудио-данных, которые предварительно записаны на накопителе информации определенного вида;
 - плата видео-захвата – позволяет осуществить оцифровку видео-информации и записать видео-поток на носитель информации определенного рода;
 - FM-тюнер – реализует воспроизведение передач радиостанций FM-диапазона;
 - TV-тюнер – позволяет принимать и воспроизводить телевизионный сигнал;
 - мультимедиа микрофон – позволяет вводить монофонический аудио-поток;
 - акустическая система – реализует воспроизведение стерео аудио-потока.

Определение

Элементы структуры персонального компьютера принято дифференцировать на **внутренние (internal)** – располагаются внутри системного блока компьютера и подключаются через слоты расширения на материнской плате и **внешние (external)** – находятся вне системного блока, подключаются посредством внешних портов на нем.

5.4. Особенности расположения внутренних и внешних элементов и устройств современных компьютеров

С развитием архитектуры современных компьютеров появилось большое количество внутренних и внешних устройств, которые реализуют выполнение определенного набора функций: ввод информации, вывод информации, обработка данных, сохранение данных, передача данных между ЭВМ по разным каналам связи.

Внешние и внутренние элементы и устройства компьютера представлены на рис. 5.1.

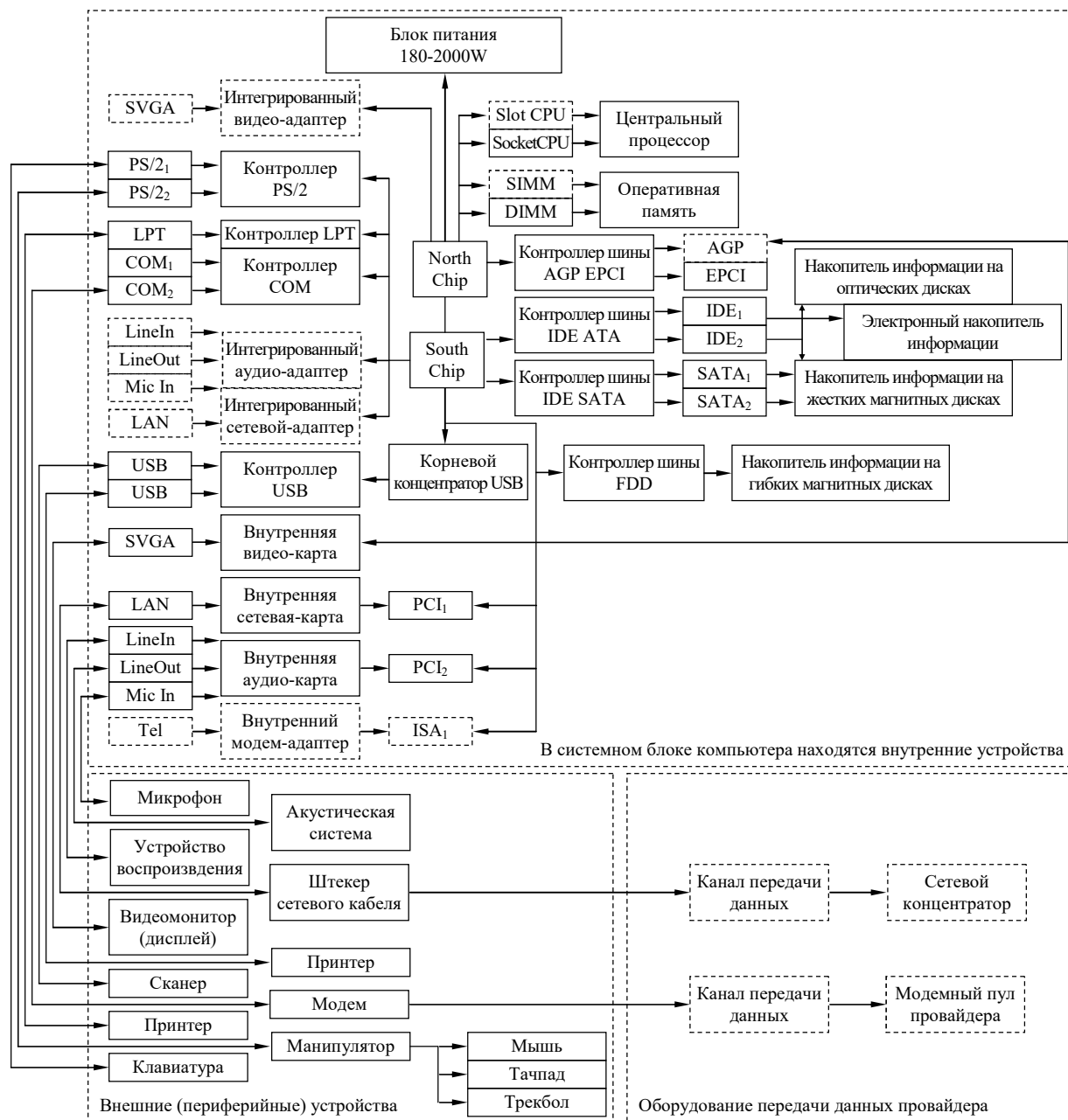


Рисунок 5.1. Расположение внешних и внутренних элементов компьютера

Современные материнские платы выступают основой для соединения устройств разного типа и назначения, а также содержат интегрированные адаптеры: сетевая карта (разъем LAN на внешней панели), звуковой адаптер (три разъема на внешней панели: Line In – линейный вход, Line Out – линейный выход, Mic In – микрофон), видео карта (разъем VGA на внешней панели), встроенный модем (разъем Tel Line).

5.5. Внешние элементы персональной электронной вычислительной машины

Внешние элементы в основе архитектуры и конфигурации компьютера выступают минимально необходимым набором основных компонентов, который достаточен для реализации вычислительного процесса при решении задач пользователя.

Определение

Внешними элементами компьютера (ПЭВМ) являются: системный блок, монитор, клавиатура и манипуляторы разного типа (mouse – мышь, trackball – трекбол, touchpad – тачпад), а также различное вспомогательное периферийное оборудование (принтер, сканер, модем, дигитайзер, плоттер и прочее) для частных задач.

Все внешние компоненты реализуют определенный набор функций и задач, а также подключаются к системному блоку компьютера, который снабжается электрической энергией посредством использования сети питающего напряжения 220В.

Сеть питающего напряжения 220В имеет ряд существенных недостатков:

- аварийное отключение электроснабжения – обусловлено организационными и техническими мероприятиями при обслуживании объектов энергоснабжения, а также аварийным срабатыванием аппаратуры защиты сетей и цепей питания (магистральный рубильник и автоматический предохранитель);
- непрогнозируемое спонтанное отключение электрического тока – вызвано природными явлениями и авариями, которые инициируют повреждение магистрального провода, подводящего кабеля или распределительного провода;
- сетевые импульсные помехи – связаны с флуктуациями фронтов напряжения и тока, которые вызывают кратковременную систематическую перегрузку и асинхронизацию стробирующих и тактирующих импульсов в процессе функционирования разных электронных компонентов компьютера (ЭВМ).

Для профилактики и предотвращения выхода из строя компонентов компьютеров, а также элементов и устройств информационных систем применяют:

- организационные меры по предотвращению выхода из строя компонентов:
 - систематический осмотр теплоотводящих компонентов различных электрических схем и узлов компьютера: кулеры и вентиляторы в их основе;
 - проверка соответствия фактического срока полезного использования компонентов ЭВМ с допустимыми нормами согласно действующим требованиям;
 - систематическая замена вышедших из строя компонентов и устройств;
 - архивирование и резервное копирование данных на накопители информации;
- технологические меры по предупреждению нештатных и аварийных ситуаций:
 - установка аппаратуры защиты сети: автоматические предохранители;
 - применение сетевых фильтров для подавления сетевых импульсных помех, которые выражаются в кратковременных выбросах и провалах тока;
 - использование источников бесперебойного питания для поддержания параметров питающего напряжения и его номинала в случае отключения сети.

5.5.1. Системный блок и блок питания

Системный блок содержит предустановленный блок питания разной мощности.

Определение

Системный блок – металлический корпус, который обеспечивает экранирование электромагнитного излучения и содержит электронное оборудование компьютера.

Выделяют 2 основных вида и несколько разновидностей исполнения корпусов:

- вертикальное расположение (tower, башня) – вариант исполнения системного блока для установки на пол или специальную подставку интегрированную в стол:
 - baby-tower – предназначен для установки одного устройства размером 5,25" (CD-ROM, DVD-ROM, ZIP Drive, ZAP Drive, Mobile Rack for HDD, кулер);
 - mini-tower – обеспечивает установку 1^{го} устройства 3,5" и 2^х устройств 5,25";
 - midi-tower – позволяет установить 1-2 устройства 3,5" и 2-3 устройства 5,25";
 - big-tower – допускает установку 1-2^х устройств 3,5" и 3-5^{ти} устройств 5,25".
- горизонтальное расположение (desktop, подставка) – вариант, который предусматривает установку системного блока компьютера на стол под монитор:
 - ultra-slim – предназначен для установки только одного устройства 5,25";
 - slim – обеспечивает установку 1^{го} устройства 3,5" и 2^х устройств 5,25";
 - small – позволяет установить 1-2 устройства 3,5" и 2-3 устройства 5,25";

В состав системного блока компьютера входит стандартный набор компонентов:

- блок питания – обеспечивает энергоснабжение всех компонентов компьютера:
 - стандарт АТ – высоковольтное включение посредством нажатия кнопки для коммутации двух сигнальных линий, на которые подается напряжение 220В;
 - стандарт АТХ – низковольтное включение посредством нажатия кнопки, подключенной к материнской плате для коммутации одной сигнальной линии;
- железный корпус с индикаторами на передней панели – предназначен для экранирования электромагнитного излучения, а светодиодные индикаторы отражают состояние компьютера: регламентированная работа от сети или аккумулятора, спящий или ждущий режим, чтение-запись данных на накопители информации (некоторые модификации корпуса содержат внешние порты USB);
- внутренние устройства компьютера располагаются внутри системного блока:
 - материнская плата (System Board) – прикрепляется к правой стенке системного блока, предусматривает подключение разъема питания АТ или АТХ;
 - микросхемы большой и средней степени интеграции на материнской плате (Nord Chipset, South Chipset) – обслуживают компоненты материнской платы;
 - центральный процессор (Central Processing Unit) – крепится в сокет или слот;
 - модули памяти (Memory Modules) – вставляются в разъемы SIMM или DIMM;
 - платы расширения, подключаемые к системной плате (Integrated Cards);
 - накопители информации различных типов (Disc Drives) – FDD, HDD, CDROM;
 - интерфейсные кабели (Interface Cables) – плоские и цилиндрические;
 - система охлаждения (Cooling System) – кулер блока питания и процессора.

Определение

Блок питания – обеспечивает энергоснабжение системной платы и внутренних устройств, конструктивно собран и комплектуется вместе с корпусом системного блока, обладает определенной мощностью зависящей от поколения архитектуры ЭВМ (рис. 5.2).

Выделяют два основных стандарта (типа) при производстве блоков питания:

- стандарт АТ – кнопка включения/выключения компьютера находится на передней панели системного блока и обеспечивает высоковольтную (110/220В) коммутацию двух токовых петель, которые обеспечивают подведение питающего напряжения, что приводит в включению/выключению блока питания (блок питания имеет дополнительную кнопку включения/выключения питания);
- стандарт АТХ – кнопка включения/выключения компьютера находится на передней панели системного блока, а включение/выключение компьютера обеспечивается коммутацией низковольтной (5В) токовой петли посредством нажатия кнопки, которая соединена с материнской платой (блок питания предусматривает низковольтное включение, но имеет на своем корпусе дополнительную кнопку высоковольтного включения/выключения питания).

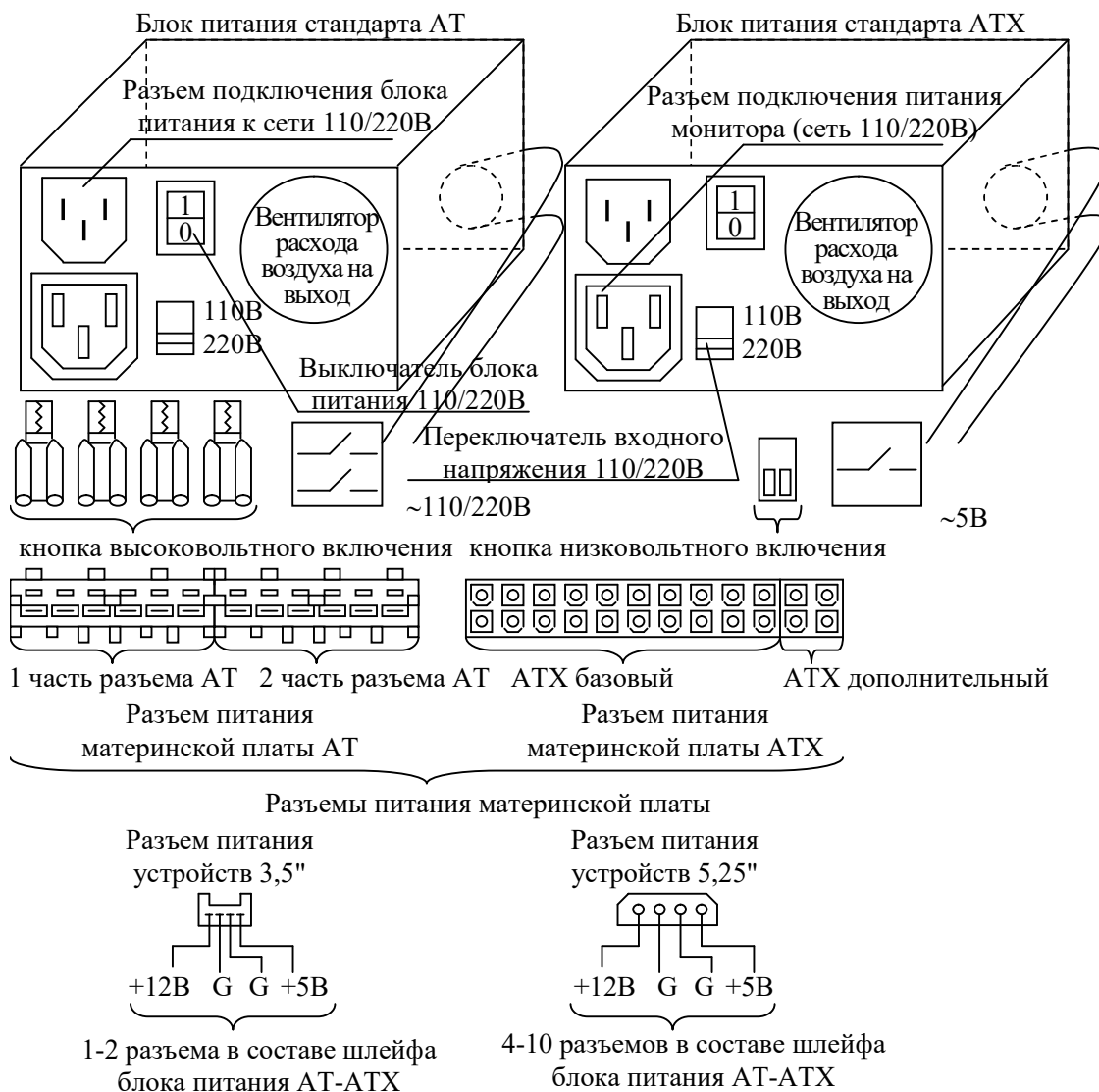


Рисунок 5.2. Блок питания стандартов АТ и АТХ

Структурно блок питания включает несколько основных компонентов (рис. 5.2):

- разъем питания блока питания от силовой сети с напряжением 110/220В;
- разъем питания на блоке питания для подключения монитора от сети 110/220В;
- кнопка включения/выключения блока питания 110/220В (может отсутствовать);
- вентилятор расхода воздуха на выход, который обеспечивает вывод теплого воздуха из объема блока питания и системного блока с целью снижения температуры;
- схема понижения и распределения напряжения (внутри блока питания);
- разъем для питания материнской платы стандарта АТ или АТХ (рис. 5.2);
- разъемы для подключения питания внутренних устройств 3,5" и 5,25".

Выделяют два основных вида разъемов питания для подключения внутренних устройств компьютера, которые отличаются размером, но содержат по четыре контакта:

- разъем питания устройств 3,5 дюйма – обеспечивает энергоснабжение накопителя информации на гибких магнитных дисках (FDD – Floppy Disc Drive);
- разъем питания устройств 5,25 дюйма – обеспечивает энергоснабжение накопителя информации на жестких магнитных дисках (HDD – Hard Disc Drive) и накопителя информации на лазерных оптических дисках (CD-ROM – Compact Disc-Read Only Memory, CD-RW-ROM – Compact Disc ReWritable ROM).

Особенности архитектуры и топология связи компонентов аппаратного обеспечения, которое расположено в основе компьютера класса Pentium 4 (рис. 5.3).

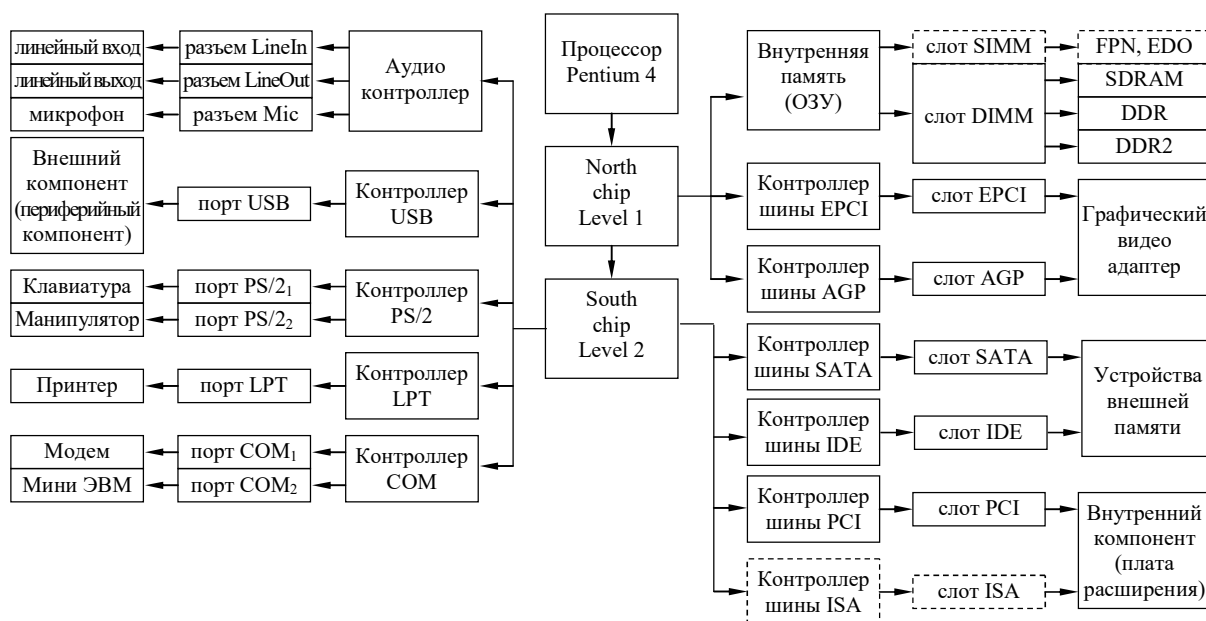


Рисунок 5.3. Топология связи элементов архитектуры компьютера

Аудио контроллер на некоторых материнских платах может отсутствовать, тогда необходимо подключить аудио-адаптер в шине PCI как внутренний компонент компьютера.

Котроллер шины USB обслуживает от двух до четырех портов и устройств USB.

Контроллер порта PS/2 обслуживает два порта PS/2, которые позволяют подключить клавиатуру и один из манипуляторов типа мышь, тачпад или трекбол.

Контроллер порта LPT реализует управление одним параллельным портом LPT.

Контроллер порта COM обслуживает два последовательных порта COM.

Структура элементов и устройств персонального компьютера (ЭВМ) может быть представлена в виде функциональной схемы следующим образом (рис. 5.4).

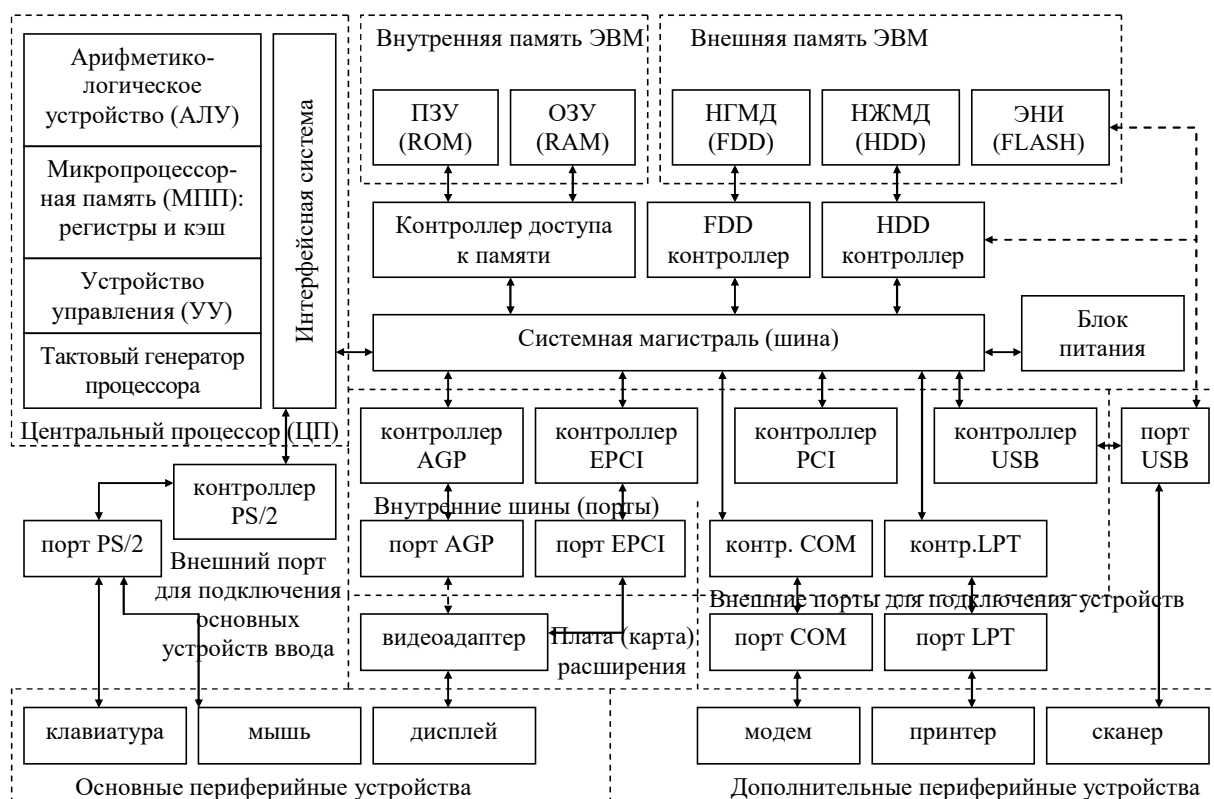


Рисунок 5.4. Элементы и устройства архитектуры современных компьютеров

Материнская плата компьютера обеспечивает интерфейс взаимодействия между всеми внутренними и внешними периферийными компонентами компьютера.

Добавление внутреннего компонента компьютера (ЭВМ) осуществляется внутрь системного блока посредством использования слотов EPCI, AGP, PCI, ISA, IDE (S)ATA.

Подключение внешнего периферийного устройства компьютера осуществляется посредством использования внешних портов и разъемов: USB, LPT, COM, PS/2.

Определение

Конфигурация компьютера определяется техническими характеристиками аппаратного обеспечения, представляющего собой набор внутренних (устройства подключаемые посредством внутренних портов и шин расширения) и внешних (устройств компьютера работающих через внешние порты) элементов структуры ЭВМ.

Персональные компьютеры (ЭВМ), которые относятся к идентичной архитектуре могут иметь принципиально разную конфигурацию на уровне элементов, в частности:

- персональный компьютер 1: Pentium 3 1300 МГц; MB Asus Intel 815; ОЗУ 512Мб; HDD 80Гб; FDD 1,44Мб; 4 USB; Internal Audio Creative Sound Blaster Live 8.1; Internal Video nVidia 8800PE, Internal LAN Adaptor 100/1000Мб;
- персональный компьютер 2: Pentium 3 1300 МГц; MB Asus Intel 815; ОЗУ 1Гб; HDD 200Гб; FDD 1,44Мб; 4 USB; Internal Audio Creative Sound Blaster Live 8.1; Internal Video nVidia 8800PE, Internal LAN Adaptor 100/1000Мб.

Компьютеры содержат идентичную материнскую плату, но разные ОЗУ и HDD.

5.5.2. Видеодисплей

Видеодисплей выступает основным устройством вывода видео информации.

Определение

Видеодисплеем (монитором) называют устройство вывода информации, которое реализует визуальное отображение данных поступающих с видеоадаптера в символьном или графическом режиме посредством использования мерцающих транспарантов, светодиодной матрицы, электрической электронной вакуумной лампы, пассивной или активной жидкокристаллической матрицы или плазменной панели.

Структурно-функциональная схема видеодисплея (монитора), который функционирует на основе электрической электронно-лучевой трубки представлена на рис. 5.5.

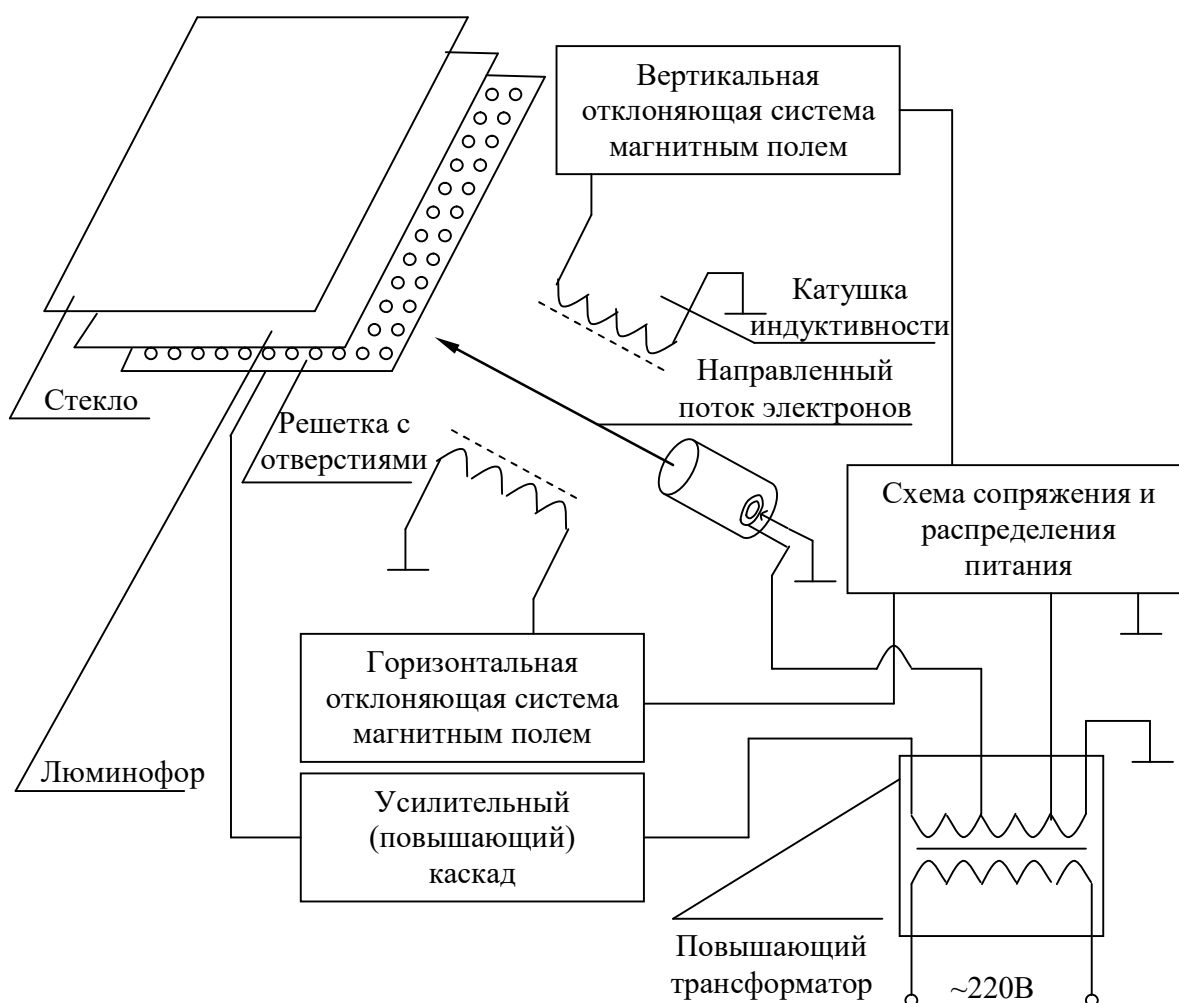


Рисунок 5.5. Структура электронно-вакуумной трубки в основе видеомонитора

Определение

Период послесвечения люминофора – интервал времени, который измеряется между первичным попаданием элементарной частицы (электрона) на поверхность люминофора и полным падением интенсивности (яркости) при его свечении.

Определение

Частота регенерации экрана – понимают количество полных обновлений графического изображения на поверхности экрана (люминофора) посредством обстрела электронной пушкой за 1 минуту (значение лежит в интервале 50-300 Гц).

Определение

Цветопередача – способность монитора компьютера передавать без искажений совокупность цветов и их тонов или полутонов, которые принимает каждый элементарный пиксель экрана в отдельности при отображении графического изображения.

Отличительными особенностями и техническими параметрами видеомонитора собранного на основе жидкокристаллической матрицы являются следующие:

- низкий вес и габаритные размеры – отсутствие электрической электронной вакуумной лампы, вертикальной и горизонтальной отклоняющей системы магнитным полем, повышающего трансформатора, сопрягающих контуров;
- низкая частота регенерации экрана – вместо пульсирующей электронной пушки используется источник фотонного излучения и матовое стекло, что позволяет генерировать равномерный относительно непрерывный поток света, поэтому интервал обновления отображаемого изображения на экране дисплея проявляется только при просмотре существенно динамического изображения;
- высокое энергосбережение и низкая рабочая температура при использовании.

Дисплей на основе жидкокристаллической матрицы представлен на рис. 5.6.

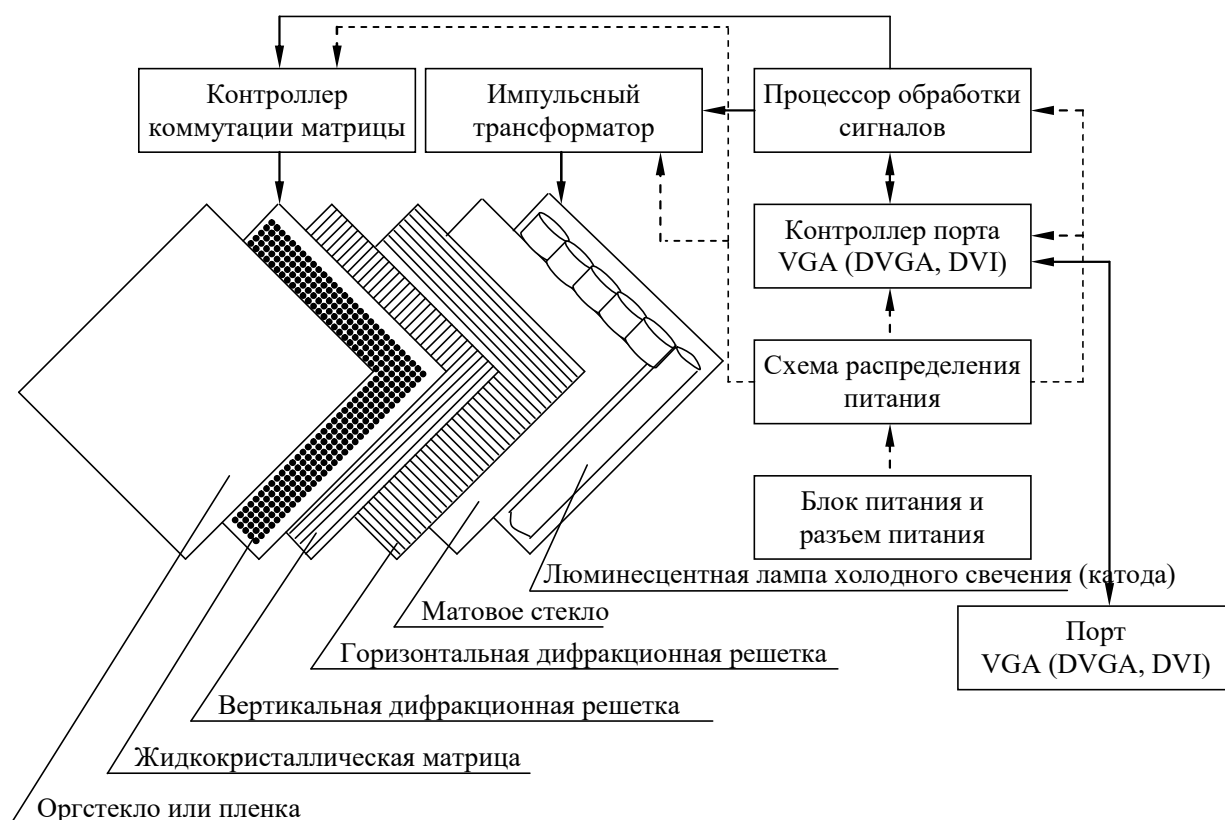


Рисунок 5.6. Структура дисплея на основе жидкокристаллической матрицы

Контроллер коммутации матрицы обеспечивает коммутацию вертикальных и горизонтальных сигнальных линий, которые реализуют изменение коэффициента преломления в объеме элемента жидкокристаллической матрицы видеодисплея.

Импульсный трансформатор позволяет реализовать питание линейки люминесцентных ламп холодного свечения, которые наряду с матовым стеклом генерируют однородный относительно непрерывный поток фотонного излучения (света).

5.5.3. Клавиатура

Клавиатура является основным устройством ввода текстовой информации посредством использования различных клавиш, которые выполняют разные функции.

Определение

Клавиатурой называют устройство ввода информации, которое содержит несколько светодиодных индикаторов (Numb Lock, Caps Lock, Scroll Lock), а также несколько полей с клавишами: поле основного ввода, поле цифрового ввода, поле управления курсором, поле функциональных клавиш, поле служебных клавиш.

Структура и расположение клавиш на клавиатуре представлена на рис. 5.7.

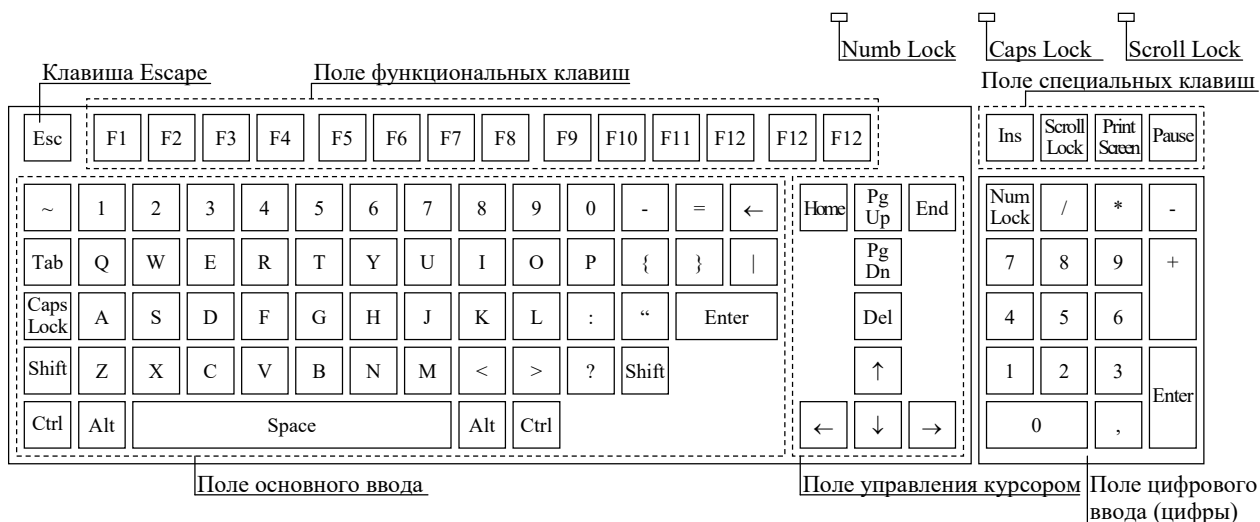


Рисунок 5.7. Структурная схема расположения клавиш на клавиатуре

Клавиатура подключается к системному блоку компьютера разными способами:

- при помощи провода, который подключается к разъему PS/2 на системном блоке;
- беспроводным способом, посредством использования радиопередатчика с зарядным устройством, подключаемого и питающегося через порт USB на системном блоке.

Клавиатура содержит области (поля) с клавишами разного назначения:

- клавиша Esc – нажатие инициирует останов текущей операции или действия;
- поле функциональных клавиш – содержит последовательность горизонтально расположенных клавиш, которые выполняют определенные функции при работе с приложениями, например для компонентов Microsoft Office: F1 – вызов справочной системы, F2 – создание нового документа, F3 – открытие существующего документа, F4 – сохранение документа, F5 – отмена предыдущего действия, F6 – повтор последнего действия, F7 – проверка орфографии, F8 – перемещение файла (папки), F9 – ответить на письмо, F10 – ответить всем на письма, F11 – переслать содержание письма, F12 – отправить почту;
- поле основного ввода – содержит совокупность клавиш с буквами, цифрами и символами, которые обеспечивают возможность ввода текстовой информации;
- поле управления курсором – обеспечивает управление текстовым курсором;
- поле цифрового ввода – при активности светодиодного индикатора Numb Lock обеспечивает ввод цифровых данных (цифр) в ходе аналитических расчетов.

В верхнем правом углу клавиатуры находятся три светодиодных индикатора:

- индикатор Caps Lock – отражает состояние удержания верхнего регистра при вводе последовательности букв (при активности вводятся заглавные буквы);
- индикатор Num Lock – активизация поля цифрового ввода на клавиатуре;
- индикатор Scroll Lock – активизация непрерывной прокрутки вверх и вниз.

Особенности подключения обычной и радио-клавиатуры к системному блоку компьютера посредством использования разъема PS/2 и USB отражены на рис. 5.8.

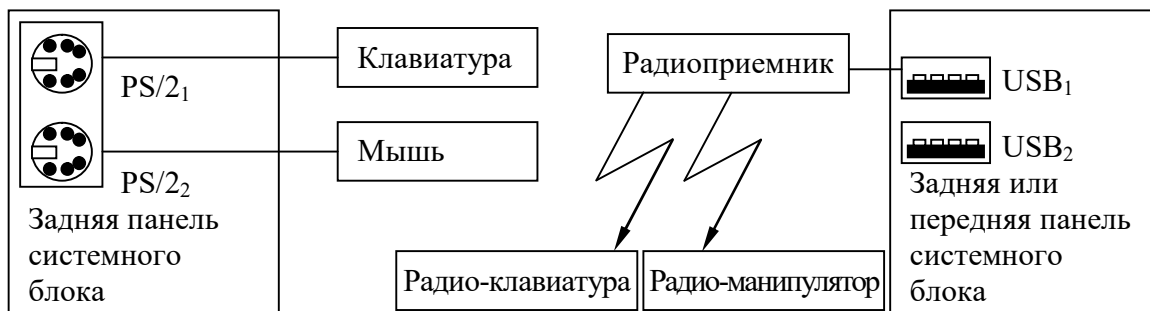


Рисунок 5.8. Схема подключения обычной и радио-клавиатуры к системному блоку компьютера

Обычная клавиатура подключается посредством использования порта PS/2 или USB, а радио-клавиатура имеет встроенный радиопередатчик состояния клавиш, взаимодействует через радиоприемник, который подключается при помощи порта USB.

Радио-манипулятор типа мышь и приемник для порта USB представлены на рис. 5.9.

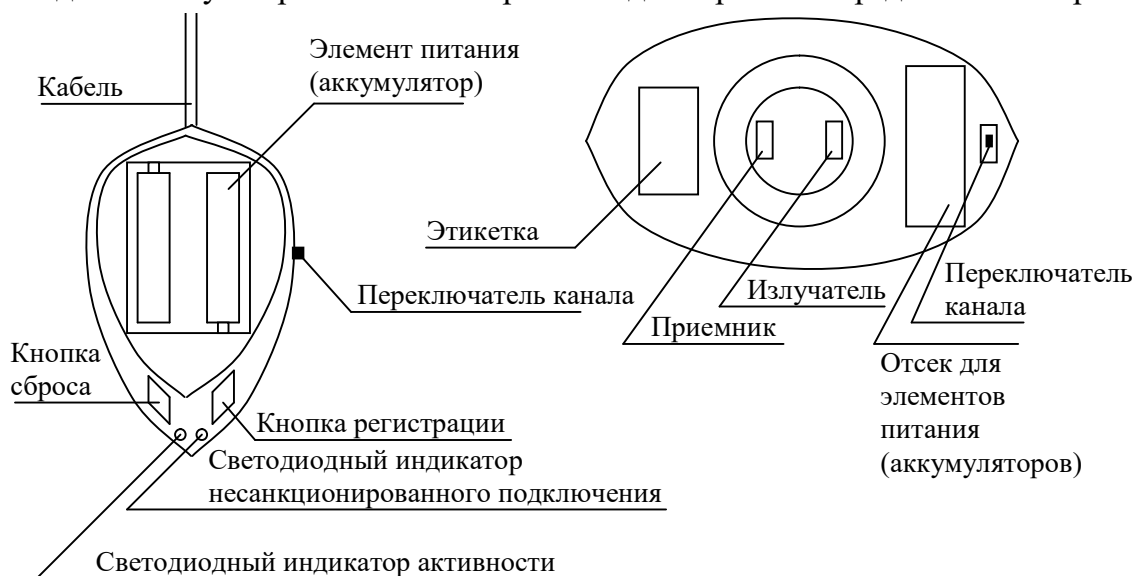


Рисунок 5.9. Особенности структуры радио манипулятора типа мышь и приемника

На корпусе радио-манипулятора типа мышь имеется несколько элементов:

- кнопка сброса – обеспечивает сброс внутреннего процессора манипулятора;
- кнопка регистрации – позволяет записать идентификатор нового устройства;
- переключатель канала – реализует выбор одного из радиоканалов обмена;
- светодиодный индикатор активности – отражает активность при смене состояния;
- светодиодный индикатор несанкционированного подключения – отражает наличие аналогичного манипулятора, не зарегистрированного в памяти приемника.

5.5.4. Манипулятор типа мышь, трекбол и тачпад

Манипулятор выступает вспомогательным устройством ввода информации, применяется в локальных и сетевых операционных системах с разным интерфейсом:

- MS DOS (командный интерфейс) – манипулятор типа мышь, трекбол или тачпад используется только при установке графической оболочки (Norton Commander);
- MS Windows (графический интерфейс) – используется только при работе пользователя с графическим интерфейсом (GUI – Graphical User Interface).

Определение

Манипулятором называют вспомогательное устройство ввода информации, которое содержит две или более кнопок для выделения графических объектов, активизации процедур, запуска на выполнение программ (левая кнопка), позволяет реализовать отображение свойств элементов интерфейса, пиктограмм, папок, файлов, ярлыков (правая кнопка), а также обеспечивает управление процессом перемещения курсора при изменении местоположения своего корпуса на плоскости (рис. 5.10).

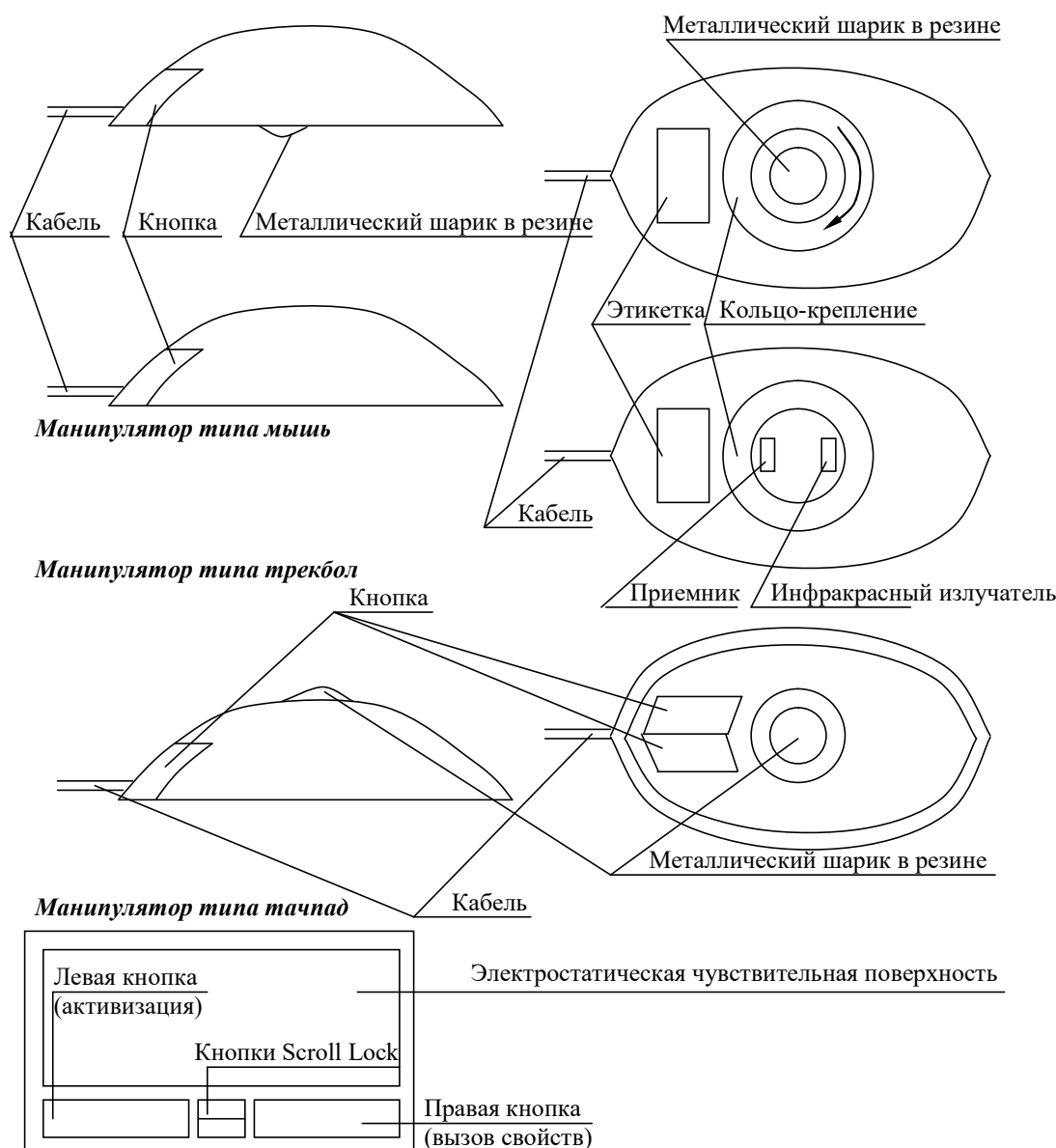


Рисунок 5.10. Особенности структуры манипулятора типа мышь, трекбол и тачпад

5.6. Внутренние элементы структуры персонального компьютера

Согласно представленному определению, внутренние элементы структуры компьютера (ЭВМ) располагаются внутри системного блока и подключаются через внутренние шины, порты и слоты расширения, среди которых можно выделить:

- CPU (Central Processor Unit) Socket/Slot – разъем (сокет или слот) для плоской и вертикальной установки (подключения) центрального процессора;
- SIMM (Single In-line Memory Module) – слот для подключения односторонних модулей памяти устаревших стандартов (отсутствует на новых материнских платах):
 - FPM (Fast Page Module) – вид динамической памяти, который позволяет быстро получать доступ к данным, которые находятся на одной и той же странице, что и данные передаваемые во время предыдущего цикла;
 - EDO (Extended Data Out) – модифицированный вариант FPM, который позволяет начинать выборку данных из следующего блока памяти в то же самое время, когда отправляет данные в предыдущий блок (синхронно);
- DIMM (Dual In-line Memory Module) – слот для подключения двухстороннего модуля памяти разного объема нескольких технологических стандартов:
 - SDRAM (PC66, PC100, PC133);
 - SDRAM DDR (PC1600, 2100, 2400, 2700, 3000, 3200, 3500, 3700, 4000, 4200);
 - SDRAM DDR2 (PC2-3200, 4200, 5300, 6000, 6400, 7200, 8000);
- RIMM (Rambus Memory Module) – слот для установки модулей памяти стандартов:
 - RIMM-16 (RIMM1200, 1400, 1600, 2100, 2400);
 - RIMM-32 (RIMM3200, 4200, 4800);
- AGP (Accelerated Graphics Port) – высокоскоростной (ускоренный) графический порт для подключения видеоадаптера с графическим процессором;
- PCI (Peripheral Component Interconnect) – шина обеспечивающая межсоединение различных периферийных компонентов и центрального процессора компьютера при передаче данных, спецификация 2.1 (33МГц, 32 разряда), Intel 1993;
- EPCI (Extended Peripheral Component Interconnect) – расширенная шина обеспечивающая межсоединение периферийных компонентов и CPU при передаче данных;
- ISA (Industry Standard Architecture) – шина архитектуры промышленного стандарта 8 разрядов (IBM PC/XT) и 16 разрядов (IBM PC AT), 8МГц;
- EISA (Extended ISA) – расширенная архитектура шины промышленного стандарта, 32 разряда, частота передачи данных по сигнальным линиям до 132 Мб/сек;
- IDE ATA (Integrated Drive Electronics) – промышленный интерфейс и протокол доступа к накопителям информации на жестких магнитных и оптических дисках;
- IDE SATA (IDE Super AT Attachment) – модифицированный интерфейс и протокол доступа к накопителям информации на жестких магнитных дисках;
- SCSI ATAPI (AT Attachment Packet Interface) – интеллектуальный пакетный интерфейс периферийных устройств для AT-совместимых компьютеров.

5.6.1. Системная плата компьютера

Системная или материнская плата выступает основной платформой, которая реализует сопряжение всех внутренних и внешних компонентов персональной ЭВМ.

Определение

Системная (материнская) плата (System board) – это основной структурный элемент современной ЭВМ, который обеспечивает сборку конфигурации компьютера за счет сопряжения всех внутренних и внешних устройств и элементов между собой.

Внешний вид и технические характеристики всех и каждого в отдельности элементов внутренней структуры компьютера, которые подключаются к материнской плате, как правило, связаны с поколением архитектуры центрального процессора.

По способу подключения центрального процессора к материнской плате (рис. 5.11):

- посредством разъема типа слот (slot) – предполагает установку плоскости центрального процессора перпендикулярно плоскости материнской платы;
- при помощи разъема типа сокет (socket) – предусматривает установку плоскости центрального процессора параллельно плоскости материнской платы.

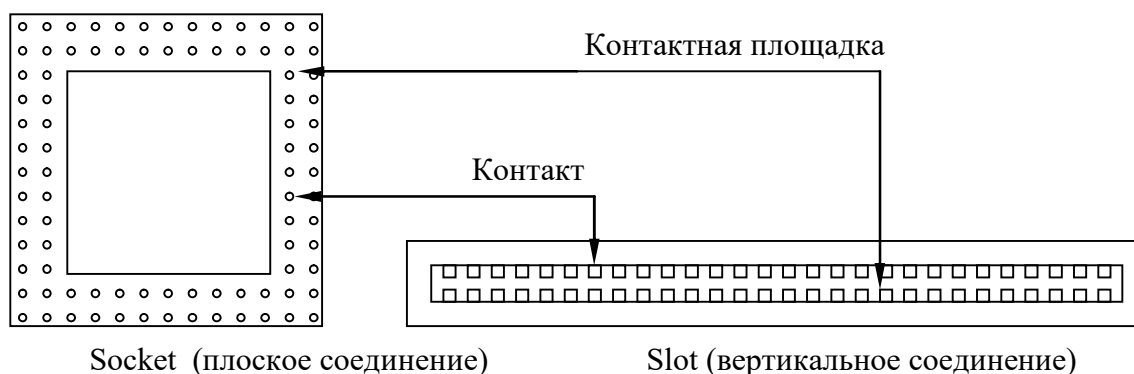


Рисунок 5.11. Способы подключения центрального процессора к материнской плате

По количеству подключаемых различными способами к материнской плате и используемых в дальнейшем центральных процессоров в основе компьютера:

- однопроцессорные – предусматривают подключение только одного процессора;
- двухпроцессорные – содержат два разъема для подключения двух процессоров;
- многопроцессорные – позволяют использовать несколько процессоров.

Настройка параметров внутренних слотов и внешних портов материнской платы с учетом установленных компонентов конфигурации ЭВМ осуществляется:

- программно – посредством программы BIOS (Basic Input Output System);
- аппаратно – при помощи переключателей (jumpers) на материнской плате.

Совместимость аппаратного обеспечения достигается за счет разработки международных стандартов на интерфейсы и разъемы аппаратного обеспечения, поэтому рядовой пользователь, который не обладает специальными навыками может наращивать и модернизировать архитектуру ЭВМ согласно своим потребностям.

Высокая переносимость разного программного обеспечения компьютера обеспечивается за счет разработки программ на основе единых принципов и стандартов.

В частности, обеспечивает классический принцип иерархии памяти компьютера.

5.6.2. Центральный процессор

Центральное процессорное устройство (CPU – Central Processing Unit) выступает основным устройством управления обработкой потоков инструкций и данных, которые сопровождают арифметические, логические, циклические операции над числами.

Отличительными характеристиками центрального процессора являются:

- производитель процессора: i – Intel, AMD – Analog Micro Devices;
- поколение (семейство) процессора: i286, i386, i486, iPentium, iPentium 2, iPentium 3, iPentium 4 HP (Hyper Trading), iPentium 4 DC (Dual Core);
- тактовая частота процессора и модулей памяти: 20МГц, 33МГц, 66МГц, 133-266МГц, 333-500МГц, 533-800МГц, 800-1300МГц, 1300МГц-4000МГц;
- разрядность шины адреса (ША) и шины данных (ШД): 4, 8, 16, 32, 64 бита.

Разрядность ША \geq ШД, поскольку необходимо обеспечить адресацию всех элементарных ячеек оперативного запоминающего устройства, объем которого существенно возрастает при переходе от одного поколения архитектуры ЭВМ к другому.

Типовая структура центрального процессора включает несколько компонентов:

- устройство управления – реализует управление внутренними устройствами процессора и внешними компонентами в течение операционного цикла;
- тактовый генератор – генерирует последовательность синхронизирующих импульсов, обеспечивающих синхронность и синфазность работы всех компонентов ЭВМ;
- арифметико-логическое устройство – обеспечивает выполнение арифметических, логических, регистровых и прочих операций в ходе операционного цикла;
- блок внутренних инструкций – дополнительный набор инструкций для математического сопроцессора, интегрированного мультимедиа адаптера на материнской плате;
- регистровая внутренняя память – сверхбыстрая память относительно малого объема, которая предназначена для хранения значений операндов, промежуточных и результирующих значений арифметических, логических и прочих видов операций в течение смежных тактов работы процессора, согласно пакетно-конвейерной технологии обмена данными (Pipelined Burst Cache) выделяют:
 - многоуровневую систему приоритетов при реализации обработки информации;
 - кэш первого уровня – сверхбыстрая память, которая располагается на первом уровне кэша и все данные с максимальным приоритетом используются в течение текущего (следующего) такта работы центрального процессорного устройства, а временно неиспользуемые данные перемещаются на нижний уровень конвейера в кэш второго уровня (временно не используются);
 - кэш второго уровня – сверхбыстрая память, которая располагается на втором уровне кэша и все обрабатываемые данные используются в течение нескольких смежных тактов функционирования центрального процессора.

Определение

Центральный процессор – выступает центральным устройством управления, которое обеспечивает непрерывную обработку потоков данных инициированных операционной системой, прикладными программами и действиями пользователей.

5.6.3. Оперативное запоминающее устройство

Выступает связующим звеном по данным с кэш памятью и внешней памятью.

Определение

Оперативное запоминающее устройство (ОЗУ) – элемент внутренней архитектуры компьютера, который обеспечивает сохранение и извлечение данных, которые временно не используются в течении операционного цикла смежных тактов обработки данных центральным процессором, а их выгрузка на машинный носитель информации не является целесообразной по каким-либо причинам.

Оперативное запоминающее устройство представлено совокупностью модулей памяти, которые подключаются посредством интерфейсов разного поколения:

- SIMM (Single In-Line Memory Module) – Intel 386, 486, Pentium;
- DIMM (Dual In-Line Memory Module) – Intel Pentium, Pentium 2, Pentium 3;
- RIMM (Rambus Memory Module) – Intel Pentium 4, Intel Pentium 4 Dual Core.

Согласно логической организации модули памяти подразделяются на совокупность банков памяти, каждый из которых поддерживает функцию отдельной адресации всех ячеек памяти, а также чтение, запись и обновление информации.

Эволюция микросхем памяти стандарта SDRAM (Synchronous Dynamically Read Only Memory) обусловило появление микросхем DDR (Double Data Rate).

Физически модуль памяти состоит из набора интегральных микросхем разного объема, которые расположены на одной или двух сторонах платы печатного монтажа содержащей одностороннюю или двухстороннюю контактную площадку (табл. 5.1).

В зависимости от принципа записи данных микросхемы памяти дифференцируются:

- масочные микросхемы памяти (ROM – read only memory) поддерживают запись информации посредством прожигания специальных проводящих вставок на матрице в объеме кристалла микросхемы, а считывание информации заключается в измерении состояния проводимости (сопротивления) разных элементов записанной матрицы, стирание информации не предусматривается;
- масочные микросхемы памяти с ультрафиолетовым стиранием (Ultraviolet ROM) – принцип записи и чтения информации аналогичен простым масочным микросхемам памяти, но предусмотрено восстановление исходного состояния матрицы посредством воздействия ультрафиолетового излучения;
- электрически программируемые (EPROM – electrically programmable read only memory) – запись, чтение и обновление ячеек памяти реализуется посредством электрических сигналов, а повторную запись информации не предусматривают;
- электрически перепрограммируемые микросхемы (EEPROM – electrically erasable ROM) – аналогичны электрически перепрограммируемым, но предусматривают возможность стирания и перезаписи информации посредством электрических сигналов;
- многократно электрически перепрограммируемые микросхемы (Flash EEPROM) – аналогичны электрически перепрограммируемым, но количество циклов перезаписи (стирания-записи) очень большое (100000-10000000 раз).

Таблица 5.1

Основные виды и технические характеристики современных микросхем памяти

Стандарт модуля	Формат	Вид м/с	Тактовая частота, МГц	Кол-во циклов за такт	Частота шины данных, МГц	Размер шины, байт	Скорость обмена, МБ/с
FPM	SIMM	60ns	22	1	22	8	177
EDO	SIMM	60ns	33	1	33	8	266
PC66	SDRAM DIMM	10ns	66	1	66	8	533
PC100	SDRAM DIMM	8ns	100	1	100	8	800
PC133	SDRAM DIMM	7/6ns	133	1	133	8	1066
PC1600	DDR DIMM	DDR200	100	2	200	8	1600
PC2100	DDR DIMM	DDR266	133	2	266	8	2133
PC2400	DDR DIMM	DDR300	150	2	300	8	2400
PC2700	DDR DIMM	DDR333	166	2	333	8	2667
PC3000	DDR DIMM	DDR366	183	2	366	8	2933
PC3200	DDR DIMM	DDR400	200	2	400	8	3200
PC3500	DDR DIMM	DDR433	216	2	433	8	3466
PC3700	DDR DIMM	DDR466	233	2	466	8	3733
PC4000	DDR DIMM	DDR500	250	2	500	8	4000
PC4200	DDR DIMM	DDR533	266	2	533	8	4266
PC2-3200	DDR2 DIMM	DDR2-400	200	2	400	8	3200
PC2-4200	DDR2 DIMM	DDR2-533	266	2	533	8	4266
PC2-5300	DDR2 DIMM	DDR2-667	333	2	667	8	5333
PC2-6000	DDR2 DIMM	DDR2-750	375	2	750	8	6000
PC2-6400	DDR2 DIMM	DDR2-800	400	2	800	8	6400
PC2-7200	DDR2 DIMM	DDR2-900	450	2	900	8	7200
PC2-8000	DDR2 DIMM	DDR2-1000	500	2	1000	8	8000
RIMM1200	RIMM-16	PC600	300	2	600	16	1200
RIMM1400	RIMM-16	PC700	350	2	700	16	1400
RIMM1600	RIMM-16	PC800	400	2	800	16	1600
RIMM2100	RIMM-16	PC1066	533	2	1066	16	2133
RIMM2400	RIMM-16	PC1200	600	2	1200	16	2400
RIMM3200	RIMM-32	PC800	400	2	800	32	3200
RIMM4200	RIMM-32	PC1066	533	2	1066	32	4266
RIMM4800	RIMM-32	PC1200	600	2	1200	8	4800

5.6.4. Внешнее запоминающее устройство

Внешнее запоминающее устройство (накопитель информации) предназначено для долговременного хранения временно неиспользуемых данных и обеспечивает:

- загрузку программных компонентов операционной системы непосредственно из загрузочного раздела системного логического диска после завершения первичной инициализации оборудования посредством микропрограммы базовой системы ввода вывода (BIOS – Basic Input Output System) на системной плате;
- выгрузку из кэш памяти и ОЗУ временно неиспользуемых данных, которые не обрабатываются в данный момент операционной системой и пользователем ЭВМ.

Определение

Внешнее запоминающее устройство (ВЗУ) – элемент внешней архитектуры компьютера, обеспечивающий долговременное хранение временно неактуальных данных, которые не используются на протяжении длительного интервала времени операционной системой и прикладным программным обеспечением при решении задач в данном сеансе работы пользователя на определенном компьютере (ЭВМ).

В зависимости от конструктивных особенностей исполнения выделяют:

- накопители информации на гибких магнитных дисках (НГМД):
 - накопители информации стандарта FDD (Floppy Disc Drive);
 - накопители информации стандарта ZIP-Drive и ZAP-Drive;
- накопитель информации на магнитных лентах (STR – Streamer);
- накопители информации на жестких магнитных дисках (HDD – Hard Disc Drive):
 - внутренний и внешний накопители информации стандарта IDE (Integrated Drive Electronics, Integrated Data Equipment) ATAPI (Attachment Packet Interface);
 - внутренний (internal) и внешний (external) накопители информации на магнитных диска стандарта IDE SATA (Serial Attachment Packet Interface);
 - внутренний накопитель информации на магнитных дисках стандарта SCSI (Smart Cable Serial Interface или Small Computer Systems Interface);
- лазерные накопители информации на оптических дисках (ЛНОД):
 - лазерные накопители информации на оптических дисках типа CDROM
 - лазерные накопители информации на оптических дисках типа DVDROM;
- накопители информации на электронных картах памяти (НЭКП):
 - внутренние накопители информации на электронных картах (Internal EC);
 - внешние накопители информации на электронных картах (External EC);
 - внешние носимые накопители информации на электронных картах (Flash Drive).

Конструктивно внешнее запоминающее устройство представляет собой разборное или неразборное устройство, которое включает носитель информации (магнитный диск, оптический диск, интегральную микросхему) и устройство чтения-записи информации (дисковод или устройство чтения электронных карт памяти, которые содержат специальный контроллер) для реализации обмена информацией через интерфейс и разъем определенного типа (FDD, IDE ATA, SCSI, IDE SATA, USB).

Внутренний накопитель информации содержит два разъема (интерфейс обмена данными и питание), а внешний накопитель информации получает питание через интерфейс.

5.7. **Накопители информации**

Накопители информации обеспечивают непрерывный контроль непосредственной записью, чтением и поиском хранимых данных на разных носителях информации посредством использования различных физических принципов регистрации данных:

- механический – используется в накопителях информации на перфокартах и перфолентах, обладает высоким уровнем шума и низкой скоростью доступа;
- электромагнитный – применяется в накопителях информации на гибких и жестких магнитных дисках, содержащих магниточувствительные поверхности;
- оптический пучок – обеспечивает запись-чтение определенных оптических дисков стандарта CD-ROM и DVD-ROM в лазерных накопителях информации;
- электрический – реализует очень быстрое чтение-запись информации в электронных картах памяти, модулях и банках памяти разного назначения.

Определение

Накопителем информации выступает устройство внешней памяти, которое обеспечивает возможность долговременного хранения временно неиспользуемых данных операционной системы (системное программное обеспечение) и пользователя (прикладное программное обеспечение) на носителях информации различного типа, а также реализует непрерывный контроль процесса чтения и записи информации.

Выделяют два типа накопителей информации, которые отличаются конструкцией:

- неразборная конструкция – интегральная совокупность устройства чтения-записи информации и собственно носителя информации определенного типа;
- разборная конструкция – два независимых устройства, которые соответственно реализуют управление процессом чтения-записи информации (устройство чтения перфокарт, устройство чтения перфолент, стример, дисковод, устройство чтения электронных карт памяти) и непосредственное хранение информации (перфокарта, перфолента, магнитная лента, дискета, электронная карта памяти) с использованием определенного физического принципа записи и чтения информации (пробивание отверстий заданной формы, регистрация устойчивых бинарных магнитных состояний, сохранение электростатического потенциала).

Определение

Носителем информации называют разборное или неразборное техническое средство с блочным или символьным типом записи информации, которое обеспечивает регистрацию логических состояний на своей поверхности или в объеме.

Выделяют ряд носителей информации, которые отличаются конструкцией:

- перфокарточный носитель информации на картонной прямоугольной основе;
- перфоленточный носитель информации на бумажной ленточной основе;
- носитель информации на магнитных лентах расположенный на бабинах;
- носитель информации на гибких и жестких магнитных дисках разного размера;
- носитель информации на оптических дисках Recordable/ReWritable CD-ROM/DVD-ROM;
- носитель информации на электронных картах памяти и модулях памяти.

5.7.1. Накопители информации на перфокартах

Накопитель информации на основе перфокарт позволяет ввести самый маленький объем данных для их последующей обработки посредством использования компьютера.

Определение

Накопитель информации на перфокарточных носителях (рис. 5.12) представляет собой набор независимых компонентов: устройство чтения перфокарт (устройство ввода перфокарты), устройство для пробивания отверстий на поверхности перфокарты, а также перфокарту, которая содержит матрицу из рядов различных букв и цифр.

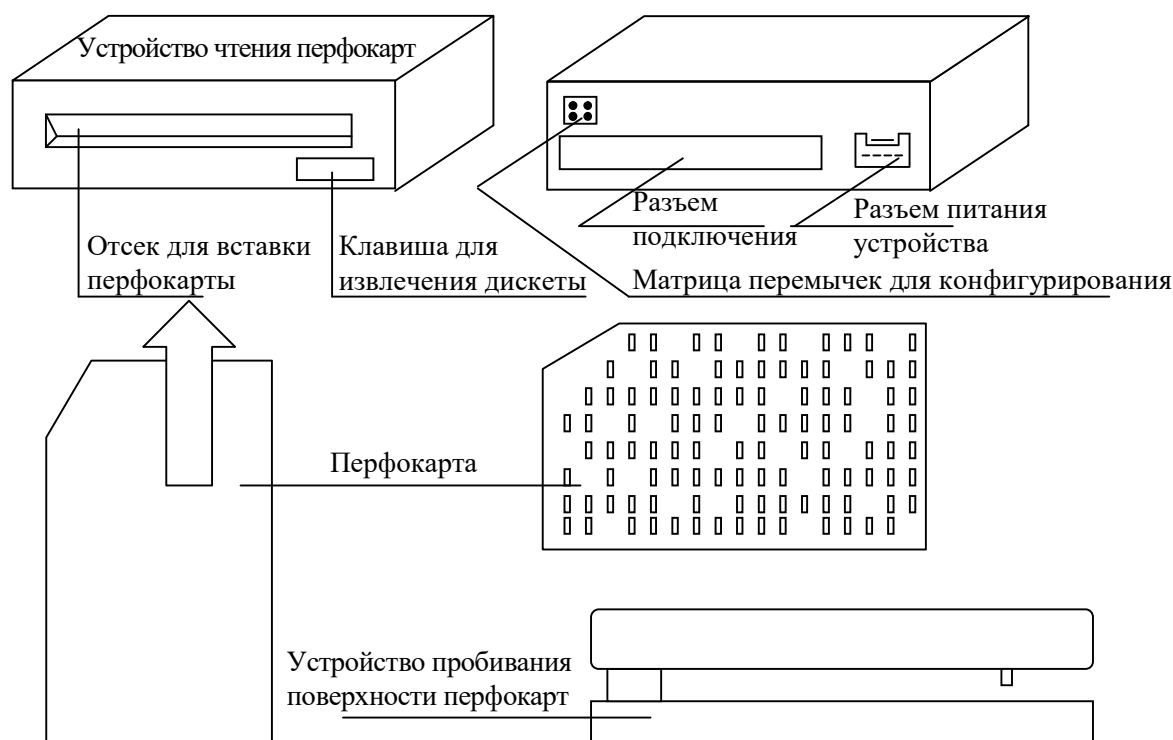


Рисунок 5.12. Структура накопителя информации на основе перфорированной карты

На рисунке представлена совокупность различных элементов реализующих ряд функций в основе накопителя информации на основе бумажных перфокарт:

- устройство чтения перфокарт – обеспечивает чтение матрицы перфокарты;
 - отсек для вставки перфокарт – позволяет установить перфокарту для чтения;
 - клавиша для извлечения перфокарты – реализует извлечение перфокарты;
 - разъем подключения питания устройства – обеспечивает питание устройства;
 - матрица переключателей для конфигурирования – реализует установку параметров;
- устройство для пробивания отверстий на поверхности перфокарты – позволяет реализовать пробивание поверхности перфокарты для обеспечения возможности программирования компьютера посредством устройства чтения;
- перфокарта – бумажная пластина с косым срезом в верхнем левом углу, содержащая прямоугольную матрицу с несколькими рядами букв и цифр.

Для непосредственного использования накопителя информации на основе перфокарт необходимо предварительно подготовить перфокарту – пробить в ее поверхности несколько отверстий, которые расположены в разных частях поверхности.

5.7.2. Накопители информации на перфолентах

Накопитель информации на основе перфолент позволяет ввести самый маленький объем данных для их последующей обработки посредством использования компьютера, при этом скорость ввода информации существенно выше по сравнению с накопителем информации на основе перфокарт. Перфокарта долговечнее по отношению к перфоленте.

Определение

Накопителем информации на перфолентах называют устройство чтения-записи данных посредством пробивания поверхности бумажной ленты, которая протягивается относительно коммутационной матрицы для чтения информации (рис. 5.13).

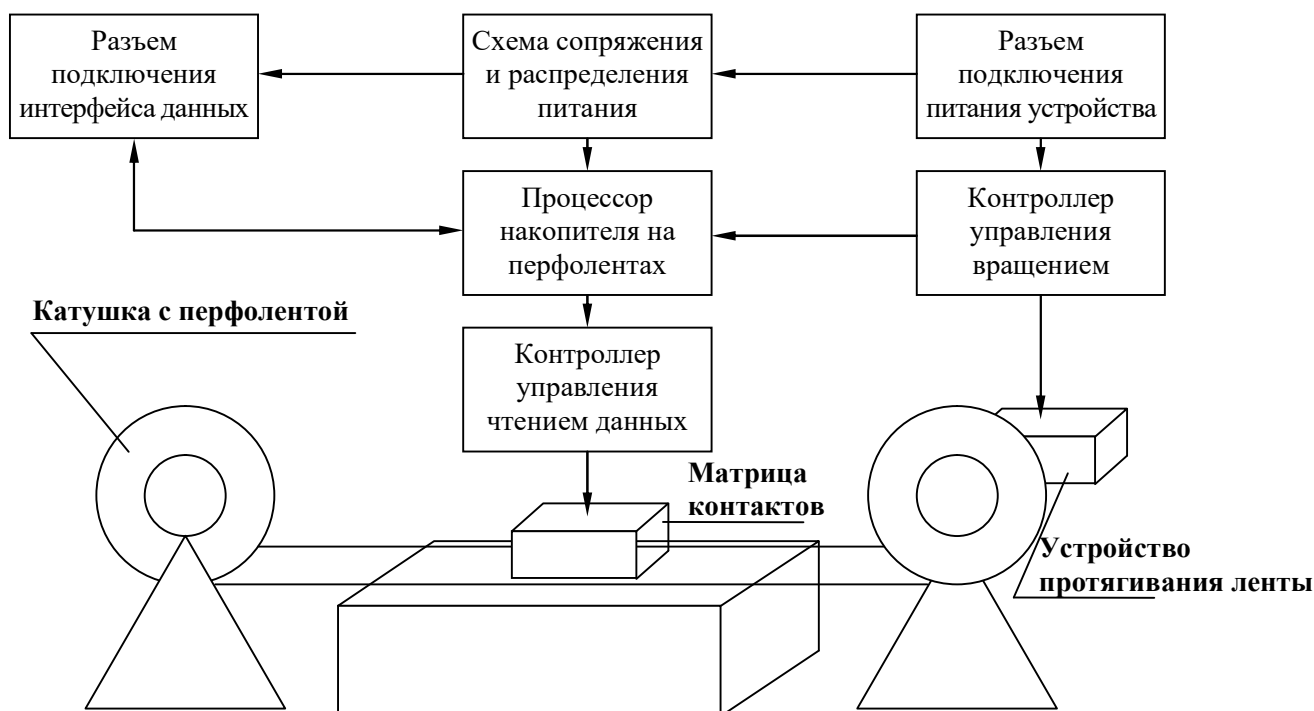


Рисунок 5.13. Структура накопителя информации на основе перфорирующей ленты

Структурными элементами накопителя информации на основе перфоленты являются:

- механические компоненты, которые реализуют поддержку протягивания ленты:
 - катушка с перфолентой – содержит намотанную на катушку бумажную ленту;
 - коммутационная матрица контактов – обеспечивает непосредственную коммутацию разных контактов в случае попадания пробитого отверстия между ними;
 - устройство протягивания ленты – обеспечивает натяжение и протяжку ленты;
- электронные компоненты, которые обеспечивают управление процессом доступа:
 - разъем подключения питания устройства – обеспечивает энергоснабжение;
 - разъем подключения интерфейса данных – реализует обмен информацией;
 - схема сопряжения и распределения питания – реализует понижение и подачу опорного напряжения для поддержки функционирования электронных элементов;
 - процессор накопителя информации на перфолентах – позволяет реализовать управление обработкой потоком непрерывно поступающих данных;
 - контроллер управления чтением данных – реализует непрерывную регистрацию состояний при коммутации элементов коммутационной матрицы.

5.7.3. Накопители информации на магнитных лентах

Накопители информации на магнитных лентах содержат долговечную ленту.

Определение

Накопителем информации на магнитных лентах (Streamer) называют устройство управления чтением-записью данных на магнитную ленту посредством ее протягивания с определенной постоянной скоростью относительно блока головок чтения-записи.

Катушка с магнитной лентой имеет разный размер и объем хранимой информации.

Накопитель информации на магнитных лентах (рис. 5.14) обеспечивает возможность хранения небольшого объема информации с малой скоростью чтения-записи данных.

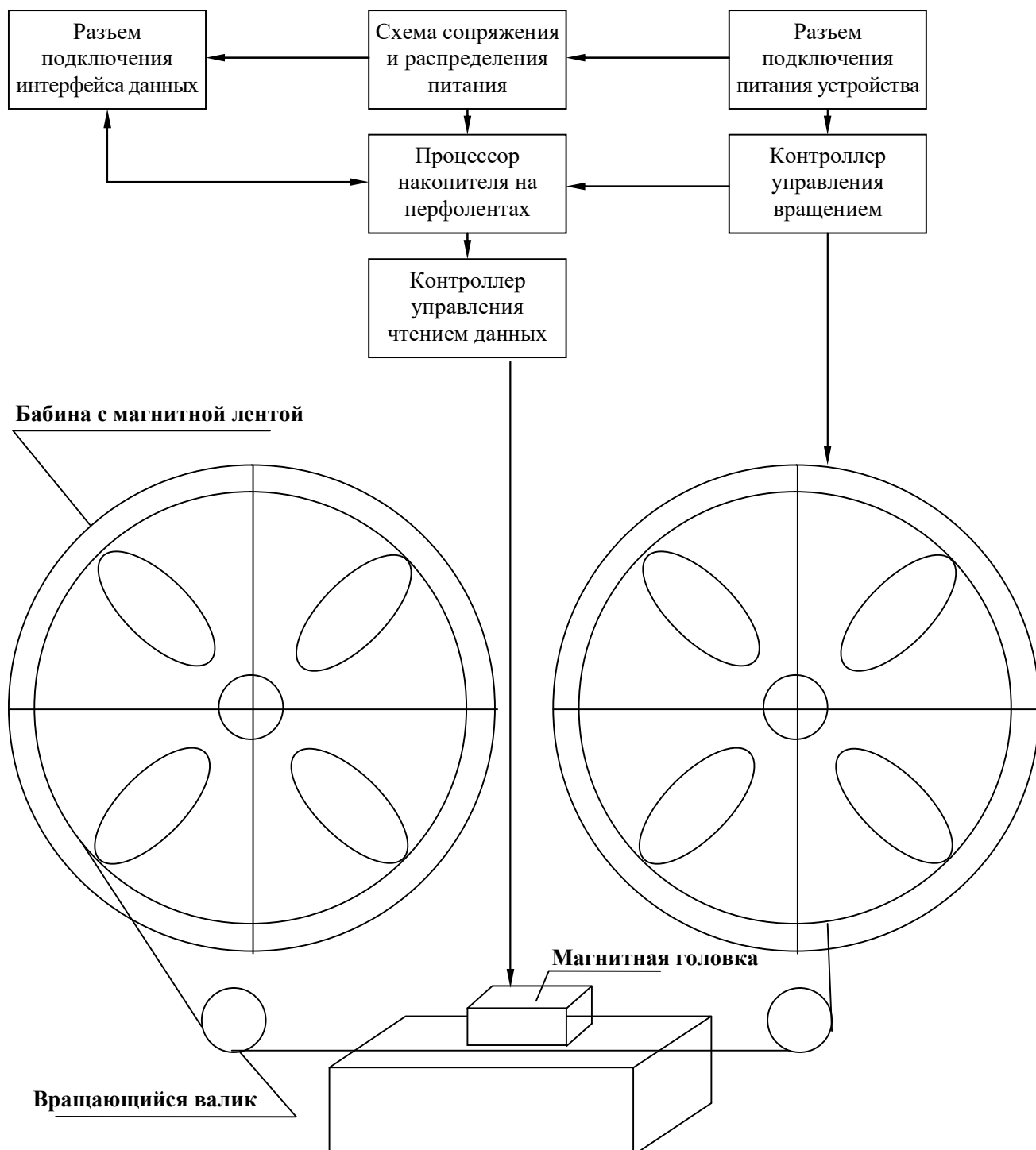


Рисунок 5.14. Структура накопителя информации на основе перфорирующей ленты

5.7.4. Накопители информации на гибких магнитных дисках

Определение

Накопитель информации на гибких магнитных дисках (FDD – Floppy Disc Drive) конструктивно исполнен в виде двух основных компонентов: устройство чтения-записи информации (дисковод) и носитель информации (дискета с магнитным диском).

Определение

Дискета с магнитным гибким диском (рис. 5.15) представляет собой пластиковый корпус с задвижкой для защиты от случайной записи (если задвижка передвинута к внешней грани корпуса, то запись заблокирована), в котором располагается пластина круглой формы с металлическим сердечником, имеющая одну или две эффективные поверхности для записи информации посредством использования электромагнитных сигналов, изготовленную из гибкого лавсана, обладающего свойствами ферромагнетика и способного регистрировать два различных устойчивых магнитных состояния (уровня намагниченности), соответствующих логическому 0 или 1.

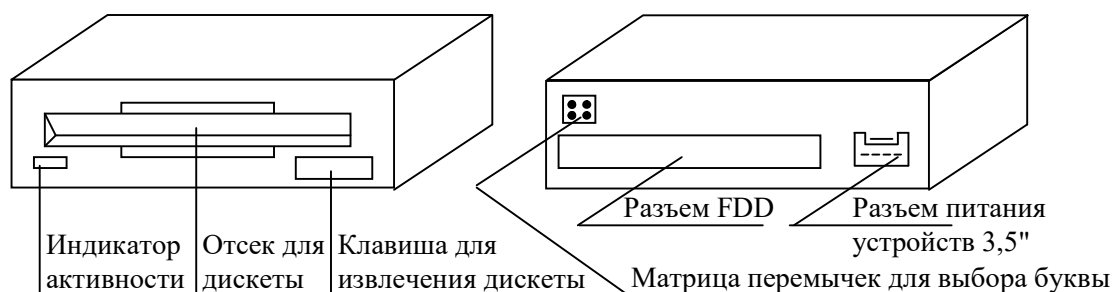


Рисунок 5.15. Особенности структуры и подключения накопителя информации на гибких магнитных дисках

На представленной схеме (рис. 5.15) различные структурные элементы накопителя информации на гибких магнитных дисках выполняют при работе разные функции:

- компоненты, предназначенные для использования конечным пользователем:
 - индикатор активности – светодиодный индикатор состояния записи-чтения;
 - отсек для дискеты – установка магнитного диска в пластиковом корпусе;
 - клавиша для извлечения дискеты – нажатие инициирует извлечение диска;
- компоненты для использования квалифицированным специалистом, который осуществляет подключение накопителя информации к материнской плате:
 - матрица перемычек (jumpers) для выбора буквы – позволяет осуществить выбор буквы логического диска при использовании накопителя информации в операционной системе: "А" – положение 1 (перемычка находится в вертикальном положении и замкнуты первые два контакта), "В" – положение 2);
 - разъем FDD – предназначен для интерфейсного соединения накопителя на гибком магнитном диске с разъемом FDD, который расположен на материнской плате посредством использования плоского коммутационного шлейфа;
 - разъем питания – обеспечивает энергоснабжение накопителя информации на гибком магнитном диске посредством подключения вилки питания (+5В, G – ноль, G, +12В) от блока питания в системном блоке компьютера.

Схема накопителя информации на гибких магнитных дисках показана рис. 5.16.

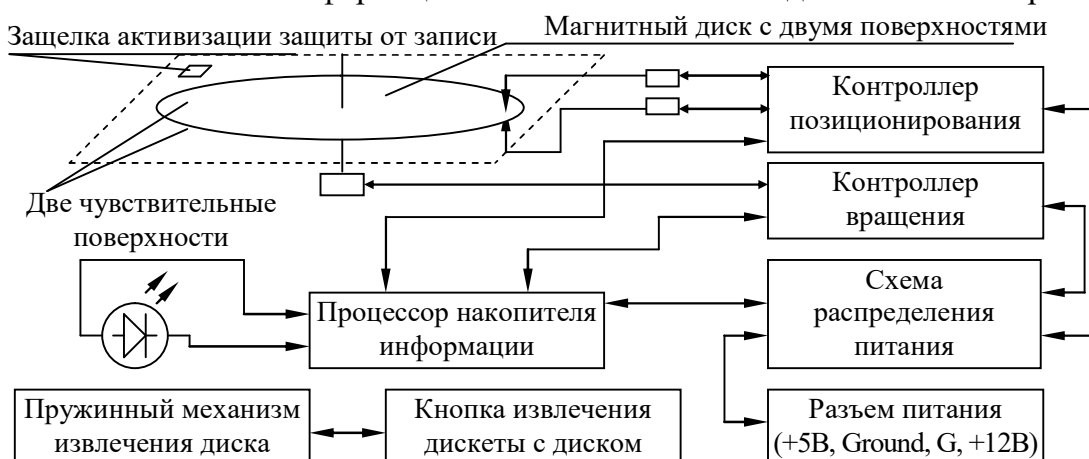


Рисунок 5.16. Функциональная схема накопителя на гибких магнитных дисках

Структура накопителя информации на гибких магнитных дисках включает:

- механические компоненты, позволяющие реализовать использование дискеты:
 - кнопка извлечения дискеты с магнитным диском – нажатие на кнопку позволяет извлечь дискету с магнитным диском с целью его последующей замены;
 - пружинный механизм для извлечения диска – реализует защелкивание при установке дискеты в отсек дисководов и последующее ее извлечение при нажатии кнопки обеспечивающей извлечение дискеты с магнитным диском;
 - защелка активации защиты от записи – позволяет реализовать защиту дискеты от записи данных с целью защиты имеющейся информации посредством перемещения защелки (jumper) в сторону внешней грани корпуса дискеты;
- электрические компоненты, которые реализуют управление чтением-записью:
 - схема распределения питания – позволяет реализовать развязку питающего напряжения, которое поступает на разъем питания (+5В, G – ноль, G, +12В) между внутренними компонентами накопителя информации на гибких магнитных дисках с целью обеспечения процесса их функционирования;
 - контроллер позиционирования – реализует управление позиционированием головок чтения-записи в процессе функционирования накопителя информации;
 - контроллер вращения – управляет вращением оси накопителя информации;
 - процессор накопителя информации – реализует мониторинг состояния и управление контроллером позиционирования, контроллером вращения и светодиодным индикатором в процессе чтения-записи данных на дискету;
 - светодиодный индикатор – регистрирует активность накопителя информации при осуществлении операций чтения или записи данных на дискету.

Головки чтения-записи информации могут синхронно перемещаться в горизонтальном и вертикальном направлении (обозначены стрелками), что позволяет позиционировать их относительно любой точки поверхности диска, а каждая точка поверхности магнитного диска способна регистрировать два устойчивых состояния уровня намагниченности, которые соответствуют двум логическим состояниям (1 бит информации): 1 – состояние логической единицы, 0 – состояние логического нуля.

5.7.5. Накопители информации на жестких магнитных дисках

Накопитель информации на жестких магнитных дисках конструктивно является неразборным устройством, которое обеспечивает параллельное чтение и запись информации на несколько магнитных поверхностей каждого диска в его основе.

Определение

Накопитель информации на жестких магнитных дисках (HDD – Hard Disc Drive) конструктивно исполнен в виде единого устройства (рис. 5.17), которое включает ось (шпиндель) с одним или несколькими магнитными дисками, содержащими две эффективные поверхности для записи и чтения информации, а также интерфейс подключения к материнской плате компьютера (слоты IDE ATA, SCSI, IDE SATA).

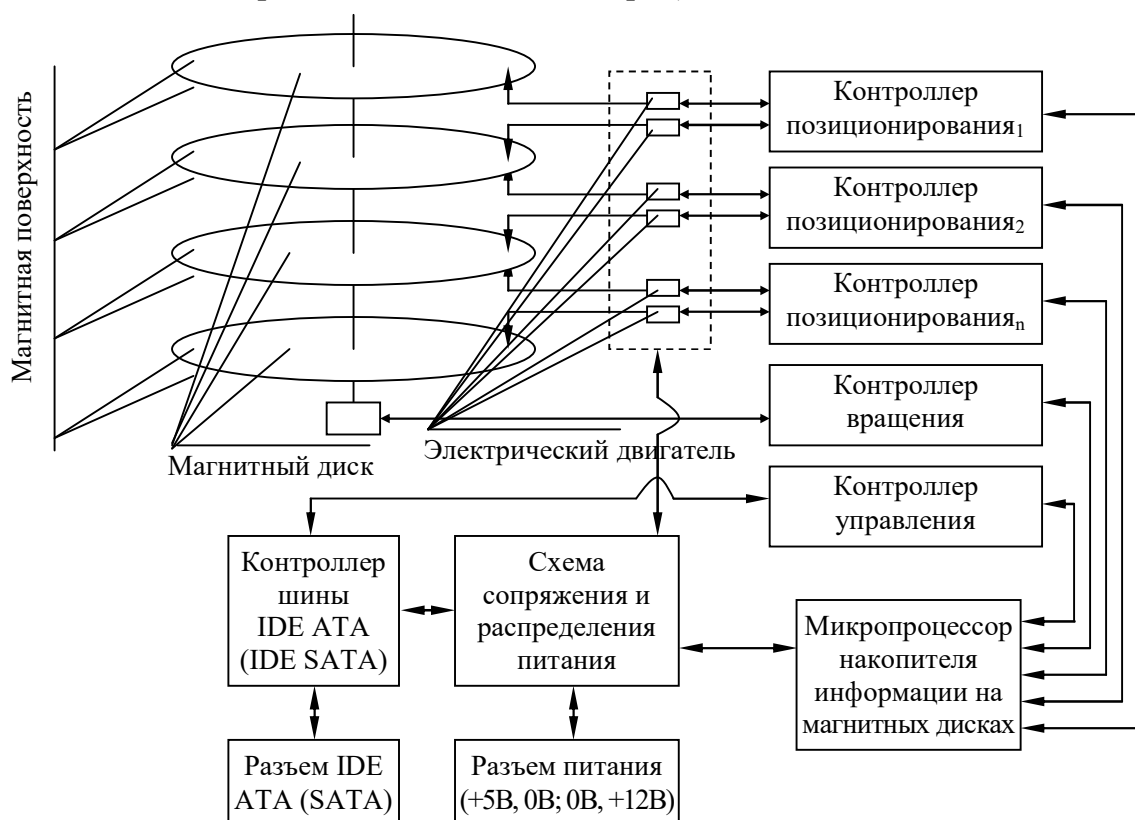


Рисунок 5.17. Особенности структуры и подключения накопителя информации на жестких магнитных дисках

Контроллеры позиционирования – обеспечивают позиционирование магнитных головок чтения-записи относительно чувствительной поверхности магнитного диска.

Контроллер вращения – реализует мониторинг и управление вращением шпинделя.

Контроллер управления – регистрация состояния и управление потоками управляющих сигналов в накопителе информации на жестких магнитных дисках.

Микропроцессор накопителя информации на жестких магнитных дисках – реализует мониторинг состояния, синхронизацию управляющих сигналов, а также управление всеми потоками данных между различными компонентами устройства.

Контроллер шины IDE (S)ATA или SCSI – реализует интерфейс взаимодействия с материнской платой или платой расширения содержащей слот стандарта SCSI.

Схема сопряжения и распределения питания – питание всех компонентов.

Накопители информации на магнитных дисках подразделяются на несколько основных видов и типов в зависимости от особенностей исполнения их конструкции:

1. При рассмотрении месторасположения и способа подключения к системному блоку:
 - внешние накопители информации – выступают мобильными устройствами, которые подключаются к системному блоку посредством внешних разъемов;
 - внутренние накопители информации – располагаются внутри системного блока и подключаются к материнской плате посредством разъемов и шлейфа;
2. При рассмотрении конструктивных особенностей исполнения устройства:
 - накопитель информации на гибких магнитных дисках – конструктивно включает два компонента: устройство чтения-записи данных (дисковод) и магнитный диск, который расположен в пластиковом корпусе (дискета);
 - накопитель информации на жестких магнитных дисках – представляет собой конструктивно единое устройство, которое сочетает функции устройства чтения-записи данных и содержит один либо несколько магнитных дисков, расположенных на единой вращающейся оси или шпинделе в устройстве.

Накопители информации на магнитных дисках имеют в своей основе магнитный диск в качестве носителя информации, на который осуществляется непосредственная запись данных и их последующее чтение. Магнитные диски разных производителей могут существенно отличаться друг от друга по химическому составу покрытия, которое способно регистрировать два различных состояния намагниченности соответствующие двум логическим состояниям: 1 – логическая единица, 0 – логический ноль.

Выделяется ряд элементов физической структуры магнитного диска:

- **стороны диска** – соответствуют двум поверхностям способным регистрировать два устойчивых магнитных состояния для каждого магнитного диска, они нумеруются, начиная с нуля: для винчестера нулевой номер имеет верхняя поверхность, а для гибкого диска нулевой номер у «лицевой» стороны дискеты;
- **дорожка** – концентрическая окружность, по которой движутся головки чтения-записи при записи, чтении или поиске данных, они нумеруются, начиная с нуля, а нулевой номер присваивается внешней дорожке на каждом магнитном диске;
- **цилиндр** – представляет собой совокупность дорожек с одинаковыми номерами на разных сторонах диска, а номера цилиндров совпадают с номерами дорожек;
- **кластер** – совокупность секторов, которые имеют смежные номера: может состоять из одного сектора (для дискет) или нескольких (для винчестера).
- **секторы** – блоки информационных элементов, в которых размещаются данные на дорожке при записи информации, они нумеруются, начиная с единицы, но помимо пользовательской информации (непосредственно файлы и папки пользователя и операционной системы, а также данные файловой системы), сектора содержат служебную информацию (собственный номер) и являются минимальными адресуемыми элементами данных для каждого магнитного диска;

При осуществлении процедуры записи информации на магнитный диск необходимо максимально исключить возможность возникновения фрагментации файлов.

Определение

Фрагментация файловой системы – называется процесс изменения содержания файловой системы, который обусловлен записью данных (файлов и папок) в несмежную группу информационных элементов физической (сектор, кластер) или логической (таблица размещения файлов) структуры магнитного диска в основе накопителя.

Фрагментация файловой системы обуславливает существенное увеличение временных издержек при обращении к магнитному диску с целью осуществления поиска, чтения или записи информации, что оказывает ощутимое влияние на продуктивную работу любого пользователя компьютера при выполнении ряда прикладных задач.

Определение

Дефрагментация файловой системы – осуществляется посредством использования специальных программных средств, которые реализуют изменение физической и логической структуры магнитного диска в основе накопителя информации с целью размещения папок и файлов в смежной группе информационных элементов.

Операционная система компьютера позволяет осуществить управление доступом к файлам и папкам только посредством использования файловой системы определенного типа (FAT – MS DOS, FAT32 – MS DOS и Windows, NTFS – Windows XP).

Для вновь приобретенного физического накопителя информации необходимо:

1. Разметить накопитель информации посредством утилиты FDisc и создать при этом:
 - основной раздел в основе логической организации данного накопителя информации под который можно выделить все имеющееся пространство накопителя информации или часть в процентном соотношении от его общего объема;
 - дополнительный раздел в основе логической организации данного накопителя информации, который предполагает создание или удаление нескольких логических дисков разного объема исчисляемого в процентном соотношении относительно общего объема ранее созданного дополнительного раздела диска;
2. Осуществить форматирование всех созданных логических дисков во всех разделах:
 - отформатировать единственный логический диск в созданном основном разделе;
 - обеспечить форматирование всех логических дисков в дополнительном разделе.

Определение

Форматирование магнитного диска – достигается при помощи утилит и специальных программ, которые позволяют создать логическую организацию в виде файловой системы как надстройку над физической структурой магнитного диска.

Форматирование всех логических дисков в основном и дополнительном разделах накопителя информации осуществляется посредством утилиты format.com:

- в операционной системе MS-DOS при помощи командного интерфейса;
- в операционной системе MS Windows посредством использования режима эмуляции сеанса MS-DOS или вкладки «Мой компьютер» на рабочем столе.

Логическая структура накопителя информации характеризуется созданием и использованием определенной файловой системы и представляется следующим образом (для FAT и FAT32 – File Allocation Table, таблица размещения файлов):

- **таблица разделов PT (Partition Table)** – состоит из четырех сегментов, которые описывают разделы диска, причем операционные системы используют только первые два сегмента, а описание раздела диска содержит данные о первых и последних секторах диска, его дорожках, секторах раздела, общем количестве секторов в разделе, типе файловой системы и также содержит признак загрузочного раздела (без него загрузка операционной системы не представляется возможной);
- **главная загрузочная запись MBR (Master Boot Record)** – содержит информацию о месторасположении программы загрузчика, которой передается управление для обеспечения дальнейшей загрузки операционной системы;
- **загрузочная запись операционной системы BR (Boot Record)** – содержит служебную информацию: программу загрузки операционной системы (загрузчик), размер кластера, количество копий таблицы размещения файлов (FAT), перечень папок и файлов в корневом каталоге диска Root, размер таблицы размещения файлов и некоторую дополнительную информацию;
- **таблица размещения файлов FAT (File Allocation Table) и ее копии** – содержит полную карту принадлежности кластеров файлам и папкам, используется операционными системами для хранения сведений о размещении файлов на диске, а также содержит данные о количестве кластеров непригодных для записи и чтения данных (bad – clusters), а поскольку таблица размещения файлов (FAT) имеет исключительно важное значение, то она дублируется несколько раз в разных местах с учетом логической структуры магнитного диска;
- **корневой каталог Root** – выступает структурой данных, в которой каждая запись соответствует файлу или подкаталогу на магнитном диске и подчиняется корневому каталогу определенного логического диска, при этом он включает:
 - имя файла или подкаталога – содержит два структурных элемента для FAT и FAT32: имя файла как таковое имеет 8 символов (буквы латинского и национального алфавита, а также цифры и некоторые служебные знаки) и расширение включающее 3 символа (позволяет определить тип файла в файловой системе и связанное с ним базовое приложение в операционной системе MS-DOS, MS Windows 3.11 for Workgroups, 98, Me, 2000, XP, Vista);
 - атрибуты, в которых определяются следующие параметры файла или подкаталога: спецификатор типа файла архивный (подлежит архивации) – A[File Ready To Archiving], только для чтения – R[ead only], скрытый – H[idden], системный – S[ystem], признак каталога – D[irectory];
 - время создания – время создания и появления файла в файловой системе;
 - дата изменения – дата внесения последних изменений в данном файле;
 - размер файла – отражает объем дисковой памяти на накопителе информации.

5.8. Лазерные накопители информации на оптических дисках

Повсеместное использование лазерных накопителей информации на основе оптических дисков обусловлено рядом их инновационных отличительных свойств:

- исключение металлических компонентов и как следствие очень малая масса;
- отсутствие большого количества механических передач, механизмов и электрических двигателей, что обеспечивает максимальную отказоустойчивость;
- низкий уровень генерируемого шума, поскольку отсутствует работа, которая совершается электрическими двигателями для механического вращения шпинделей с магнитными дисками, обладающими свойствами ферромагнетика;
- отсутствие нескольких исполнительных механизмов (механических двигателей) и контроллеров, которые обеспечивают позиционирование магнитных головок чтения-записи, что позволяет достичь низкого энергопотребления при работе;
- технология изготовления дисков полностью исключает наличие ферромагнитного покрытия и реализуется инвариантность к воздействию магнитного поля;
- относительно большой объем хранимой информации на единицу площади;
- чрезвычайно высокая компактность при хранении нескольких оптических дисков, хотя существенно затрудняется поиск диска при росте их количества.

Определение

Лазерный накопитель на оптических дисках (CD-ROM и DVD-ROM) конструктивно исполнен в виде интегральной совокупности двух независимых устройств: устройство чтения-записи информации (дисковод) и концентрическая окружность с отверстием в центре содержащая одну или две оптические поверхности (оптический диск), которые предназначены для чтения и записи разной информации.

Определение

Оптический диск для однократной (Recordable) или многократной (ReWritable) записи данных – это носитель информации, представляющий собой концентрическую окружность, которая содержит отверстие в середине и одну (вторая при этом содержит этикетку) или две эффективные поверхности из материала, который обладает свойствами изменения коэффициента преломления направленного пучка при чтении информации, а также способностью однократно или многократно (предусматривает цикл стирания) регистрировать два устойчивых логических состояния (1 и 0) посредством использования интегрированного полупроводникового лазера.

Накопитель информации на оптических дисках реализует возможность записи и перезаписи данных посредством использования специального программного обеспечения:

- формат записи данных на оптический диск: аудио-поток (Audio-CD), данные (Data-CD), видео-поток (Video-CD), гибридный (Data+Video-CD, Data+Audio-CD);
- режим записи оптического диска: стирание (Erase-CD), запись данных (Record), динамическая перезапись данных (CD-RAM) с проверкой ошибок и защитой от опустошения буфера записи лазерного накопителя на оптических дисках (Burn Proof);
- режим проверки поверхности и файловой системы записанного оптического диска.

5.8.1. Лазерные накопители информации на оптических дисках CD-ROM

Лазерный накопитель информации на оптических дисках стандарта CD-ROM позволяет реализовать запись и последующее чтение данных разного формата.

Определение

Оптический диск стандарта CD-ROM содержит одну (SS – Single Side, а вторая, как правило, тогда содержит этикетку) или две (Double Side) эффективные поверхности для чтения и записи информации, которые чувствительны к направленному пучку лазерного излучения, что позволяет зарегистрировать два устойчивых логических состояния (0 – логический ноль и 1 – логическая единица) посредством первичного изменения (запись) и измерения (чтение) коэффициента преломления.

От количества эффективных поверхностей оптического диска, которые предназначены для записи информации существенно зависит объем хранимой информации:

- односторонний оптический диск (SS-CD-ROM) – содержит одну поверхность для записи данных, а общий объем информации составляет 650-800Мб;
- двухсторонний оптический диск (DS-CD-ROM) – общий объем 1,3-1,6Гб.

В зависимости от особенностей технологического процесса изготовления оптического диска и специфики цикла записи данных на оптический диск посредством дисководов и специального программного обеспечения можно выделить диски:

- записываемый оптический диск (CD-Recordable) – предназначен для однократного цикла записи информации на его чувствительную поверхность;
- перезаписываемый оптический диск (CD-ReWritable) – предусматривает два технологических цикла в рамках многократной записи данных на диск:
 - стирание информации (Erasing) – достигается восстановление оптических свойств (преломление) чувствительной поверхности оптического диска;
 - запись информации (Recording) – достигается регистрация двух устойчивых логических состояний на поверхности оптического диска посредством направленного пучка, который генерируется полупроводниковым лазером;
 - чтение информации (Reading) – реализуется мониторинг изменения коэффициента преломления для считывания информации с поверхности оптического диска посредством использования лазерной головки и механического привода, которые располагаются в едином корпусе накопителя.

На рис. 5.18 представлена структура стандартного оптического диска стандарта CD-ROM, который включает несколько рабочих поверхностей в едином объеме.

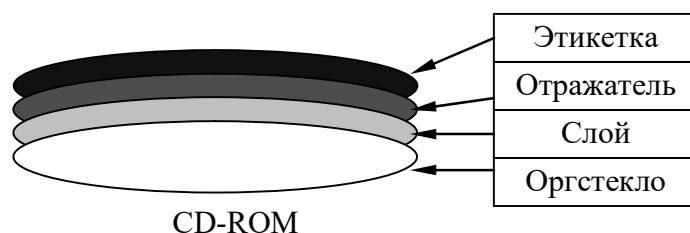


Рисунок 5.18. Структура оптического диска стандарта CD-ROM как носителя информации

5.8.2. Лазерные накопители информации на оптических дисках DVD-ROM

Лазерный накопитель информации на оптических дисках стандарта DVD-ROM позволяет реализовать однократную или многократную запись и последующее чтение разнородных данных на оптические диски разного стандарта и технологии изготовления.

Определение

Оптический диск стандарта DVD-ROM аналогичен оптическому диску стандарта CD-ROM, содержит одну (SS – Single Side, а вторая, как правило, содержит этикетку) или две (Double Side) эффективные поверхности для чтения и записи информации, каждая из которых содержит один (SL – Single Layer) или два (DL – Double Layer) чувствительных слоя к направленному пучку лазерного излучения.

От количества сторон или поверхностей диска и количества слоев в объеме каждой поверхности диска зависит эффективный объем хранимой информации:

- SS, SL – односторонний, однослойный оптический диск объемом 4,7Гб;
- SS, DL – односторонний, двухслойный оптический диск объемом 8,5Гб;
- DS, SL – двухсторонний, однослойный оптический диск объемом 9,4Гб;
- DS, DL – двухсторонний, двухслойный оптический диск объемом 17,1Гб.

В рамках технологического процесса изготовления оптических дисков DVD-ROM стандартов SS или DS (одна или две стороны), либо SL или DL (один или два слоя) возможны незначительные модификации, которые влияют на объем информации.

Структура оптических дисков для лазерных приводов имеет существенные отличия, которые зависят от количества чувствительных слоев и поверхностей (рис. 5.19).

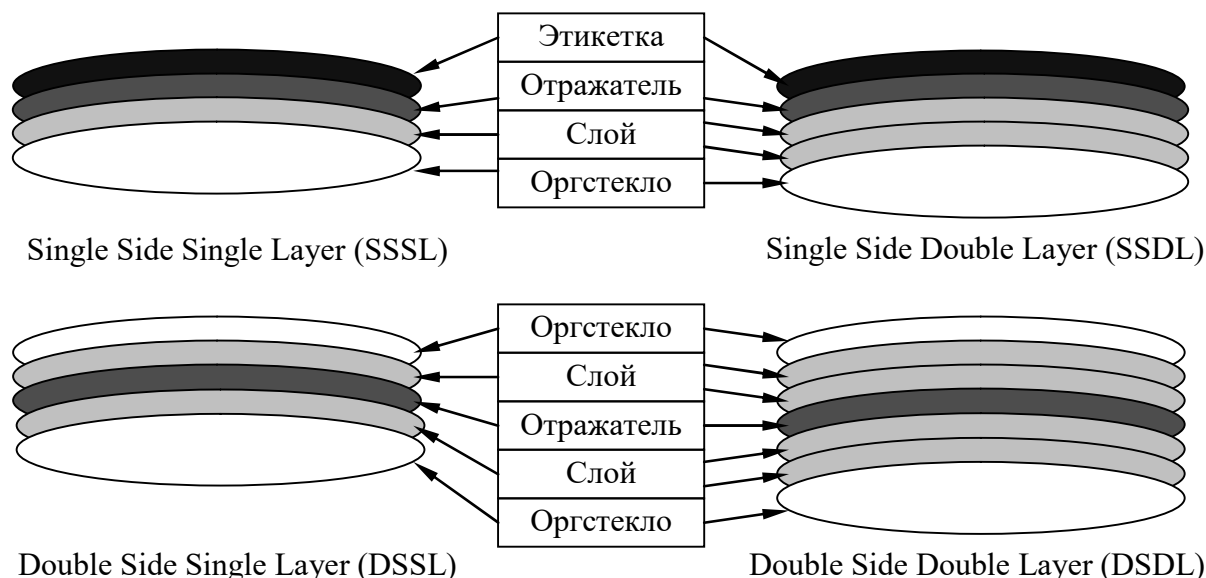


Рисунок 5.19. Структура оптического диска DVD-ROM как носителя информации

Оптические диски стандарта DVD-ROM аналогично оптическим дискам стандарта CD-ROM дифференцируются в зависимости от особенностей технологического цикла записи информации на их чувствительную оптическую поверхность:

- оптический диск для однократной записи данных (DVD-Recordable);
- оптический диск для многократной записи данных (DVD-ReWritable).

5.9. Электронные накопители информации

Электронный накопитель информации выступает принципиально новым видом устройств хранения данных, позволяя реализовать запись и последующее чтение данных, а также обладает рядом исключительных отличительных свойств:

- конструктивно абсолютно исключает возможность механического повреждения поверхности, посредством которой осуществляется запись и последующее чтение данных, поскольку не содержит механических узлов и передач:
 - накопитель информации на перфокартах и перфоленте содержит недолговечную бумажную карту прямоугольной формы, которая очень быстро изнашивается и соответственно бумажную ленту – подвержена разрыву при натягивании в устройствах считывания и пробивания поверхности носителя;
 - накопитель информации на магнитных лентах также содержит быстро изнашиваемую магнитную ленту, которая разрывается при натягивании в стримере, а также подвержена воздействию магнитного поля, что обуславливает невозможность четко различать два уровня намагниченности или два логических состояния при последующей попытке чтения данных;
 - накопитель информации на магнитных дисках содержит один или несколько магнитных дисков с чувствительными поверхностями, которые поступательно вращаются вокруг своей оси и содержат подвешенные головки чтения-записи информации за счет использования электрических двигателей и механических передач, а также электрических контроллеров и схем сопряжения, поэтому они подвержены воздействию магнитного поля, механическому повреждению даже вследствие износа в течение срока полезного использования и электрическому повреждению вследствие возникновения импульсных помех в силовой сети питания 110-220 В;
- обладают существенно низким энергопотреблением вследствие использования энергосберегающих технологий в основе аппаратного обеспечения;
- имеют малые габаритные размеры и очень большой объем хранимых данных;
- технологически предусматривают возможность использования микросхем памяти с очень низким показателем времени упреждающего доступа при поиске, а также минимальное время чтения и записи пакета данных на носитель.

Выделяют несколько основных видов электронных накопителей информации:

- внутренний электронный накопитель информации – представляет собой неразборное устройство, которое размещается внутри системного блока, подключается посредством разъема IDE ATA или IDE SATA, выступает эквивалентом накопителя на жестких магнитных дисках (HDD – Hard Disc Drive), но не содержит механических частей, потребляет меньше электрической энергии;
- внешний электронный накопитель информации – представляет собой разборное (карта памяти и модуль для чтения карт памяти) или неразборное (носимая карта памяти с разъемом USB – Flash Drive) устройство внешней памяти.

5.9.1. Внутренний электронный накопитель информации

Внутренний электронный накопитель информации (рис. 5.20) выступает эквивалентной заменой накопителя информации на жестких магнитных дисках, поскольку позволяет обеспечить существенное энергосбережение и наивысшую отказоустойчивость, поскольку в нем абсолютно отсутствуют механические узлы и сопрягающие механизмы, которые вращаются под действием электрических двигателей, осуществляющих работу по перемещению магнитных головок чтения-записи над поверхностями магнитного диска и обеспечивающих вращение оси (шпинделя).

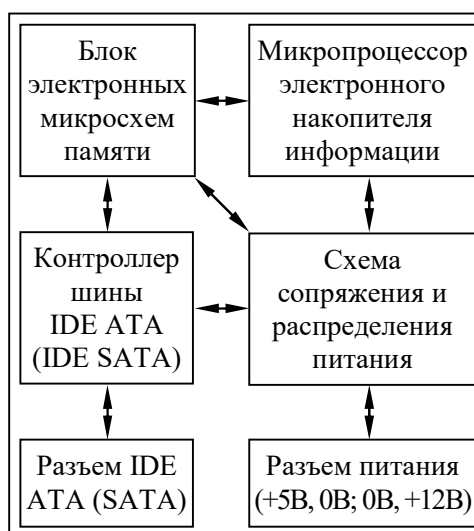


Рисунок 5.20. Структура внутреннего электронного накопителя информации

Внутренний электронный накопитель информации содержит ряд элементов:

- блок электронных микросхем памяти – представлен одной, двумя или целой батареей микросхем памяти, которые реализуют носитель информации;
- микропроцессор электронного накопителя информации – реализует мониторинг состояния и управление потоками данных и сигналов, которые передаются между интерфейсом и компонентами электронного накопителя информации;
- контроллер шины IDE (S)ATA, SCSI – реализует интерфейс взаимодействия между разъемом IDE (S)ATA или SCSI на электронном накопителе информации и соответствующим разъемом на материнской плате компьютера;
- схема сопряжения и распределения питания – реализует энергоснабжение компонентов электронного накопителя информации за счет распределения различных номиналов напряжения от его разъема питания, к которому подключается вилка питания (+5В, 0В; 0В, +12В) содержащаяся в блоке питания системного блока компьютера (мощность блока питания зависит от конфигурации);
- разъем IDE (S)ATA – обеспечивает возможность подключения коммутационного шлейфа для реализации интерфейса взаимодействия между электронным накопителем информации и материнской платой компьютера (ЭВМ);
- разъем питания (+5В, Ground; G – ноль, +12В) – позволяет подключить вилку питающего кабеля, которая содержится в блоке питания компьютера.

Определение

Внутренним электронным накопителем информации называют интегральную совокупность устройства чтения записи информации (контроллер) и набор микросхем памяти (носитель информации), которые обеспечивают сохранение и извлечение данных пользователя в процессе работы на ЭВМ, при этом интерфейс взаимодействия реализован посредством контроллера шины и разъема IDE (S)ATA, SCSI.

Структурная схема подключения и расположения внутреннего электронного накопителя информации в системном блоке компьютера представлена на рис. 5.21.

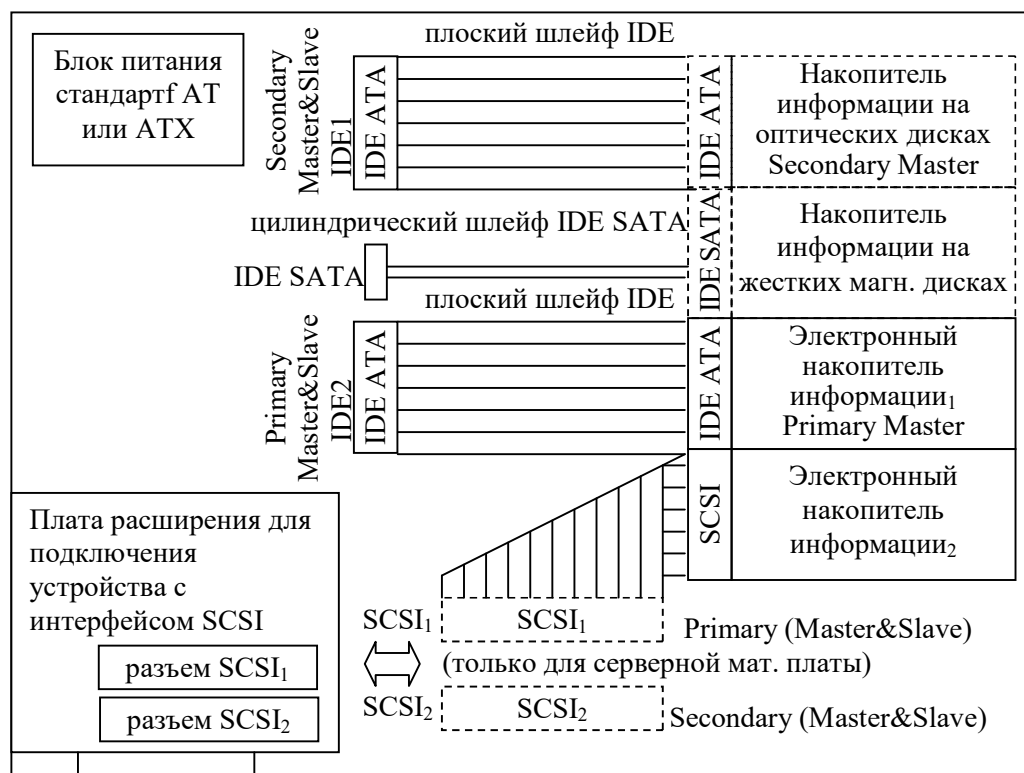


Рисунок 5.21. Схема расположения внутреннего электронного накопителя информации

Внутренний электронный накопитель информации подключается посредством:

- разъема IDE ATA – разъем IDE ATA на материнской плате соединяется с разъемом IDE ATA на внутреннем электронном накопителе информации посредством плоского многоконтактного коммутационного шлейфа IDE ATA;
- разъема IDE SATA – разъем IDE SATA на материнской плате соединяется с разъемом IDE SATA на внутреннем электронном накопителе информации посредством использования цилиндрического коммутационного шлейфа IDE SATA;
- разъема SCSI – разъем SCSI на материнской плате соединяется с разъемом SCSI на внутреннем электронном накопителе информации посредством использования плоского многоконтактного коммутационного шлейфа SCSI:
 - разъем SCSI на серверной материнской плате непосредственно реализует подключение любого внутреннего электронного накопителя информации;
 - на обычной материнской плате осуществляется установка платы расширения через слот PCI с разъемами USB, через которые непосредственно реализуется подключение внутреннего электронного накопителя информации к компьютеру.

5.9.2. Внешний и носимый электронный накопитель информации

Внешний электронный накопитель информации предназначен для реализации сохранения и чтения данных, при этом обладает относительно высокой мобильностью и низким весом, уровнем шума и энергопотреблением при его использовании.

Определение

Внешним электронным накопителем информации называют интегральную совокупность устройства чтения записи информации (контроллер) и набор микросхем памяти (носитель информации) обеспечивающие сохранение и извлечение данных пользователя, при этом интерфейс взаимодействия реализован посредством корневого концентратора шины USB (Universal Serial Bus) и разъема USB.

Схема подключения внешнего электронного накопителя информации посредством использования модуля с внешними разъемами USB, платы с внешними разъемами USB (используется название плата типа «выкидыш» USB) или платы расширения для подключения устройства с интерфейсом USB представлена на рис. 5.22.

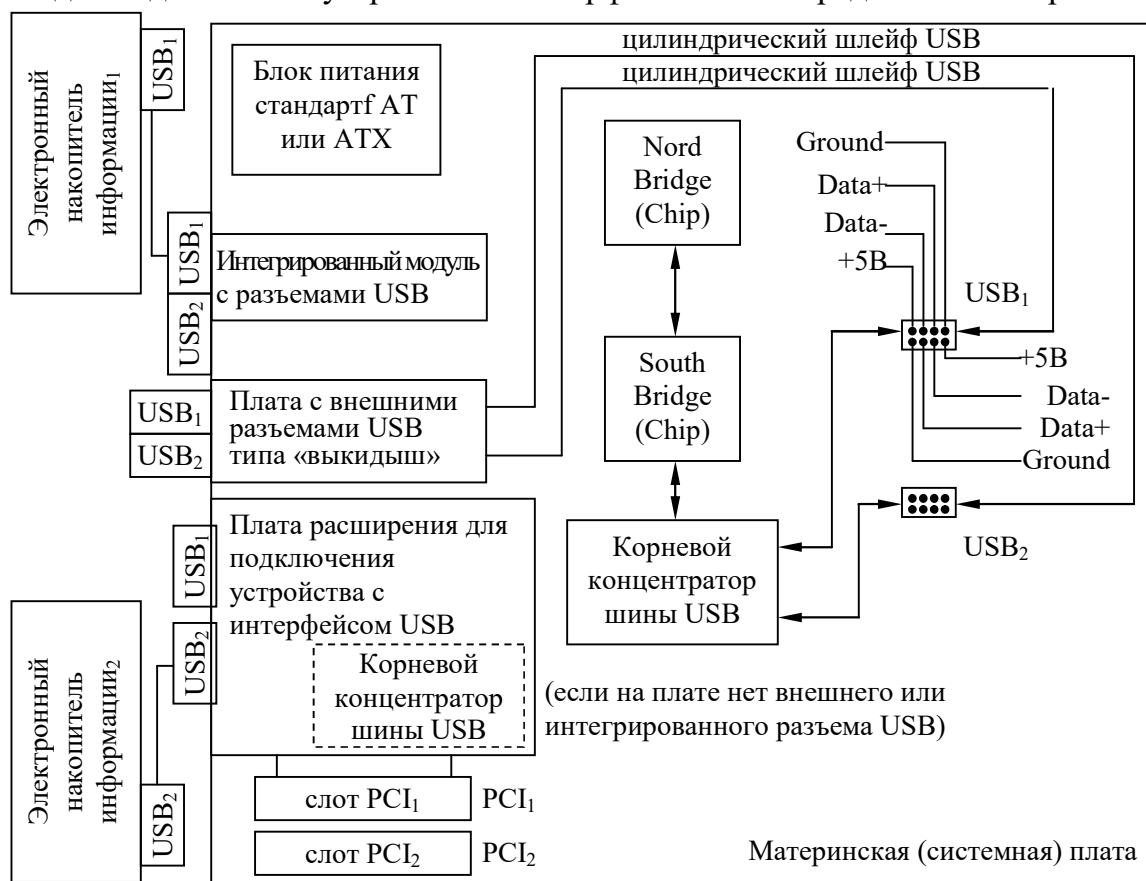


Рисунок 5.22. Схема внешнего электронного накопителя информации

Корневой концентратор шины USB предназначен для обслуживания устройств с интерфейсом USB: электронные накопители информации, принтеры, сканеры.

Модуль с внешними разъемами USB интегрирован непосредственно на материнскую плату и доступен непосредственно на задней панели системного блока.

Плата с внешними разъемами USB расположена в задней панели системного блока и прикрепляется винтом или интегрирована в переднюю панель системного блока, а разъемы на материнской плате и плате расширения соединяются шлейфом.

Определение

Внешним носимым электронным накопителем информации называют компактно выполненное устройство, которое включает микроконтроллер и одну или две микросхемы памяти, обеспечивающие чтение и запись информации посредством интерфейса взаимодействия между компьютером при подключении к разъему USB.

Внешний электронный накопитель информации выполняется в виде двух независимых типов устройств, которые подключаются к системному блоку компьютера:

- съемный внешний электронный накопитель информации – выполняется в виде конструктивно единого устройства, которое одновременно содержит устройство чтения-записи данных и блок электронных микросхем памяти, при этом возможны несколько модификаций: портативное устройство для хранения больших объемов информации (External Drive) или компактное устройство для хранения малых объемов информации (External Flash Drive);
- съемный компактный носимый электронный накопитель информации – представляет собой два независимых устройства: устройство чтения-записи данных (Card Reader) и блок электронных микросхем памяти (Memory Card).

Структура внешнего накопителя информации представлена на рис. 5.23.

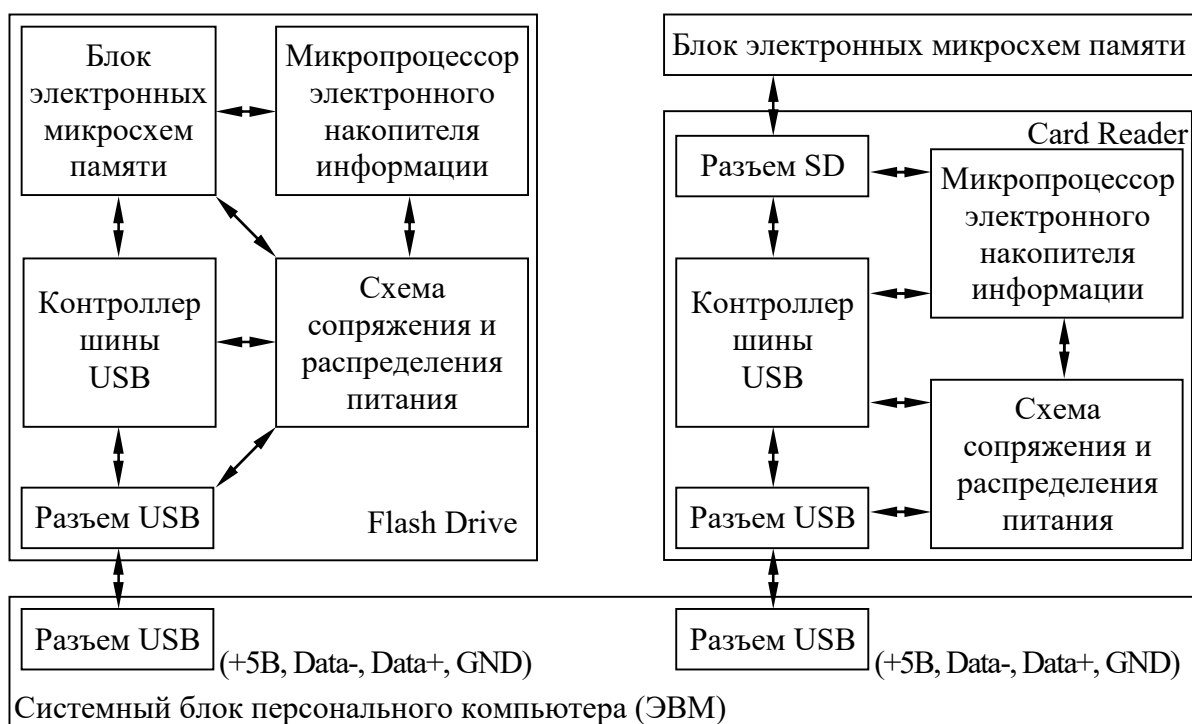


Рисунок 5.23. Структура внешнего электронного накопителя информации

На представленной схеме имеется несколько электронных компонентов:

- микропроцессор электронного накопителя информации – реализует непрерывную обработку потоков данных между внутренними компонентами внешнего накопителя информации и корневым концентратором шины USB;
- блок электронных микросхем памяти – выступает компонентом компьютера, который обеспечивает непосредственное сохранение данных и является непосредственным носителем информации с самым низким энергопотреблением.

5.10. Периферийное оборудование компьютера

Внешние компоненты по отношению к системному блоку компьютера принято считать периферийным оборудованием, которое выполняет различные дополнительные функции, связанные с решением задач потенциальных пользователей.

В зависимости от сложности и типа решаемых задач пользователя периферийное оборудование дифференцируется на несколько основных классов и типов устройств:

- устройства ввода – ручной, автоматизированный и автоматический ввод данных;
- устройства вывода информации – вывод информации на разные носители;
- устройства передачи информации и сетевого взаимодействия – обмен данными.

Периферийное оборудование предназначено для расширения функций базовой конфигурации компьютера и зависит от сложности его архитектуры и назначения.

Установка дополнительного периферийного оборудования осуществляется посредством использования специальных разъемов и слотов несколькими способами:

- используются специальные разъемы (интерфейсы) на платах расширения, которые устанавливаются внутрь системного блока ЭВМ и подключаются к материнской плате через несколько слотов на ее плате печатного монтажа:
 - слот ISA (только для i386, i486, iPentium) – позволяет подключить плату сканера, содержащую внешний специальный разъем для подключения сканера;
 - слот PCI – позволяет подключить плату сканера или внешней памяти, которая содержит внешний специальный разъем для подключения сканера и памяти (установка платы расширения с разъемом SCSI позволяет установить внутренний и внешний накопитель информации на жестких магнитных дисках, а также внешний или внутренний лазерный накопитель на оптических дисках);
- используются стандартные внешние разъемы на материнской плате компьютера, которые специально вынесены на внешнюю панель системного блока:
 - параллельный порт LPT – позволяет подключать внешний принтер и внешний сканер, а также прочее имеющееся периферийное оборудование;
 - последовательный порт COM – позволяет подключать внешний модем для передачи информации по коммутируемому каналу данных (Dial Up);
 - универсальный последовательный порт USB (Universal Serial Bus) – позволяет подключать внешний принтер, внешний сканер, внешний модем;
- используется набор стандартных внутренних слотов на системной плате:
 - слот ISA – установка внутреннего модема, сетевого адаптера, звуковой платы;
 - слот PCI – установка внутреннего модема, сетевого адаптера, звуковой платы.

Определение

Периферийное оборудование – совокупность технических средств и компонентов, которые реализуют определенную функцию и задачу пользователя (ввод информации, вывод информации, передача данных и прочие), а также подключаются посредством использования внешнего порта и разъема или внутренней платы расширения с отдельным внешним интерфейсным разъемом специального назначения.

5.10.1. Устройства ввода информации

Устройства ввода информации позволяют реализовать непосредственную загрузку информации разного рода с носителей информации, которые подразделяются на:

- устройства ввода текстовой информации с носителей разного рода и назначения:
 - устройство чтения перфолент и перфокарт – носителем информации выступает перфолента (содержит отверстия в определенной последовательности согласно принятой закономерности) или перфокарта (имеет срез слева вверху, содержит матрицу из последовательности цифр и букв, которые пробиваются специальным устройством по желанию программиста);
 - клавиатура (проводная и беспроводная) – содержит несколько полей клавиш (основного ввода, цифрового ввода, управления курсором, функциональные, специальные – Caps Lock, Numb Lock, Scroll Lock, Pause, Break), позволяет управлять текстовым курсором в процессе ввода данных;
 - манипулятор типа мышь, тачпад, трекбол, джойстик – содержит две или три кнопки (выделение/активация объекта, вызов свойств объекта), механизм регистрации перемещения курсора манипулятора, переключатель канала для передачи по радиоканалу положения мыши и состояния клавиш;
- устройства ввода растровой графической информации с разных носителей:
 - цифровое перо – содержит сенсорную подставку с несколькими кнопками (некоторые из них дублируют кнопки манипулятора) и карандаш с накопчиком для нанесения простых геометрических фигур на плоскости;
 - сканер съема штрихкодов – реализует преобразование графического изображения штрихкода в соответствующую кодовую последовательность;
 - цифровой ручной сканер – корпус с одной или несколькими клавишами, которые обеспечивают активизацию оптической системы, позволяющей реализовать быстрое сканирование растрового графического изображения;
 - планшетный сканер – устройство прямоугольной формы, которое предназначено для сканирования графического изображения с плоской поверхности;
- устройства ввода векторной графической информации для ее обработки:
 - электронный планшет для черчения (дигитайзер) – содержит панель из нескольких кнопок, линейки для рисования чертежей на плоской поверхности;
- устройства для ввода речевой информации в определенном помещении:
 - аудио адаптер – электронный компонент с несколькими разъемами (MIC – внешний микрофон, Line In – линейный вход, Line Out – линейный выход, Midi/Game – внешний акустический синтезатор или манипулятор-джойстик), все разъемы интегрированы в основу материнской платы или выполнены в виде отдельного устройства подключаемого посредством слота ISA и PCI;
 - мультимедиа комплект для распознавания речи – внешние наушники и внешний микрофон, которые предназначены для работы с речевой информацией.
- устройства ввода видео-потока для его регистрации на носитель информации:
 - плата видео-захвата – обеспечивает оцифровку непрерывного видео-потока.

5.10.2. Устройства вывода информации

Наиболее распространенным среди пользователей устройством вывода информации является принтер, который позволяет реализовать формирование изображения на печатном носителе посредством использования контактного ударного, бесконтактного безударного способа, а также матричной, струйной или лазерной технологии печати.

Устройства вывода информации позволяют реализовать автоматизацию вывода текстовой, графической, речевой и прочей информации, подразделяются на:

- устройства вывода текстовой и графической информации – позволяют осуществить распечатку текста или графического изображения на плоскую поверхность посредством использования контактного ударного или бесконтактного безударного способа, а также матричной, воздушно-пузырьковой или лазерной технологии печати на разнородные носители информации (бумага):
 - матричный или игольчатый принтер – красящая лента пробивается иглками;
 - струйный принтер – выстреливаются жидкие ферромагнитные чернила;
 - лазерный принтер – лазером формируется теньевая маска и подается тонер;
- устройства вывода аудио информации – позволяют выводить аудио-данные:
 - аудио-адаптер – реализует формирование стереофонического многоканального звукового потока из файла на машинном носителе информации компьютера посредством использования наушников или акустической системы;
 - наушник(и) – реализуют воспроизведение звукового потока низкой громкости;
 - стереофонические динамики – реализуют воспроизведение стерео сигнала;
 - акустическая система 2.1 – два стереофонических динамика и сабвуфер;
 - акустическая система 4.1 – четыре стереофонических динамика и сабвуфер;
 - акустическая система 6.1 – четыре динамика, два сателлита, и сабвуфер;
 - акустическая система 8.1 – четыре динамика, четыре сателлита и сабвуфер;
- устройства вывода видео информации – позволяют выводить видео-данные:
 - алфавитно-цифровое табло – позволяет отображать подсвеченные и мерцающие транспаранты, которые содержат буквы, цифры, слова и словосочетания;
 - проекционная панель – реализует возможность проецирования статического или динамического графического изображения посредством использования кинопроектора или прочего устройства отображения информации;
 - электронно-лучевой монитор – проекционная система основана на электронной вакуумной трубке, которая содержит электронную пушку, вертикальную и горизонтальную отклоняющую системы магнитным полем;
 - жидко-кристаллический монитор – проекционная система основана на пассивной или активной жидкокристаллической матрице с вертикальной и горизонтальной дифракционной решеткой и источником матового света;
 - плазменная панель – проекционная система нового поколения, основанная на свечении галогена за счет электрических разрядов, которые подаются на специальную чувствительную матрицу в замкнутом объеме.

5.10.3. Устройства сетевого взаимодействия и передачи данных

Устройства сетевого взаимодействия и передачи данных обеспечивают реализацию обмена информацией разного рода и содержания посредством использования канала передачи данных определенного типа, подразделяются на несколько видов:

1. Сетевое оборудование, которое используется для передачи данных на стороне клиента:
 - окончное оборудование передачи данных по коммутируемому кабельному каналу передачи аудио, видео информации и данных пользователя:
 - телефон и факс – позволяет оперативно передавать голосовые и факсимильные сообщения посредством использования канала передачи данных;
 - Dial-Up модем и факс-модем – позволяет передавать данные (скорость от 300бит/сек до 57600 бит/сек) и факсимильные сообщения (скорость от 1200 бит/сек до 14400 бит/сек) посредством специальных протоколов;
 - xDSL модем – позволяет очень быстро передавать данные по асимметричному кабельному каналу связи (скорость от 300 бит до 500Мбит/сек);
 - окончное оборудование передачи данных по радиочастотному каналу связи:
 - Bluetooth модем – позволяет передавать данные по радиоканалу связи;
 - Wireless модем – реализует высокоскоростную передачу данных среднего объема посредством радиочастотного канала связи с высокой скоростью;
 - окончное оборудование передачи данных по кабельному каналу связи:
 - сетевой адаптер стандарта Token Ring (коаксиальный кабель, витая пара);
 - сетевой адаптер стандарта Ethernet (коаксиальный кабель, витая пара поколения 1, 2, 3, 4, 5, которая также предусматривает экранирование).
2. Коммуникационное оборудование для поддержки функционирования ЛВС:
 - сетевые концентраторы и усилители сигнала при его непосредственной передаче:
 - сетевой концентратор стандарта Token Ring объединяет взаимодействие нескольких сегментов сети и ПЭВМ на автоматизированных рабочих местах;
 - сетевой концентратор стандарта Ethernet позволяет реализовать трансляцию сигналов между определенным количеством коммуникационных портов, к которым подключены автоматизированные рабочие места пользователей;
 - сетевой усилитель (Switch) позволяет реализовать увеличение длины сегментов сети за счет усиления и ретрансляции сетевых пакетов между одним входным и одним коммуникационным портом или входным портом и несколькими рабочими станциями, а также прочим сетевым оборудованием;
 - сетевой адаптер – реализует сетевое взаимодействие между несколькими ПЭВМ, за которыми работают ряд пользователей различных категорий;
 - мост – объединяет вычислительную сеть на основе различных топологий, использующих однотипные или различные протоколы взаимодействия и сетевую ОС, которая обеспечивает передачу информации между сетями;
 - шлюз – обеспечивает защищенный доступ к ресурсам другой сети, причем допускается существенное отличие протоколов взаимодействия.

3. Коммуникационное оборудование, которое используется на стороне сервера:

- сетевой экран – обеспечивает мониторинг сетевой активности на уровне слотов и сокетов, а затем блокирует и индицирует попытки несанкционированного подключения к внутренней локальной сети из внешнего сегмента локальной или глобальной вычислительной сети с целью доступа к ее разным ресурсам;
- маршрутизатор – обеспечивает конфигурирование направлений и последовательности передачи пакетов от одного узла к другому в зависимости от их типа и адреса, а также управляет формированием пакетов данных и логических каналов для передачи сообщений, выполняет балансировку сетевой нагрузки;
- модемный пул – содержит совокупность портов с серверными модемами, которые обеспечивают подключение модемов клиента несколькими способами:
 - модемный пул Dual-Up модемов – подключение модема по коммутируемому каналу доступа с полной занятостью линии связи (телефонной);
 - модемный пул xDSL модемов – подключение модема по коммутируемому каналу доступа с возможностью совершать исходящие вызовы (телефонный кабель) или при использовании кабельного канала передачи информации.

Организация локальной вычислительной сети предполагает ряд мероприятий:

- анализ вертикально или горизонтально интегрированной организационной структуры предприятия: выделение подразделений, отделов, секторов, рабочих мест, а также обеспечение закупки и внедрения средств автоматизации;
- выделение функций структурных подразделений организации и задач пользователей согласно специфике информационных процессов в сфере использования;
- разработка структуры всех информационных процессов: по подразделениям, по документообороту, по товарам, работам и услугам, по себестоимости;
- разработка структуры информационной системы, которая включает организационное, технико-технологическое, аппаратное и программное обеспечение;
- закупка аппаратного и программного обеспечения, которое позволяет реализовать автоматизацию всех информационных процессов во всех подразделениях;
- проектирование структуры вычислительной сети с учетом расположения автоматизированных рабочих мест с ЭВМ и на основе стандартной топологии;
- настройка коммуникационных серверов, серверов БД и информационных хранилищ, Вэб серверов, серверов с учетными записями и прочими данными;
- настройка коммуникационного оборудования: концентраторов, маршрутизаторов, сетевых экранов и шлюзов, а также прочего сетевого оборудования;
- установка коммуникационного оборудования, монтаж и ввод в эксплуатацию локальной вычислительной сети или определенного сегмента сети;
- сопровождение, техническое обслуживание, модернизация локальной вычислительной сети, а также поиск нового сетевого оборудования для нее;
- повышение квалификации, обучение, тренировка технического персонала.

Выделяют несколько базовых топологий организации вычислительных сетей:

- линейный принцип построения (шинная организация) – включает последовательность подключенных друг к другу ЭВМ различных пользователей, а в случае выхода из строя одного из узлов передача информации невозможна;
- разветвленный и многоуровневый принцип построения вычислительных сетей – предполагает использование сложных топологий при их организации:
 - топология типа «кольцо» – выступает гибридом шинной топологии и представляет собой замкнутую в кольцо последовательность узлов сети;
 - топология типа «звезда» – выступает гибридом шинной топологии и представляет собой несколько последовательно соединенных между собой узлов сети, которые подключены к единому – центральному узлу сети;
 - топология типа «кольцо-шина» – образована за счет замкнутой последовательности узлов, каждый из которых имеет ответвление в виде шины.

Многоуровневый принцип построения топологии сетей достигается посредством реализации гибридной архитектуры, которая содержит в своей основе комбинацию существующих топологий организации вычислительных сетей: линейная (шина), звезда, кольцо, разветвленная (гибридная), разветвленная многоуровневая.

Шинная топология предполагает создание одноранговой вычислительной сети с единым каналом передачи данных, а подключение каждого компьютера для работы пользователя осуществляется посредством использования сетевого адаптера.

Недостатком использования шинной топологии является невозможность передачи данных между ЭВМ в случае дисфункции определенного сегмента сети.

Организация сети на основе шинной топологии представлена на рис. 5.24.

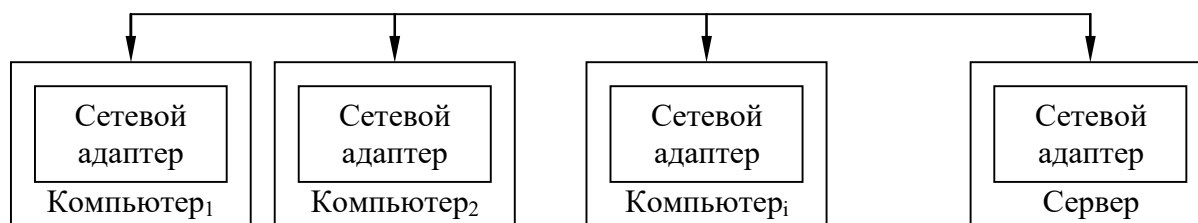


Рисунок 5.24. Линейная топология локальной вычислительной сети

Топология типа «звезда» при организации сети предполагает использование концентратора, который имеет несколько портов для подключения вилки сетевого кабеля непосредственно соединяющего сетевой адаптер установленный внутри ПЭВМ. Концентратор сети устанавливается в специально оборудованном помещении и может функционировать независимо в автономном режиме или подключаться к одному из ПЭВМ, который при этом выступает сервером (транспортным, файловым).

Недостатком использования топологии типа «звезда» является высокая вероятность выхода из строя единственного концентратора сети, что потенциально может обусловить невозможность передачи данных между всеми узлами одновременно.

При этом возникает необходимость резервирования концентратора (сервера).

Организация сети на основе топологии типа «звезда» отражена на рис. 5.25.

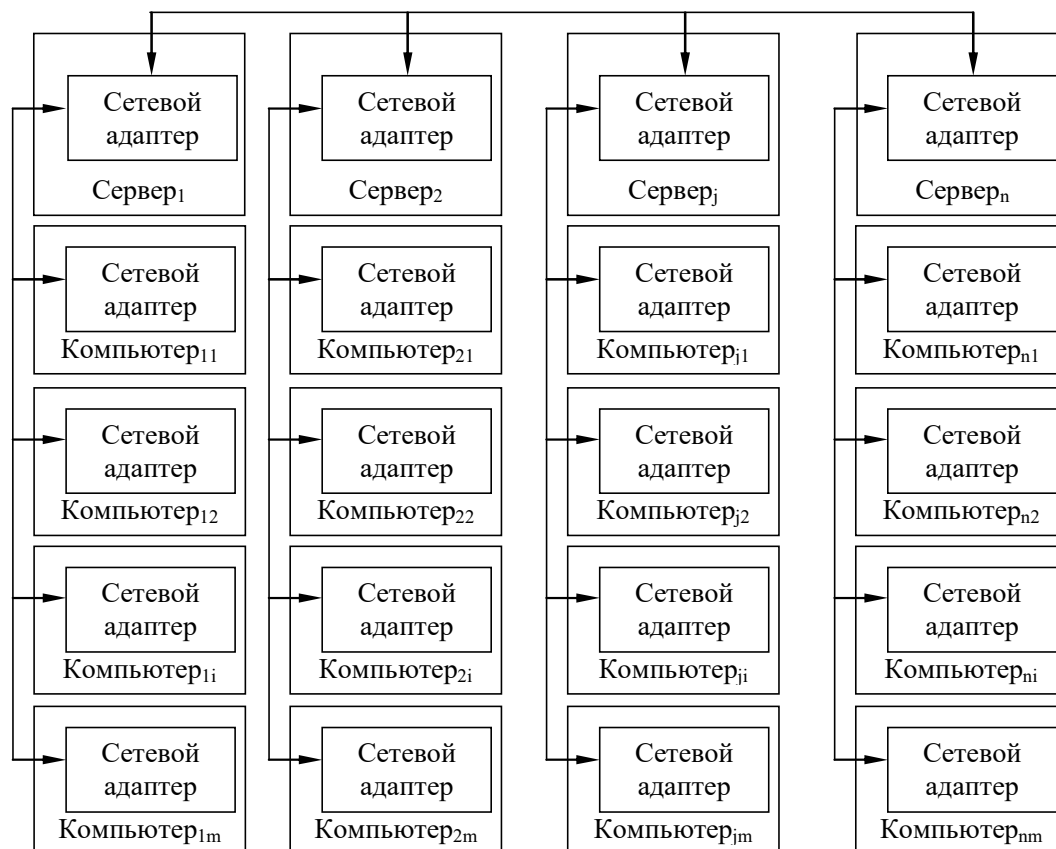


Рисунок 5.25. Линейная многоуровневая топология локальной вычислительной сети

В представленной схеме (рис. 5.25) присутствуют несколько серверов, к каждому из которых подключено несколько рабочих станций (компьютеров), при этом возможно использование концентратора между всеми серверами, что повышает вероятность отказа сети, поэтому он не используется из целей отказоустойчивости.

Каждый из представленных серверов реализует один сегмент вычислительной сети и обеспечивает подключение всех компьютеров в пределах него, но при этом представляется возможным несколько вариантов авторизации каждого компьютера:

- независимая авторизация каждого ЭВМ, хранение пароля на компьютерах;
- авторизация каждого компьютера осуществляется в пределах сегмента сети, к которому он подключен, а хранение учетных записей реализуется на сервере сегмента;
- авторизация каждого компьютера осуществляется независимо по всей сети, а хранение учетных записей осуществляется только на одном из серверов.

При использовании топологии типа «звезда» допустимо каскадирование, которое предполагает подключение последнего порта одного концентратора к первому порту другого концентратора сети и позволяет создать топологию «звезда-звезда».

Топология вычислительной сети типа «кольцо» имеет ряд ключевых преимуществ:

- передача информации в случае повреждения и дисфункции части канала передачи данных вследствие выхода из строя аппаратуры передачи данных;
- балансировка сетевой нагрузки посредством использования различных сетевых операционных систем (Linux, Unix, Windows 2000 Server/Advanced Server/XP).

Сеть при использовании топологии типа «кольцо» представлена на рис. 5.26.

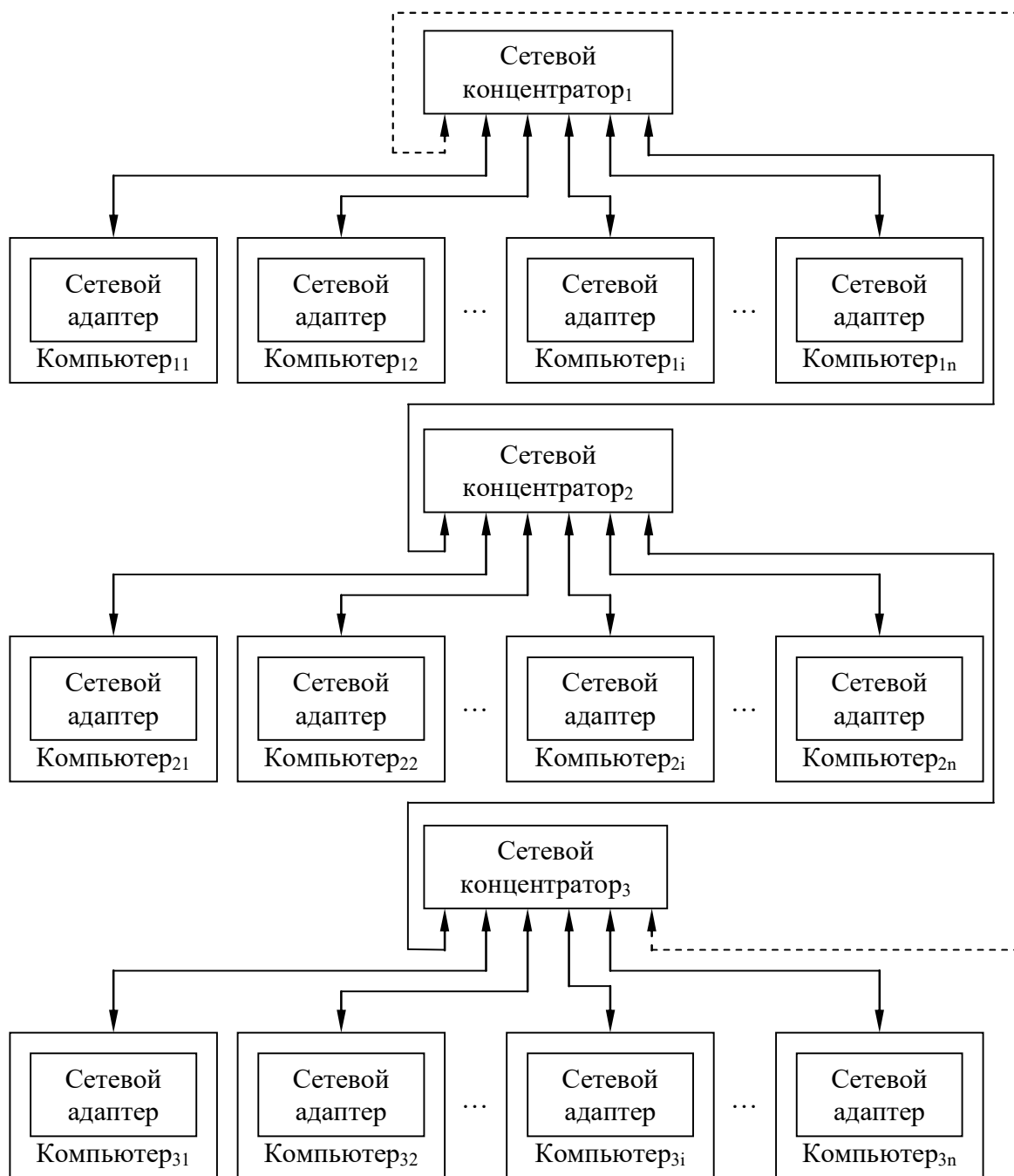


Рисунок 5.26. Топология локальной вычислительной сети типа «звезда»

Для вертикально интегрированных организационных структур, которые содержат несколько отделов и подразделений топология сети существенно усложняется, в частности возможно создание гибридной топологии типа «кольцо-звезда»:

- на уровне каждого отдела реализуется топология «звезда» – устанавливается концентратор сети и несколько ЭВМ или сетевой сервер и несколько ПЭВМ;
- реализуется топология типа «кольцо» – создается посредством соединения сегментов сети между собой за счет каскадирования всех концентраторов сети, которые расположены в разных подразделениях и отделах предприятия.

С учетом технических характеристик трех концентраторов возможно соединение первого порта первого концентратора с последним портом третьего концентратора.

Организация вычислительной сети из трех серверов и трех концентраторов посредством гибридной топологии типа «кольцо-звезда» представлена на рис. 5.27.

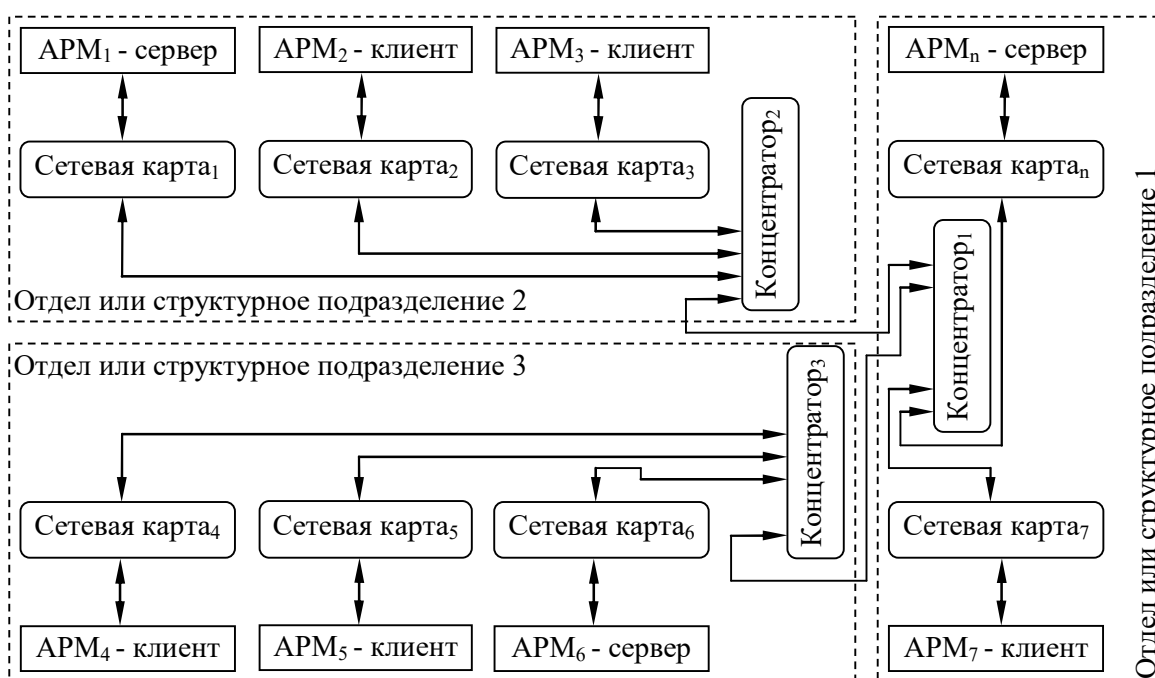


Рисунок 5.27. Топология локальной вычислительной сети типа «кольцо-звезда»

Представленные три концентратора в созданной вычислительной сети обслуживают все компьютеры каждого из трех структурных подразделений организации:

- в первом структурном подразделении находится один сервер и один компьютер-клиент, которые подключены в концентратору вычислительной сети;
- во втором подразделении находится один сервер и два компьютера-клиента, которые подключены ко второму концентратору вычислительной сети;
- в третьем подразделении организации расположены один сервер и два компьютера-клиента, которые подключены к третьему концентратору сети.

Созданная вычислительная сеть обладает относительно высокой надежностью, поскольку выход из строя одного из концентраторов сети обуславливает отключение только одного из сегментов, что позволяет двум оставшимся сегментам сети работать в регламентированном порядке независимо от третьего сегмента сети.

Возможны следующие случаи эксплуатации созданной вычислительной сети:

- первый отдел организации содержит автоматизированные рабочие места системного администратора (сервер с информационным хранилищем) и помощника системного администратора (компьютер для реализации мониторинга состояния вычислительной сети и управления разделяемыми ресурсами);
- второй отдел организации имеет автоматизированные рабочие места администратора второго отдела (файловый сервер с БД отдела) и нескольких сотрудников (компьютеры), которые занимаются получением и отправкой данных;
- третий отдел организации включает несколько автоматизированных рабочих мест администратора отдела (файловый сервер с БД отдела) и нескольких сотрудников (компьютеры), которые занимаются систематизацией и обработкой данных.

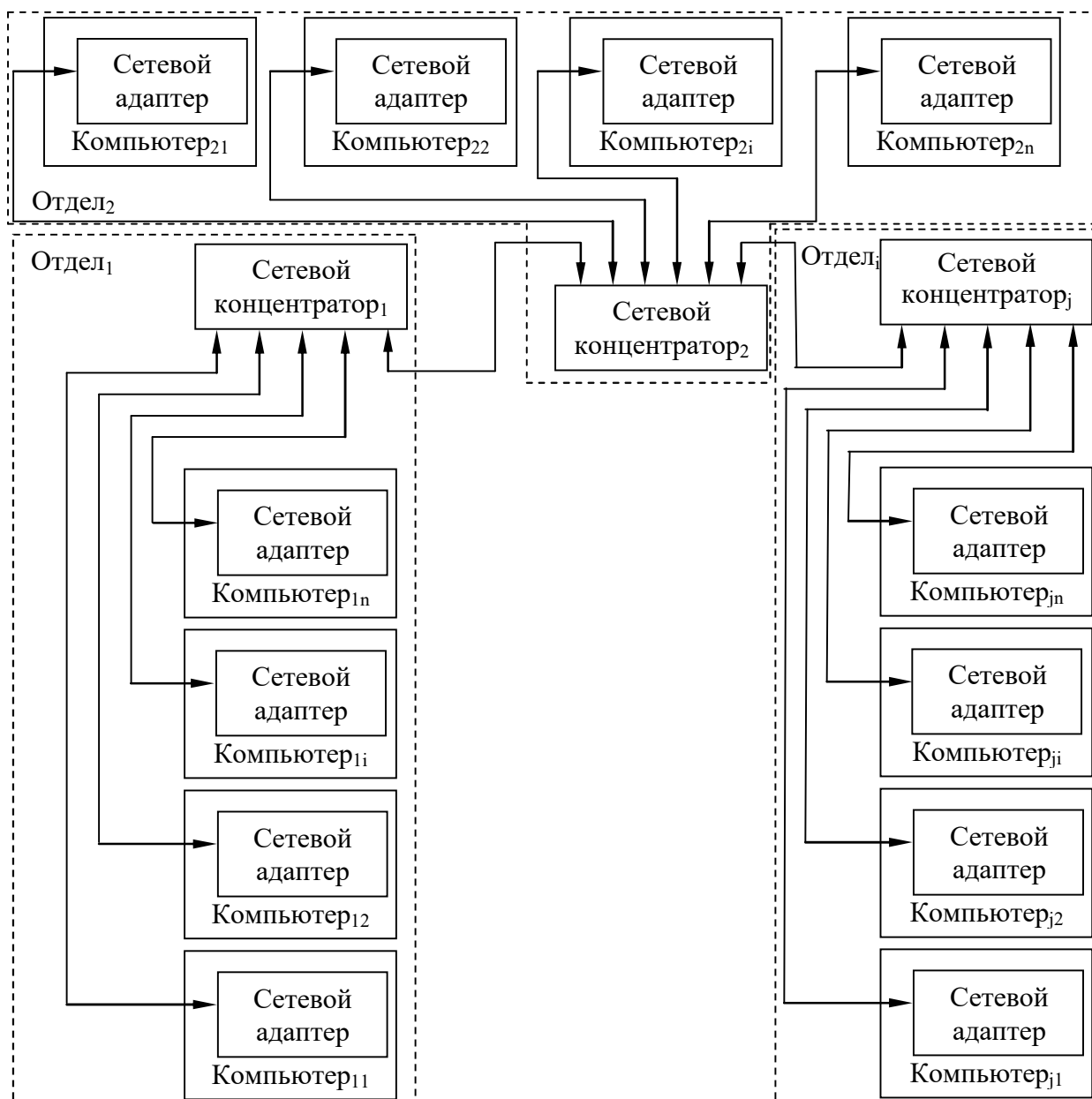


Рисунок 5.28. Топология локальной вычислительной сети типа «кольцо»

Топология организации ЛВС типа «кольцо-звезда» (рис. 5.28) наиболее приемлема с точки зрения надежности, поскольку в случае дисфункции одного из сетевых адаптеров или концентраторов возможна передача данных между оставшимися узлами сети:

- ЭВМ с сетевыми адаптерами, которые имеют возможность доступа к ресурсам;
- концентратор обслуживает сегмент сети (отдел) и реализует обмен данными.

В сложных предметных областях (медицина, наука, техника, экономика, политика, образование) с обширным набором функций и задач существуют дополнительные принципы организации сложных систем с использованием распределенной архитектурой:

- интеграция ресурсов сети (аппаратное и программное обеспечение, БД);
- стандартизация и специализация номенклатуры сетевого оборудования;
- повышение качества образовательной и организационной информации за счет учета ее свойств (достоверность, оперативность, актуальность, точность);
- создание единого информационного пространства ряда систем на основе ИКТ.

В частности, информатизация информационно-образовательных сред образовательных учреждений и создание систем автоматизированного (дистанционного) обучения регулируется на основе требований, выработанных государственными и международными органами, регламентирующими развитие системы образования.

Разработка инфраструктуры информационно-образовательной среды рассматривается на нескольких уровнях ее существования в пределах системы образования:

- локальный уровень (отдельное образовательное учреждение) – внедрение различных средств автоматизации в информационно-образовательную среду;
- региональный уровень (региональное представительство) – реализуется информатизация региональных представительств образовательных учреждений;
- международный уровень (образовательный консорциум) – единая информационно-образовательная сеть между образовательными учреждениями, оперирующими независимо или на основе дилерских и брокерских отношений.

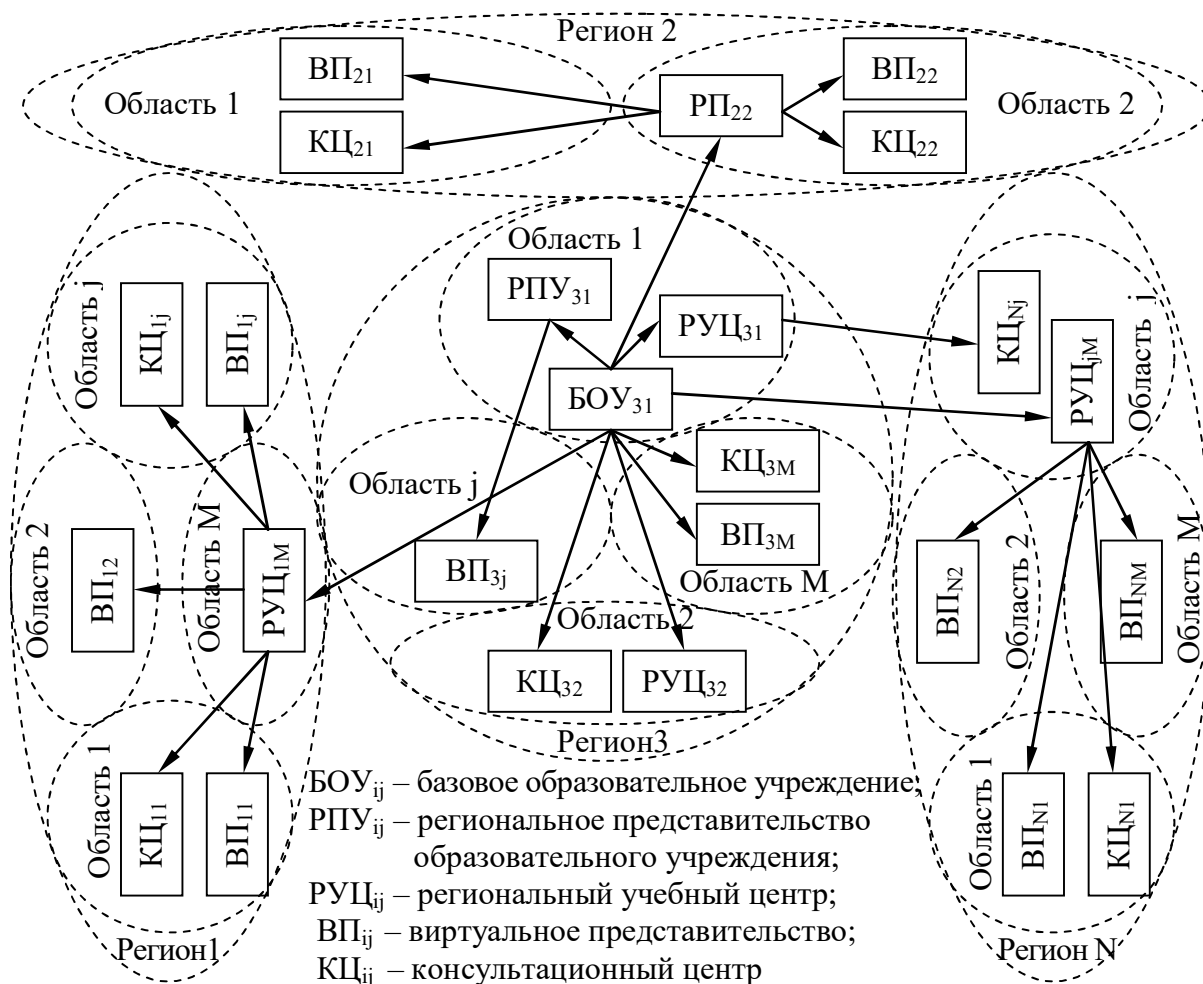
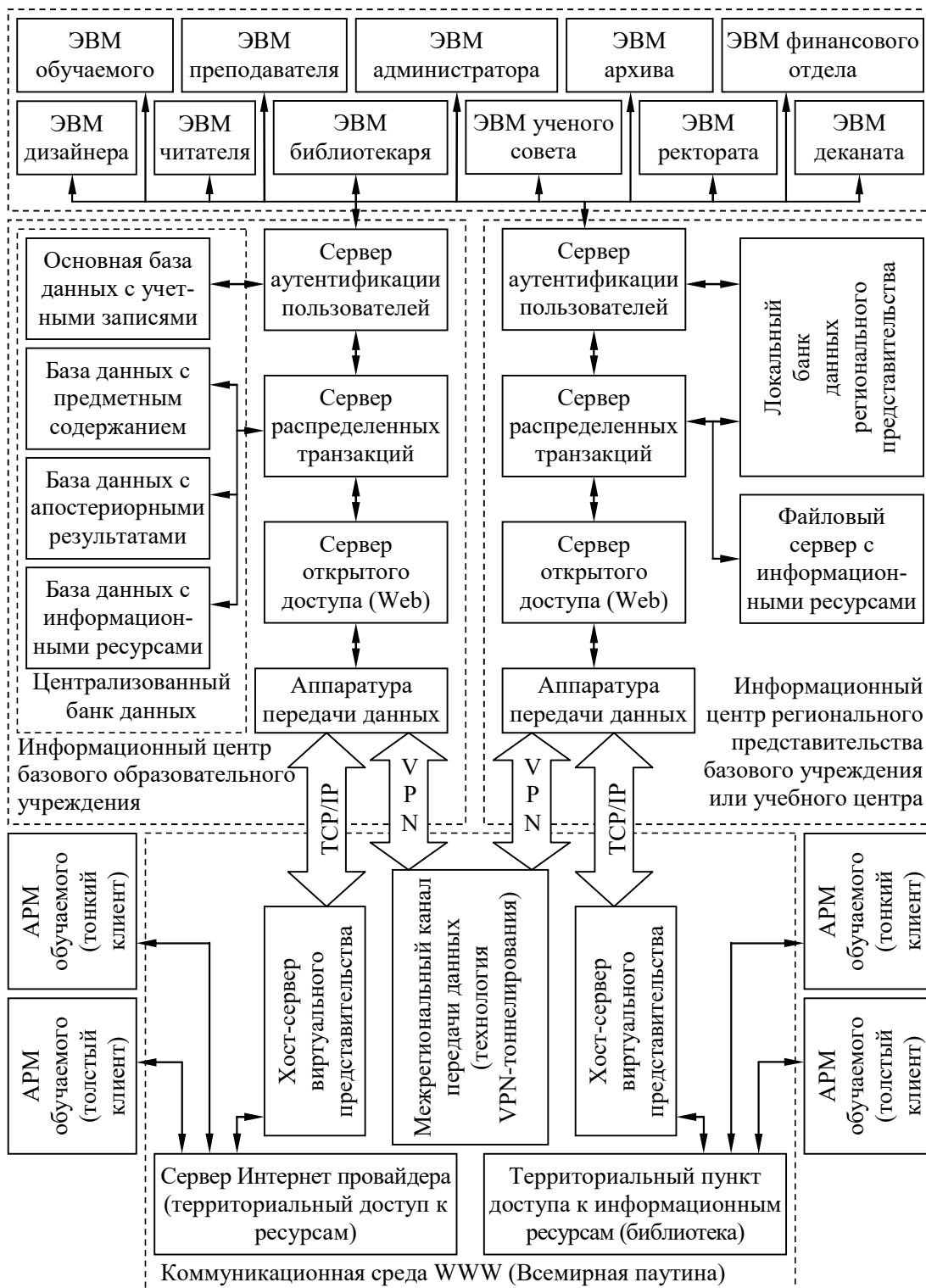


Рисунок 5.29. Топология организации распределенной информационно-образовательной среды автоматизированного (дистанционного) обучения

На рис. 5.29 базовое образовательное учреждение имеет разветвленную сеть региональных и виртуальных представительств в крупных научных и образовательных центрах, а также несколько учебных центров и консультационных центров находящихся в территориально распределенных удаленных регионах и областях.

При анализе обмена разнородной информацией между субъектами расположенными на локальных или удаленных ЭВМ и средствами обучения в пределах базового образовательного учреждения, а также производной от него сети учебных и консультационных центров предлагается типовая схема взаимодействия на рис. 5.30.



VPN – Virtual Private Network (виртуальная частная сеть);
 TCP/IP – Transmission Control Protocol / Internet Protocol (Протокол передачи данных);
 WWW – World Wide Web (всемирная паутина)

Рисунок 5.30. Типовая схема взаимодействия информационного центра образовательного учреждения и автоматизированных рабочих мест

5.11. Устройства печати информации на разные носители

Определение

Устройствами печати информации представляют собой технические средства (периферийные компоненты), которые обеспечивают символьный, строчный и страничный вывод информации посредством использования контактного ударного или безударного, бесконтактного способов формирования изображения на основе игольчатой (матричной), струйной (воздушно-пузырьковой) или лазерной технологий печати.

Определение

Символьный способ формирования изображения при печати предполагает посимвольный вывод информации посредством использования матрицы с оттисками различных букв, цифр и символов на основе одной из существующих технологий печати (в основном применяется контактная ударная технология печати).

Определение

(По)строчный способ формирования изображения при печати предполагает вывод информации посредством использования контактного ударного и безударного способа и матричной, струйной или лазерной технологии печати с использованием:

- игольчатой матрицы определенного размера – обеспечивает формирование изображения за несколько проходов игольчатой матрицы относительно строки печатного носителя (лист бумаги) посредством ударов иглолок по красящей ленте
- струйной печатной головки – реализует формирование изображения за несколько проходов печатной головки, которая обеспечивает выстреливание жидких или гелеобразных капель одного или разного размера (воздушно-пузырьковая технология – Bubble Jet Technology) одного или нескольких цветов;
- лазерной проекционно-магнитной головки – обеспечивает формирование черно-белого или цветного изображения в рамках нескольких технологических заделов печати: намагничивание поверхности магнитного барабана, снятие состояния намагниченности посредством лазерного луча, реализующего сток электростатического потенциала, нанесение микродисперсных частиц тонера (порошок обладающий ферромагнитными свойствами) на поверхность печатного носителя, нагрев микродисперсных частиц тонера и смена агрегатного состояния с твердого до жидкого, пропитывание волокон листа жидким тонером и его последующее отверждение, а затем прокатывание печатного носителя (бумага, картон, прозрачная пленка) через прижимной валик для устранения неоднородностей в полученном изображении.

Определение

(По)страничный способ формирования изображения при печати предполагает использование клеше со сформированным оттиском каждой страницы, нанесение чернил на поверхность клеше, прижатие клеше к поверхности печатного носителя, пропитывание волокон листа чернилами с поверхности клеше, выдерживание печатного носителя для взаимодействия с воздушной смесью с целью отверждения чернил.

5.11.1. Матричные принтеры

Матричные принтеры являются относительно шумными устройствами печати.

Определение

Матричным принтером называют устройство вывода информации, которое обеспечивает формирование заданного изображения посредством контактного ударного способа с использованием игольчатой матрицы определенного размера или матрицы из клеше (оттисков) разных символов ударяющих по красящей ленте (рис. 5.31).

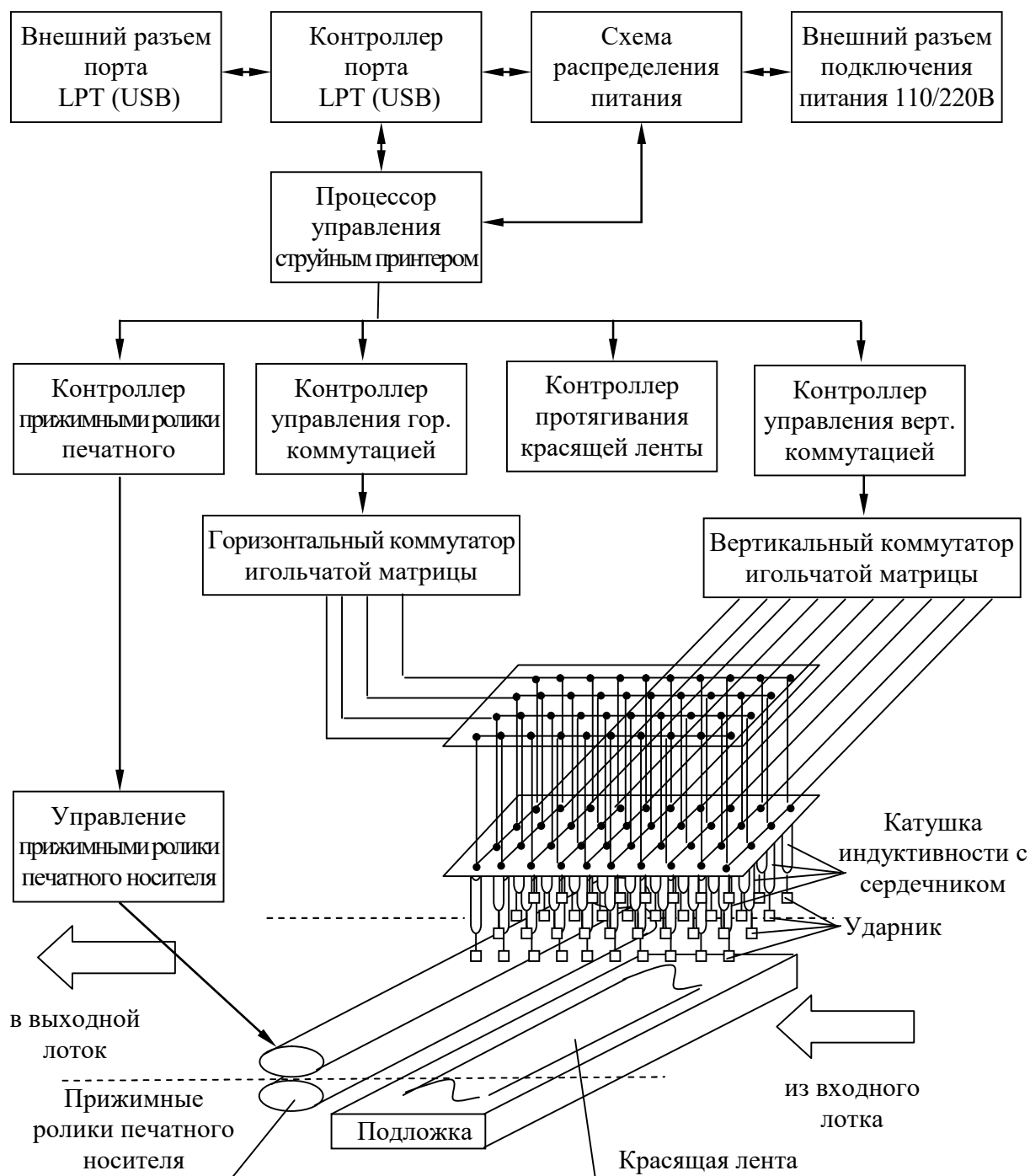


Рисунок 5.31. Структурная схема матричного принтера

Котроллер протягивания красящей ленты обеспечивает перемещение ленты, контроллер управления прижимными роликами смещает печатный носитель (бумагу), а контроллер управления коммутацией реализует удары игольчатой матрицы по носителю.

5.11.2. Струйные принтеры

Струйные принтеры выступают менее шумными устройствами печати и подразделяются на несколько основных видов в зависимости от конструкции картриджа: принтеры со съемными картриджами (печатными головками), которые конструктивно содержат в своей основе резервуар для чернил (Canon, HP); принтеры со съемными печатными головками в своей основе, которые конструктивно предполагают замену одной, двух, трех и более чернильниц (Canon, HP); принтеры со стационарными печатными головками (не имеют картриджа), но предусматривают замену съемных чернильниц (Lexmark, Epson).

Определение

Струйным принтером (рис. 5.32) называют устройство вывода информации, которое обеспечивает формирование изображения посредством бесконтактного безударного способа печати с использованием струйной воздушно-пузырьковой технологии на печатном носителе определенного вида (бумага, картон, фольга, пленка, металл).

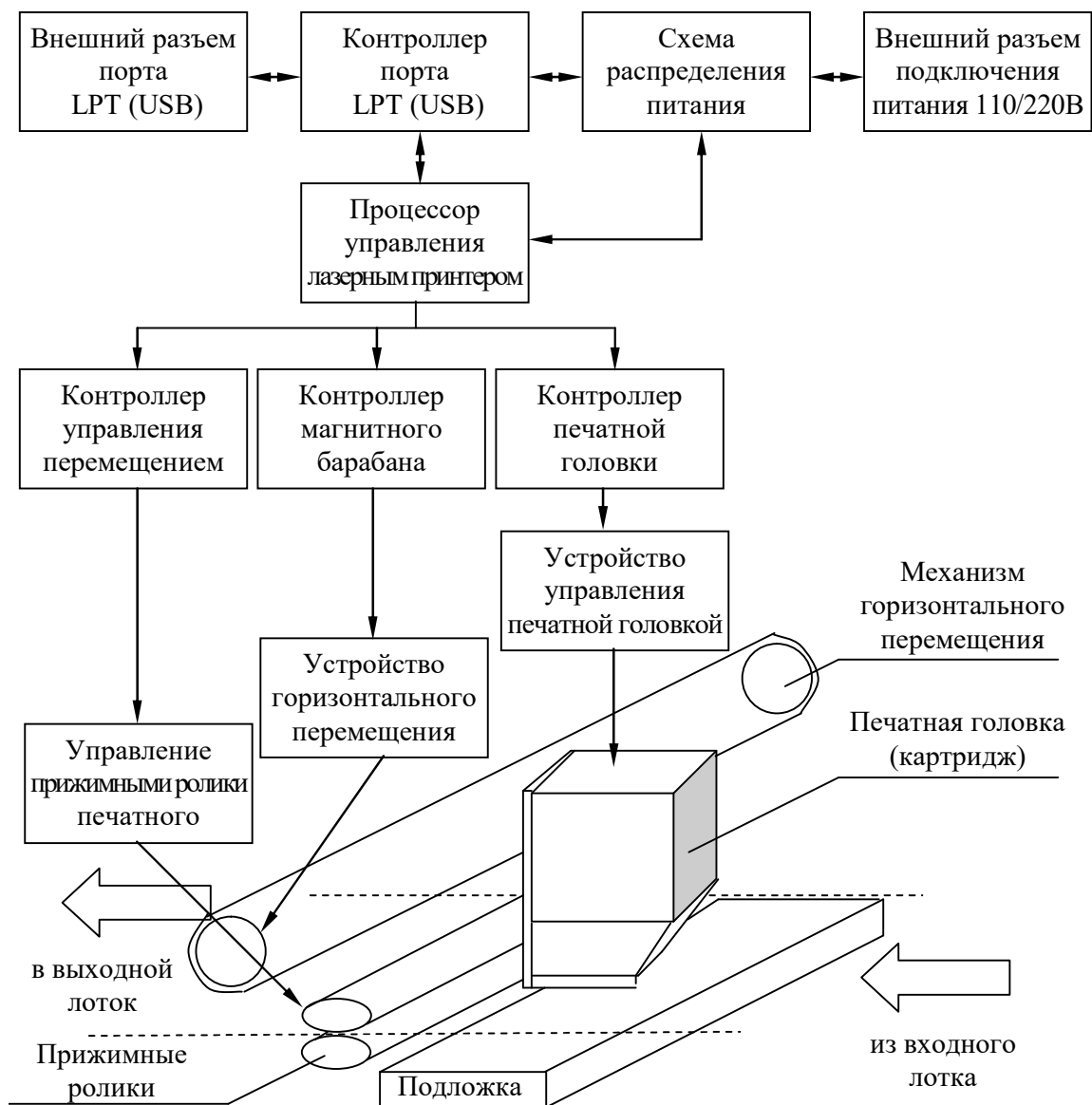


Рисунок 5.32. Структурно-функциональная схема струйного принтера

Структура воздушно-пузырьковой печатной головки струйного принтера (рис. 5.33).

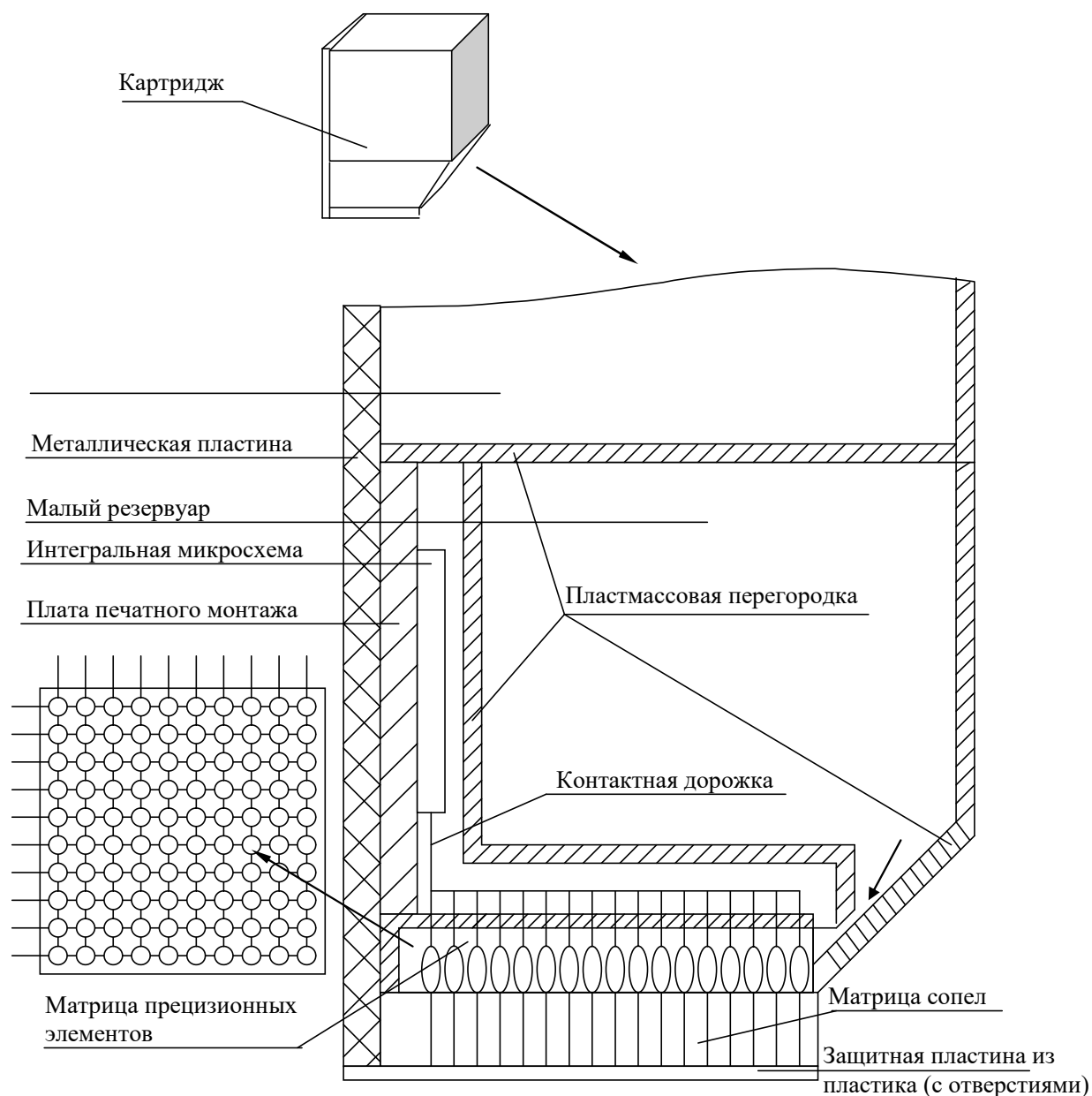


Рисунок 5.33. Структура воздушно-пузырьковой печатной головки струйного принтера

Определение

Прецизионный элемент картриджа – устройство создающее избыточное давление в объеме картриджа принтера при подаче на его клеммы электрического тока.

Прецизионные элементы картриджа располагаются несколькими способами: горизонтальное расположение – все прецизионные элементы располагаются в виде единой линии друг за другом на одинаковом или разном удалении; матричное расположение – прецизионные элементы располагаются в несколько линий (рядов) на одинаковом или разном удалении друг относительно друга.

Микросхема обеспечивает коммутацию прецизионных элементов в зависимости от геометрического расположения места точек формируемого заданное изображения.

Матрица прецизионных элементов непосредственно обеспечивает выстреливание капель чернил, которые поступают из резервуара печатной головки картриджа.

5.11.3. Лазерные принтеры

Лазерные принтеры (рис. 5.34) являются относительно бесшумными печатающими устройствами, которые подразделяются на две основные группы в зависимости от особенностей выполнения конструкции печатающей головки (картриджа):

- принтеры с неразборной одноразовой печатающей головкой (картриджем) (HP);
- принтеры с печатающей головкой, которая включает неразборный картридж со специальным отверстием для одно-двукратной заправки тонером (Xerox);
- принтеры с разборным картриджем, который состоит из двух основных частей:
 - тонер-картридж – содержит емкость с микродисперсными частицами красящего порошка с ферромагнитными свойствами, который при нагреве переходит в жидкое состояние, а при последующем охлаждении отвердевает;
 - магнитный барабан-картридж – металлический цилиндр, который обладает ферромагнитными свойствами и способен притягивать частицы тонера;
 - лазерная оптическая система – формирует теньевую маску на поверхности магнитного барабана: лазерным лучом снимается поверхностный заряд;
 - печь – подогрев намагниченного тонера на поверхности печатного носителя.

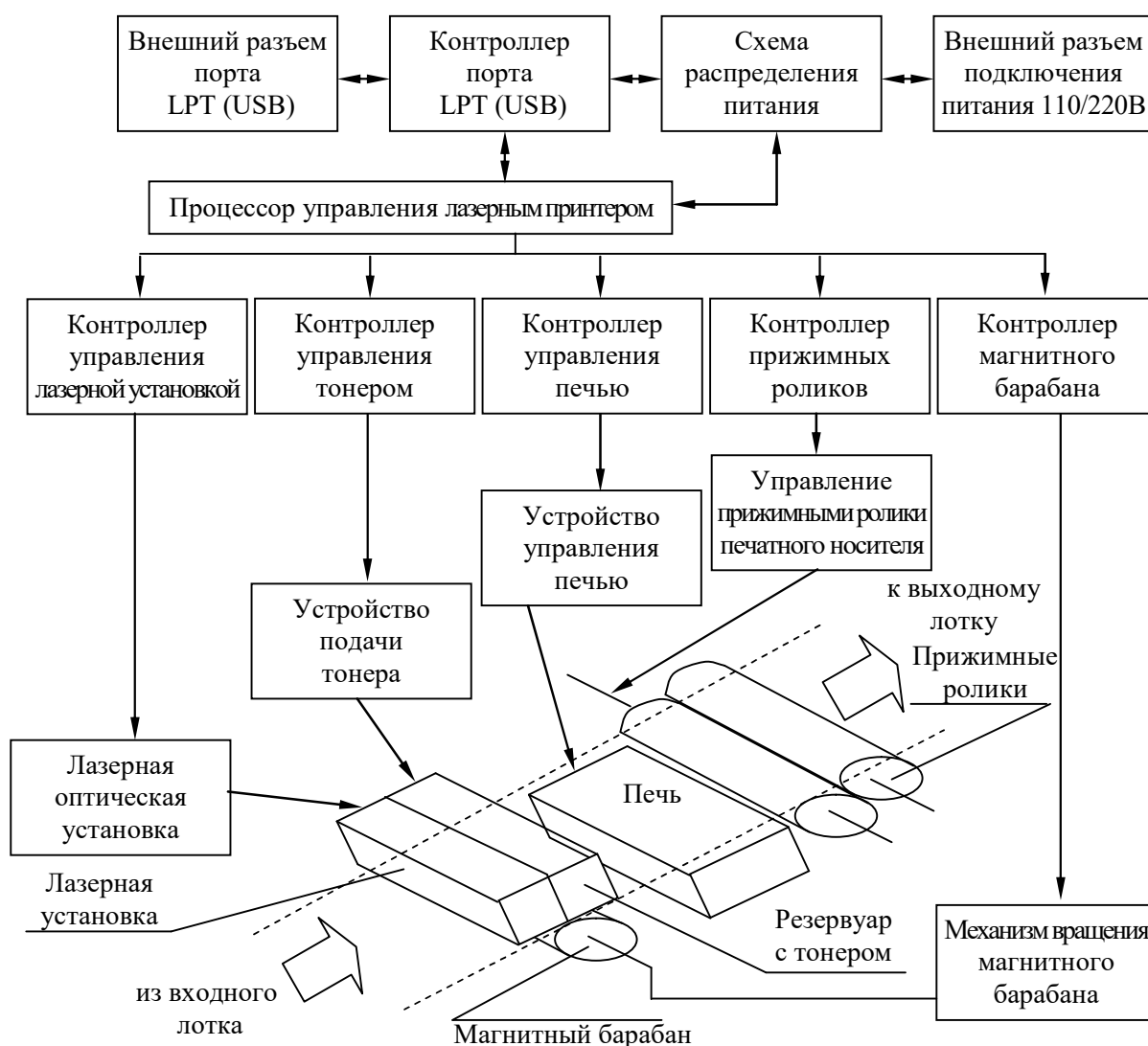


Рисунок 5.34. Структурно-функциональная схема лазерного принтера

Определение

Лазерным принтером называют устройство вывода информации, которое обеспечивает формирование изображения посредством бесконтактного безударного способа с использованием лазерной технологии печати на носителе информации определенного вида (бумага, картон, дерево, пленка, пластмассовая пластина).

Разные типы принтеров характеризуются определенным набором внутренних компонентов, которые обеспечивают поддержку формирования изображения разным способом, поэтому существует ряд основных классификационных признаков:

1. По уровню генерируемого шума при работе печатающего устройства (принтера):
 - малошумные – принтер для офсетной печати, лазерный и струйный принтеры;
 - среднего уровня шума – струйные принтеры с прецизионными элементами;
 - шумные – матричные принтеры за счет использования игольчатой матрицы.
2. По уровню энергопотребления при работе узлов и агрегатов принтера:
 - не потребляют электрическую энергию – ручные принтеры для офсетной печати с использованием наборной матрицы из букв, цифр и символов;
 - маломощные (20-500Вт) – струйные принтеры с прецизионным элементом;
 - средней мощности (500-1000Вт) – матричные (игольчатые) принтеры;
 - мощные (1000-3000Вт) – лазерные принтеры для печати на разных носителях.
3. По питающему напряжению и особенностям использования устройства:
 - автономные принтеры – питание осуществляется посредством использования аккумуляторов или батареи элементов питания с низким напряжением и током;
 - неавтономные принтеры – питание реализуется посредством использования силовой сети 110/220В и схемы распределения питания между элементами.

В результате анализа технологических особенностей реализации различных типов принтеров следует отметить их основные особенности и параметры (табл. 5.2).

Таблица 5.2

Сравнительная характеристика принтеров и используемых технологий печати

№ п.п.	Наименование печатающего устройства	Наименование технологии печати	Количество цветов	Разрешающая способность, точек на дюйм	Особенности конструкции
1.	Принтер для офсетной печати	Контактная безударная	1	10-100	Клеше с набором символов и лента с нанесенной краской
2.	Матричный принтер	Контактная ударная	1 или 2	100-200	Игольчатая матрица
3.	Струйный принтер	Бесконтактная безударная	16-16536	200-2400	Чернильный картридж
4.	Лазерный принтер	Бесконтактная безударная	1-256 (до 16536)	600-2400 (и выше)	Лазерная система

В процессе печати при помощи разных принтеров используются различные технологии формирования изображения, но в результате достигается разный уровень качества:

- лазерный принтер – позволяет получить однородное графическое изображение;
- струйный принтер – позволяет получить неоднородное четкое изображение;
- матричный (струйный) принтер – реализует возможность получения графического изображения, которое включает набор элементарных точек одинакового размера, расположенных на одинаковом или различном удалении друг от друга;
- принтер для офсетной печати – реализует возможность получения существенно неоднородного изображения, которое формируется посредством пропитывания волокон печатного носителя (бумаги) красящим веществом определенного цвета.

При работе матричного, струйного и лазерного принтера соответственно осуществляется формирование символического и графического изображения посредством использования разных способов (рис. 5.35): контактный ударный с помощью игольчатой матрицы (а), бесконтактный безударный посредством воздушно-пузырькового картриджа (б), бесконтактный безударный за счет лазерной оптической системы (в).

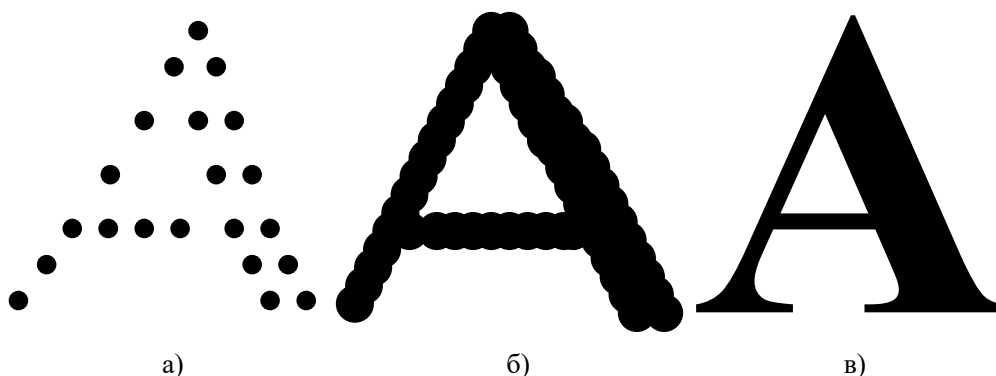


Рисунок 5.35. Особенности формирования изображения на матричном, струйном и лазерном принтере

Согласно имеющемуся рисунку можно упорядочить технологии печати, которые используются в разных принтерах по уровню качества формируемого изображения:

- матричный принтер – вследствие самой низкой разрешающей способности обладает самым низким качеством формируемого изображения с использованием одного или двух цветов: количество цветов определяется количеством полос на красящей ленте, которые обеспечивают формирование изображения;
- струйный принтер – позволяет получить более высокое качество изображения;
- лазерный принтер – реализует возможность получения четкого однородного графического изображения с использованием одного или множества цветов.

Определение

Разрешающая способность – количество точек по вертикали и горизонтали, которые потенциально используются для формирования графического изображения.

Определение

Глубина цвета – потенциальное количество цветов, которые способна принимать элементарная точка при формировании графического изображения разными способами.

5.12. Сканеры

Сканер выступает основным устройством ввода плоской растровой графики.

Сканер позволяет реализовать быстрое сканирование с высокой разрешающей способностью графического изображения, которое содержится на плоском носителе (бумага, непрозрачная пленка, негативная и позитивная прозрачная пленка, картон).

Определение

Сканером называют периферийное устройство ввода растровой графической информации и штрихкодов ручным или автоматизированным способом, которое предусматривает возможность подключения устройства автоматической подачи носителя, а также слайд адаптера для сканирования негативной и позитивной пленки.

Существует большое количество критериев классификации современных сканеров:

1. По функциональному назначению и набору решаемых задач:
 - сканеры для сканирования фотопленки (слайд-сканеры);
 - профессиональные планшетные сканеры;
 - полупрофессиональные планшетные сканеры;
 - планшетные сканеры для домашнего использования;
 - ручные сканеры для профессионального использования;
 - ручные сканеры для широкого круга пользователей;
 - ручные сканеры штрихкодов.
2. По разрешающей способности используемой чувствительной матрицы:
 - сканеры низкого разрешения (100-300 точек на дюйм) – ручные сканеры и сканеры штрихкодов, которые предназначены для выполнения частных задач;
 - сканеры среднего разрешения (300-1200 точек на дюйм) – ручные сканеры и планшетные сканеры, которые предназначены для пользователей-любителей;
 - сканеры высокого разрешения (600-4200 точек на дюйм) – планшетные и слайд сканеры, которые предназначены для профессиональной работы пользователей;
 - сканеры сверхвысокого разрешения (2400-16000 точек на дюйм) – планшетные и слайд сканеры, которые предназначены для сканирования с очень высоким качеством.
3. По возможности подключения дополнительного оборудования:
 - ручные или планшетные сканеры без возможности подключения устройств;
 - сканеры, которые предусматривают возможность подключения устройства автоматической подачи листов бумаги или пленки определенного формата
 - сканеры с поддержкой подключения слайд адаптера и лампы для подсветки;
 - сканеры, которые содержат интегрированный сетевой адаптер для работы в сети;
 - сканеры, входящие в основу факса для сканирования листов определенного формата;
 - многофункциональные центры, которые сочетают функции сканера и принтера.
4. По способу подключения сканера к системному блоку компьютера:
 - сканеры, подключающиеся через разъем на плате подключаемой к шине ISA, PCI;
 - сканеры, подключаемые посредством использования портов LPT и COM;
 - сканеры, подключаемые с помощью порта USB (Universal Serial Bus).

5.12.1. Планшетный сканер

Планшетный сканер имеет предметное стекло большого размера (формат А4, А3, А2 и так далее), которое позволяет сканировать графическое изображение с плоского носителя информации: бумага, картон, металл, позитивная и негативная прозрачная пленка, непрозрачная черно-белая и цветная пленка, деревянная пластина.

Определение

Планшетным сканером называют техническое средство ввода графического изображения с плоского носителя автоматизированным способом за один или более проходов чувствительной матрицы относительно поверхности прозрачного стекла.

Структурно-функциональная схема планшетного сканера представлена на рис. 5.36.

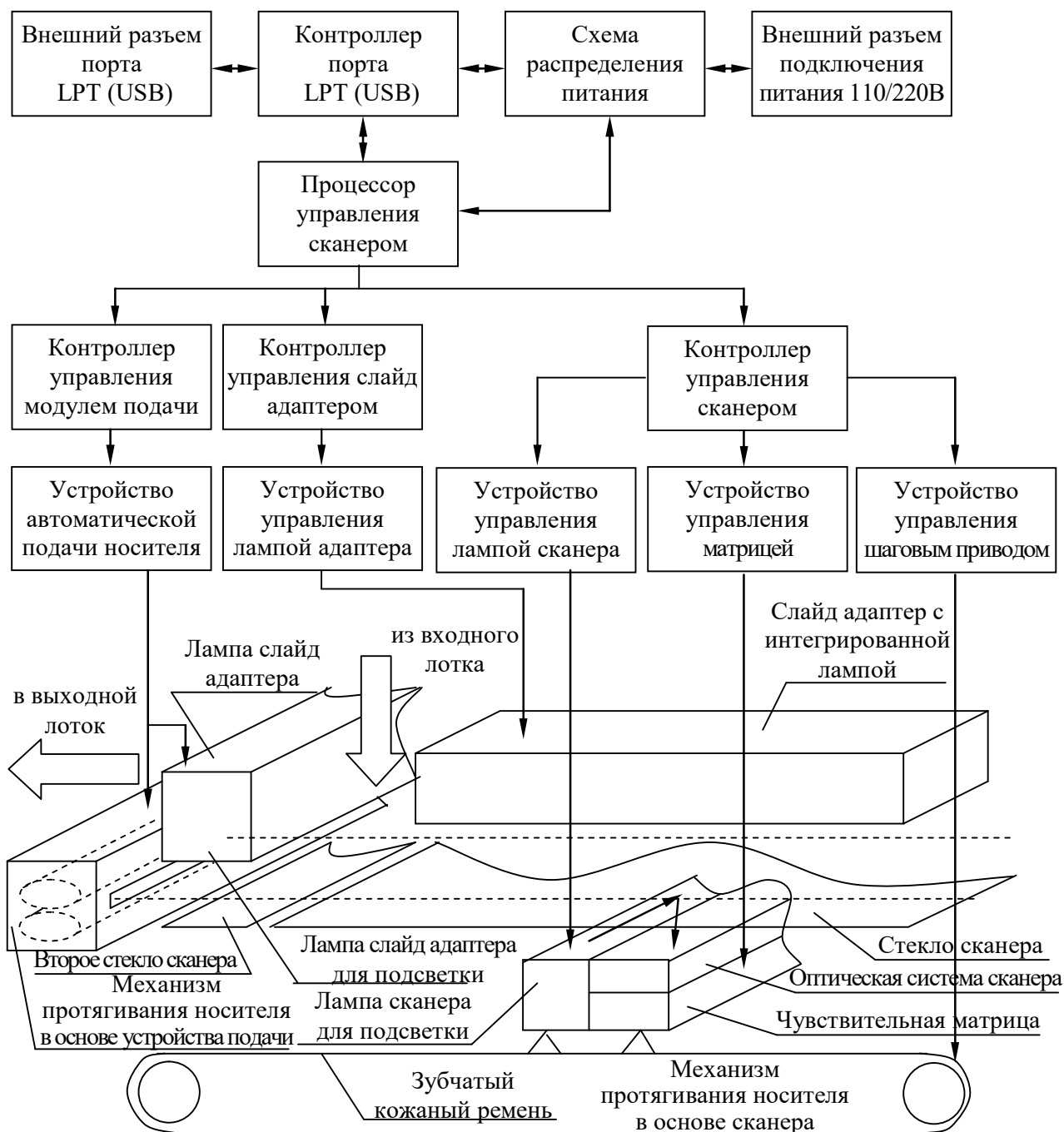


Рисунок 5.36. Структура планшетного сканера с возможностью подключения устройства автоматической подачи носителя и слайд адаптера

5.12.2. Ручной сканер

Ручной сканер позволяет реализовать сканирование произвольного фрагмента графического изображения с плоского непрозрачного носителя информации (бумага), обладает относительно большой компактностью, мобильностью и энергосбережением.

Определение

Ручным сканером (рис. 5.37) называют техническое средство ввода растровой графической информации содержащее кнопку, инициирующую начало сканирования посредством ручного перемещения его корпуса относительно плоского носителя (бумаги).

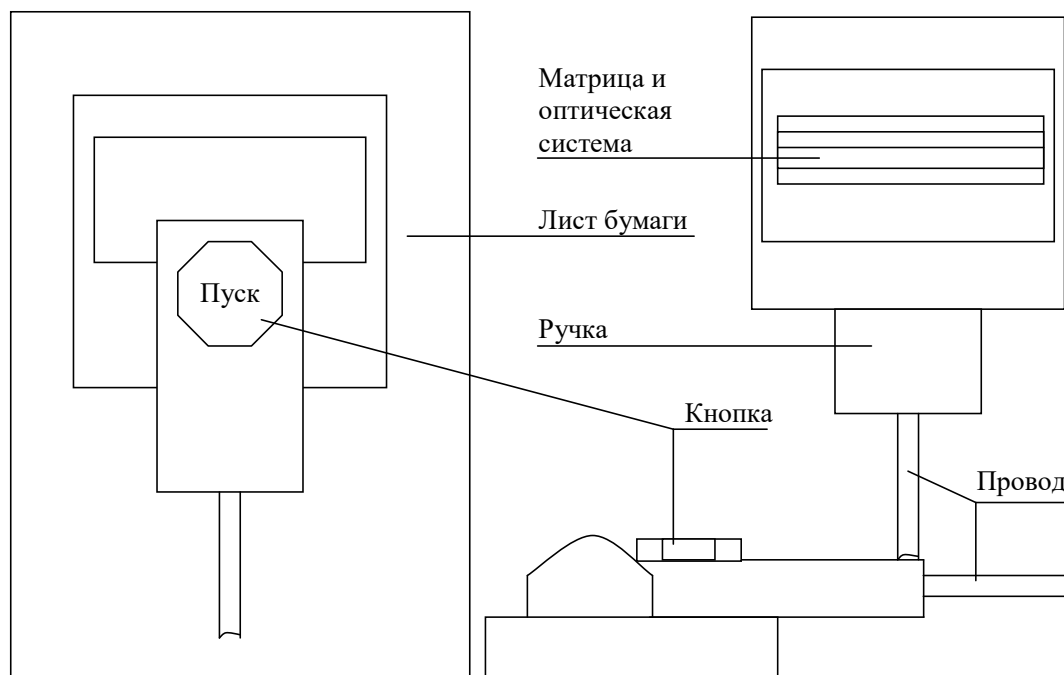


Рисунок 5.37. Особенности расположения элементов ручного сканера

На представленном рисунке обозначены различные части ручного сканера:

- матрица и оптическая система – световой поток, генерируемый интегрированной лампой отражается от поверхности носителя информации, а затем преломляется оптической системой и регистрируется чувствительной матрицей;
- ручка – предназначена для ручного передвижения корпуса сканера относительно непрозрачного носителя информации (бумага, картон, пластмасса, пленка);
- кнопка «Пуск» – нажатие инициирует начало процесса сканирования, а повторное нажатие пользователем на эту кнопку прекращает сканирование;
- провод – реализует возможность передачи управляющих сигналов и потока данных отражающего графическое изображение между процессором сканера и одним из портов (LPT, USB, COM) на системной плате компьютера.

Для обеспечения возможности сканирования графического изображения с плоского носителя посредством использования ручного сканера необходимо подключить его к разъему на системном блоке компьютера, установить драйвер и программное обеспечение, поместить поверхность сканера на плоский носитель информации, нажать кнопку и переместить ручной сканер относительно сканируемого изображения, нажать и отпустить кнопку для завершения процесса сканирования.

5.12.3. Сканер для сканирования штрих-кодов

Для реализации сканирования штрих-кодов используют сканер штрих-кодов, который обладает максимальной компактностью, но низкой разрешающей способностью.

Определение

Сканером штрих-кодов (рис. 5.38) называют техническое средство, которое обеспечивает регистрацию графического изображения соответствующего штрих-коду вручную посредством поднесения поверхности с изображением штрихов и цифр к его корпусу.

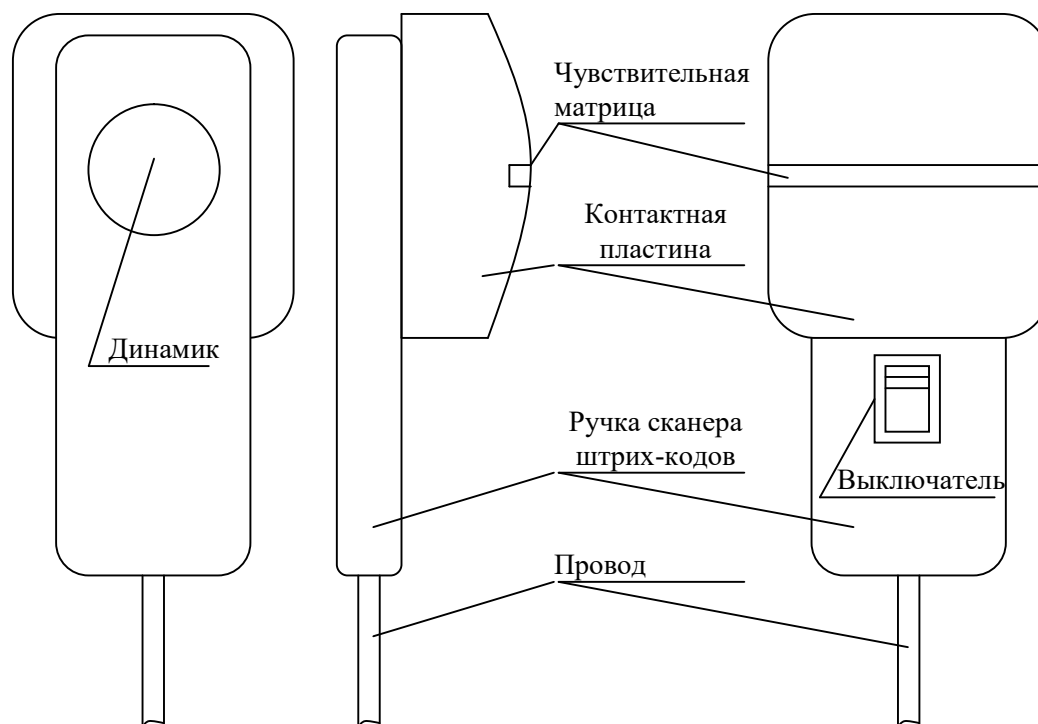


Рисунок 5.38. Особенности расположения элементов сканера штрих-кодов

Сканер штрих-кодов включает набор следующих элементов в основе конструкции:

- чувствительная матрица – световой поток, генерируемый интегрированной лампой отражается от поверхности со штрих-кодом, а затем преломляется встроенной оптической системой и регистрируется чувствительной матрицей;
- контактная пластина – обеспечивает плотное прижатие поверхности с чувствительной матрицей сканера к носителю информации содержащему штрих-код (некоторые модели сканеров штрих-кодов сканируют на расстоянии);
- ручка сканера штрих-кодов – предназначена для ручного передвижения корпуса сканера относительно непрозрачного носителя информации со штрих-кодом или с целью установки сканера в специальное крепление позволяющее подносить графическое изображение со штрих-кодом для сканирования на расстоянии;
- кнопка «Пуск» – нажатие инициирует начало процесса сканирования, а повторное нажатие пользователем на эту кнопку прекращает сканирование;
- выключатель – позволяет отключать и включать сканер при необходимости;
- динамик – сигнализирует факт однозначной идентификации штрих-кода;
- провод – позволяет передавать управляющие сигналы и поток данных отражающих графическое изображение штрих-кода через порт USB, LPT, COM.

6. Программное обеспечение современных компьютеров

ИТ направлены на оптимизацию и повышение эффективности деятельности социальных субъектов в различных предметных областях посредством внедрения средств автоматизации разного назначения, которые обеспечивают создание, распределение и использование информационных ресурсов, продуктов и услуг.

Вид ИТ зависит от динамически изменяющихся потребностей дифференцированных потребителей, особенностей процесса информационного обмена между социальными субъектами и техническими средствами, а также обусловлен структурой потоков информации и документооборота принятых в предметной области.

Современные ИТ обеспечивают автоматизацию процессов сбора, систематизации, обработки, накопления, поиска и распространения информации разного вида.

Для получения необходимой информации потребитель должен обратиться к ее источникам с помощью информационной системы. Доступ к внешним источникам информации обеспечивается посредством других информационных систем реализующих взаимодействие через вычислительные сети, каналы и магистрали, а обращение к внутренним источникам реализуется посредством собственных компонентов.

Решение того или иного круга задач (обработка данных, управление, поддержка принятия решений) требует активизации тех или иных компонентов информационной среды, которые обладают определенным уровнем автоматизации.

Программное обеспечение современных компьютеров подразделяется на несколько основных категорий, решает разные задачи широкого круга пользователей:

- системное программное обеспечение – локальные и сетевые однопользовательские и многопользовательские операционные системы для разных целей;
- оболочки операционных систем с командным типом интерфейса взаимодействия;
- прикладное программное обеспечение – программы для поддержки выполнения и автоматизации разных частных задач определенной категории пользователей;
 - пакеты прикладных программ для автоматизации документооборота в офисе;
 - пакеты прикладных программ для поддержки математических расчетов;
 - пакеты программ для реализации аналитически численных расчетов;
 - программы для автоматизации конструирования и моделирования;
 - программы для поддержки процесса имитационного моделирования;
 - программные системы автоматизации бухгалтерского учета организации;
 - пакеты прикладных программ для финансового анализа и аудита;
 - системы автоматизированного перевода текста с одного языка на другой;
 - программы для обслуживания аппаратного и программного обеспечения;
 - компиляторы, трансляторы, ретрансляторы, интерпретаторы и отладчики;
 - специализированные среды программирования на языках низкого уровня;
 - интегрированные среды программирования на языках высокого уровня;
 - интегрированные среды программирования для создания сетевых программ.

Современное программное обеспечение производится широким кругом отечественных и зарубежных производителей (Apple, IBM, Microsoft, 1С, Парус).

ПО существенно неоднородно по структуре поскольку оно реализует выполнение разных задач и функций при работе пользователей в информационных системах (рис. 6.1).

Локальные однопользовательские операционные системы	Сетевые однопользовательские операционные системы	Сетевые многопользовательские операционные системы
для промышленных контроллеров	MS-DOS 1.0-6.22	Linux, Unix (POSIX)
для локальных регуляторов	Linux 1.0-3.0	MS Windows 95/95 OSR2
для карманных компьютеров	Unix 1.0-2.0	MS Windows 98
для карманных переводчиков	OS/2 1.0	MS Windows 2000/XP
для калькуляторов	Macintosh	MS Windows Vista
Системное программное обеспечение		
Пакеты прикладных программ офисного назначения	Программное обеспечение для подавления вирусной опасности	Программное обеспечение для поддержки инженерных расчетов
тестовый редактор	антивирусный монитор	математический пакет
графический редактор	антивирусный сканер	статистический пакет
система электронных таблиц	сетевой экран (Firewall)	среда моделирования
система документооборота	монитор реестра	система конструирования
система бухгалтерского учета	система комплексной защиты	система разработки баз данных
Прикладное программное обеспечение		

Рисунок 6.1. Общая классификация системного и прикладного программного обеспечения

Системное ПО подразделяется на несколько базовых категорий (классов):

- однопользовательские и многопользовательские локальные и сетевые ОС;
- сервисные программы и утилиты для анализа аппаратного и программного обеспечения;
- программы мониторинга состояния среды программного окружения ОС, которые реализует поддержку периода исполнения ПО разного назначения;
- компиляторы, трансляторы и ретрансляторы, интерпретаторы и отладчики ПО;
- тесты аппаратного и ПО для реализации техобслуживания компьютера.

Современное прикладное ПО имеет следующую развернутую классификацию:

- офисные и коммуникационные программы, мультимедиа, гипермедиа, издательские системы, системы автоматического перевода, графические редакторы;
- инструментальные средства разработки ПО и среды программирования на языках высокого уровня, обеспечивающие разработку Интернет-приложений;
- инженерные пакеты программ для математического и статистического анализа, системы экономического анализа, системы поддержки принятия решений;
- проблемно-ориентированные программы: системы документооборота на предприятии, системы торгового и складского учета, системы финансового учета.

Прикладное ПО, сервисные программы и утилиты реализуют решение многих частных задач пользователя, а классификация ПО данного вида представлена на рис. 6.2.

Для автоматизации документооборота в организации	Для автоматизации перевода текста на разных языках	Для автоматизации конструирования и аналитических расчетов
текстовые редакторы	системы перевода текста	системы проектирования
системы электронных таблиц	электронные словари	системы моделирования
системы управления БД	системы перевода Web-ресурсов	системы численных расчетов
конструкторы презентаций	переводчики для Pocket PC	системы статистического анализа
издательские системы	словари для Pocket PC	пакеты анализа данных
почтовые системы	портативные переводчики	пакеты Data mining
среды разработки Web-ресурсов		
Пакеты прикладных программ		
Для автоматизации процесса обработки потокового аудио и видео	Для сканирования уровня безопасности информационных систем	Для снижения уровня вирусной опасности информационных систем
аудио- и видео-редакторы	сканеры уровня безопасности	комплексные системы безопасности
среды для создания анимации	сетевые сканеры безопасности	сетевые экраны
графические редакторы	средства борьбы со спамом	ПО для криптографии
Прикладные среды обработки и защиты информации		антивирусные программы
Пакет сервисных программ предназначенные для диагностики ПО	Пакеты утилит для конфигурирования и обслуживания ОС	Интегрированные среды разработки и отладки ПО, а также создания банков данных
системы комплексной защиты, диагностики и обслуживания ЭВМ	программы сканирования оптических дисков для лазерных накопителей	средства разработки архитектур и систем управления базами данных
пакеты программ для обслуживания файловой системы и восстановления данных	программы для оптимизации структуры и редактирования реестра	интегрированные среды программирования на языках высокого уровня
процедура восстановления поврежденного программного обеспечения и данных	программы диагностики аппаратного обеспечения ЭВМ	отладчики, компиляторы, интерпретаторы компьютерных программ
процедура резервного копирования данных	программы разметки накопителей на жестких магнитных и электронных дисках	средства разработки баз данных, информационных хранилищ и файл-серверов
средства дефрагментации файловой системы	мультимедиа драйверы	
архиваторы	конфигураторы расширенных функций ОС	
Пакеты сервисных программ и утилит		Пакеты для разработки ПО и создания инфологических схем реляционных баз данных

Рисунок 6.2. Классификация прикладного программного обеспечения

6.1. Современные интерфейсы взаимодействия

Интерфейс взаимодействия реализует возможность управления компьютером.

Определение

Интерфейсом взаимодействия называют совокупность директив и команд (командный) или графических элементов и правил их использования (графический), которые позволяют пользователю реализовать мониторинг состояния и управление операционной системой, а также прикладным программным обеспечением ЭВМ (рис. 6.3).

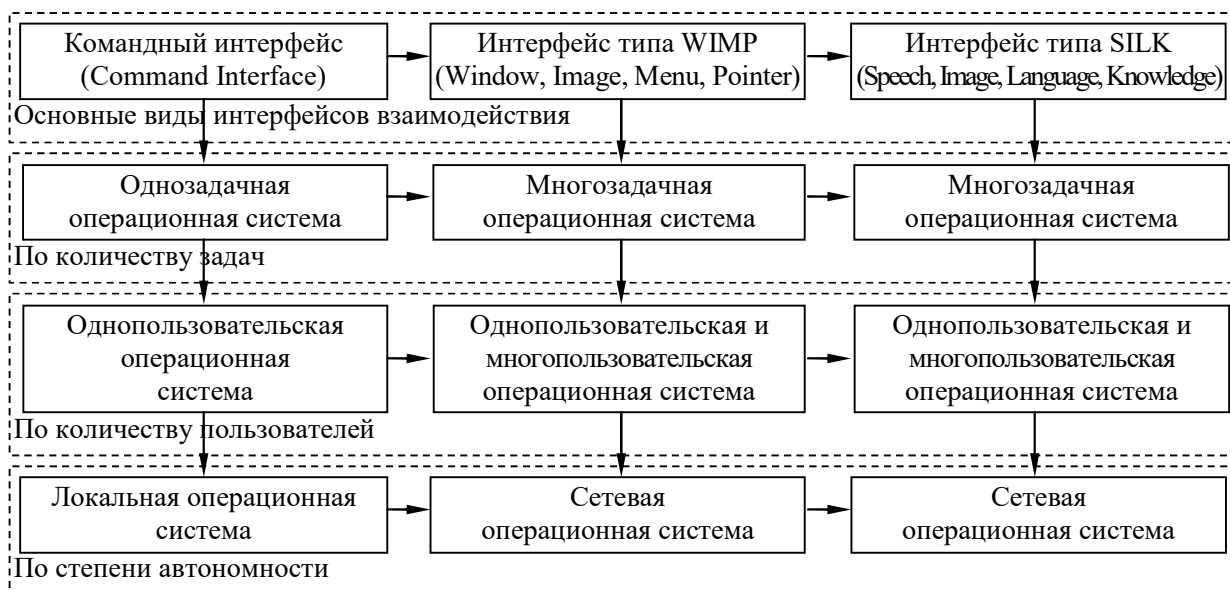


Рисунок 6.3. Классификация современных интерфейсов взаимодействия и их особенности

Определение

Командным интерфейсом называют вид интерфейса, который позволяет пользователю реализовать взаимодействие с операционной системой посредством использования совокупности директив и команд предназначенных для последующей обработки и исполнения командным процессором (интерпретатором команд).

Определение

Графическим интерфейсом называют разновидность интерфейса, который позволяет реализовать взаимодействие с операционной системой посредством использования совокупности графических элементов на основе правил работы с ними.

Существуют две распространенные разновидности графического интерфейса.

Определение

WIMP интерфейсом называют интерфейс взаимодействия, который предполагает использование набора окон (Window), пиктограмм файлов и папок (Image), меню (Menu), указателей манипулятора мышь и текстового курсора клавиатуры (Pointer).

Определение

SILK интерфейсом называют интерфейс, который предполагает использование совокупности голосовых команд на подмножестве естественного языка (Speech), пиктограмм (Image), нескольких локализаций совокупности идентификаторов его элементов (Language), а также справочной системы или базы знаний (Knowledge).

6.2. Системное программное обеспечение

Системное программное обеспечение представлено операционными системами, которые реализуют решение задач пользователей разного уровня подготовки.

Определение

Системным программным обеспечением называют операционные системы, которые позволяют реализовать мониторинг и управление всеми аппаратными и программными компонентами, которые непосредственно находятся в их основе (рис. 6.4).



Рисунок 6.4. Типовая структура операционной системы

6.2.1. Локальные однопользовательские операционные системы

Операционные системы позволяют решать множество прикладных задач мониторинга и управления при организации автоматизации крупных, средних и малых производств, технологических конвейеров сборки изделий и станков с числовым программным управлением, а также возникает необходимость их инсталляции на промышленных контроллерах, локальных регуляторах, роботах манипуляторах.

Локальные операционные системы (рис. 6.5) являются обособленными и находятся вне основных контуров управления производством, поэтому предусматривают включение или выключение осветительных установок, кондиционеров, холодильных установок в производственных помещениях организации разного профиля и назначения.

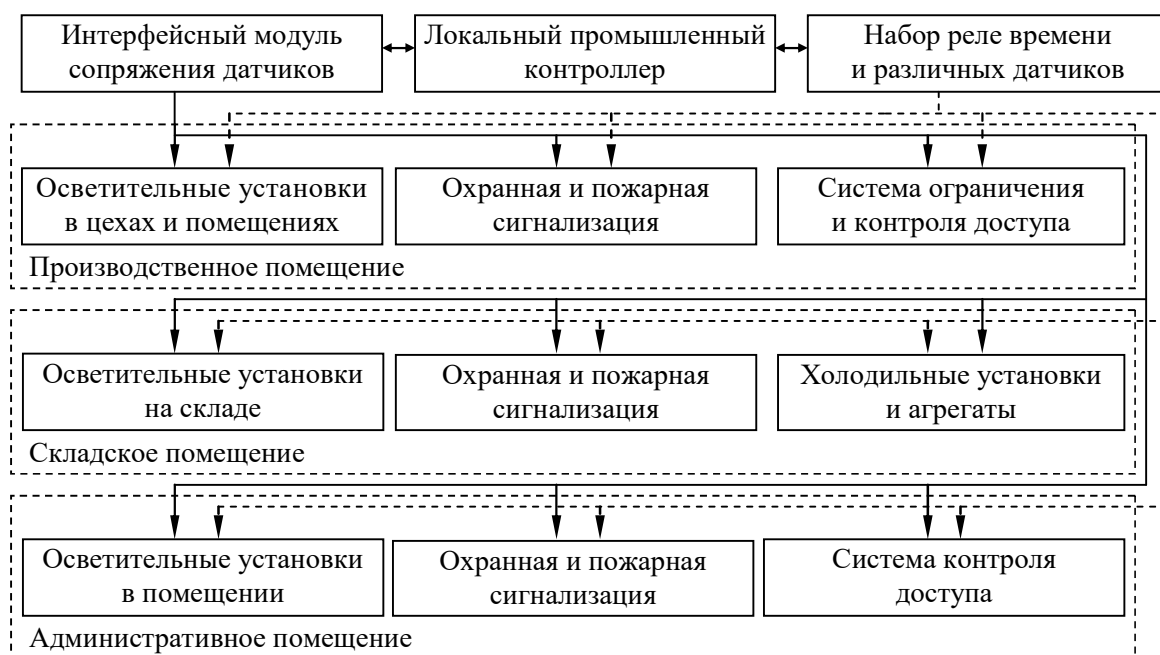


Рисунок 6.5. Автономная система управления агрегатами предприятия

Автономная система управления предполагает установку локальной операционной системы, которая выступает обособленной по отношению ко всем заделам технологического цикла производства и при этом обеспечивает анализ состояния датчиков охранной и пожарной сигнализации в административном, складском и производственном помещениях, а также реализует автоматическое включение и отключение осветительных установок, предполагает управление ограничением и контролем доступа.

Определение

Локальной операционной системой называют системное программное обеспечение, которое реализует непрерывный процесс обработки совокупности запросов инициированных прикладными программами и действиями пользователя на обособленном автоматизированном рабочем месте оснащенном компьютером.

Определение

Однопользовательской операционной системой называют локальную операционную систему, которая предназначена для работы одного пользователя непрерывно или с технологическими перерывами для реконфигурирования и отдыха.

6.2.2. Сетевые и многопользовательские операционные системы

Основной целью использования сетевых операционных систем является предоставление возможности создания рабочей группы пользователей или домена, которые позволяют реализовать открытый совместный доступ к совокупности разделяемых ресурсов доступных для использования в данной вычислительной сети.

Многопользовательские операционные системы предназначены для обеспечения возможности работы нескольких пользователей посредством ряда мероприятий:

- организационных – сеансы работы совокупности пользователей реализуются с разделением во времени: каждый пользователь работает в определенный интервал времени в рамках рабочей смены или суточного графика работы;
- технологических мероприятий – установка технических средств и систем, которые обеспечивают регулирование доступа к операционной системе.

Определение

Сетевой операционной системой называют операционную систему, которая функционирует в основе сегмента вычислительной сети и позволяет обрабатывать потоки запросов, инициированные совокупностью компьютеров расположенных на автоматизированных рабочих местах, а также реализует транспорт информационных пакетов, регулировку и балансировку сетевой нагрузки при передаче и обработке непрерывного потока данных исходящих от совокупности потребителей.

Определение

Многопользовательской операционной системой называют локальную или сетевую операционную систему, которая обеспечивает поддержку работы нескольких пользователей за одним компьютером на автоматизированном рабочем месте с разделением во времени посредством использования соответственно совокупности локальных профилей на компьютере или технологии блуждающего профиля пользователя (профиль хранится на сервере) при его работе в составе вычислительной сети.

Определение

Технология блуждающего профиля пользователя предполагает работу пользователя в локальной вычислительной сети с условием хранения его профиля на сетевом сервере, а также предусматривает возможность использования профиля определенного пользователя на любом компьютере для реализации аутентификации.

Определение

Аутентификацией называют ввод пользователем данных учетной записи пользователя (логин и пароль), которая предварительно выдана автоматизированным или ручным способом для последующего использования в данной локальной сети.

Выделяют несколько технологий при организации сегмента вычислительной сети:

- доменная архитектура применяется при работе пользователей в составе домена;
- одноранговая архитектура – применяется при работе большого количества пользователей разных категорий, которые совместно решают задачи в составе рабочей группы при работе в локальной вычислительной сети организации.

Определение

Доменом называют программный контролер сегмента локальной вычислительной сети (контроллер домена сегмента вычислительной сети), который реализует управление транспортными и файловыми серверами (если такие имеются), а также непрерывную обработку потоков запросов инициированных аутентификацией пользователей, разграничением доступа при совместном использовании информационных ресурсов, получением информации из банка данных или БД (рис. 6.6).

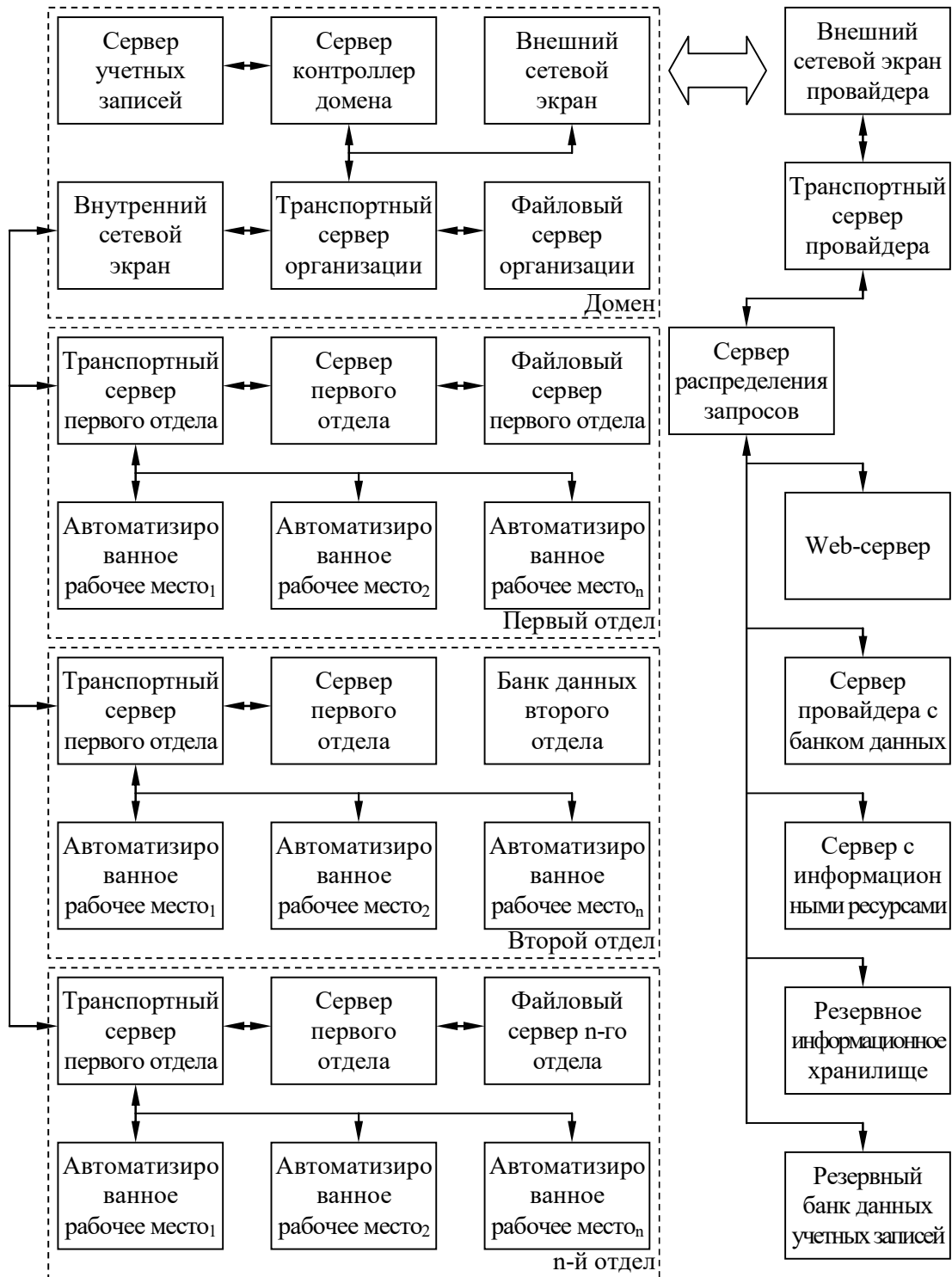


Рисунок 6.6. Особенности организации сегмента локальной вычислительной сети с использованием доменной архитектуры

При работе в составе вычислительной сети предоставляется открытый доступ к совокупности информационных ресурсов, которые используются всеми пользователями при решении разных задач, поэтому возникает необходимость создания нескольких рабочих групп на серверах в определенных подразделениях организации.

Определение

Рабочей группой называют совокупность компьютеров (ЭВМ) расположенных на автоматизированных рабочих местах, которые используются определенными пользователями в одноранговой вычислительной сети при выполнении производственной или непроизводственной функции или предписанной задачи (рис. 6.7).

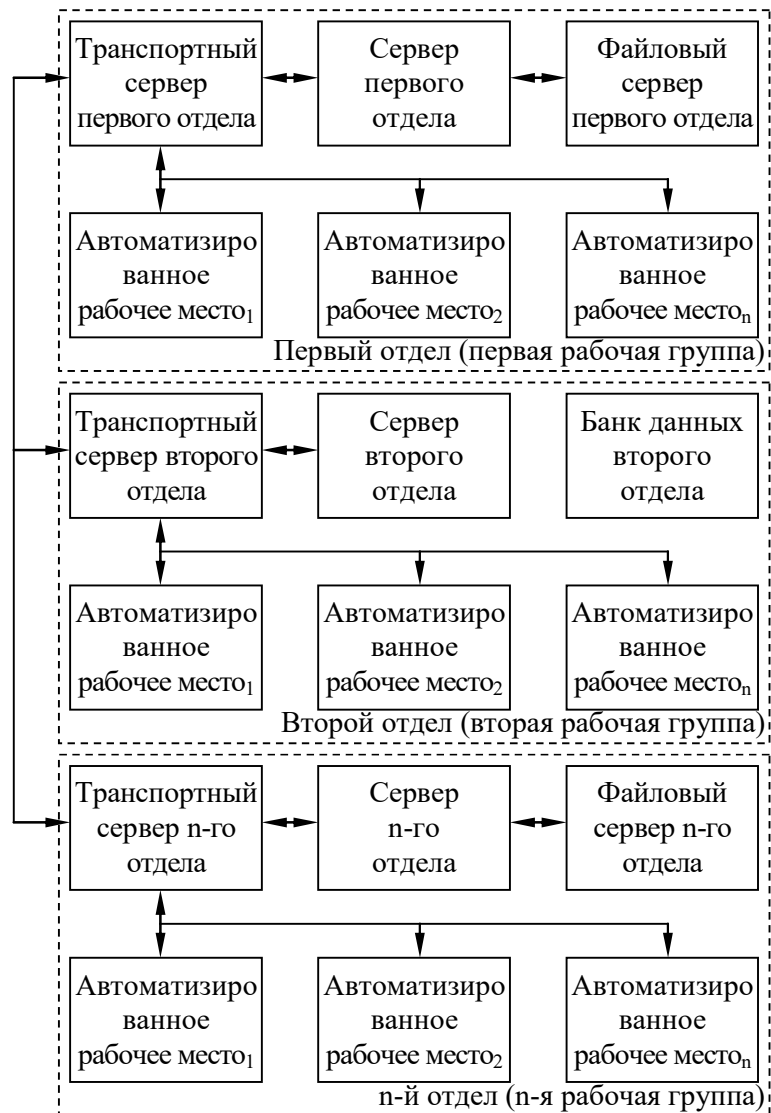


Рисунок 6.7. Особенности организации сегмента локальной вычислительной сети с использованием доменной архитектуры

Транспортный сервер отдела – реализует возможность передачи информации разного вида, которая используется определенной рабочей группой пользователей.

Сервер отдела – содержит в своей основе вспомогательную БД учетных записей, которая конфигурируется системным администратором на уровне отдела и позволяет обеспечить разграничение прав доступа пользователей к различным информационным ресурсам файлового сервера данного отдела и соседних отделов организации.

6.2.3. Оболочки

Оболочка представляет собой программное обеспечение для оптимизации работы пользователя с файловой системой в операционной системе с командным интерфейсом взаимодействия или для облегчения работы с файловой системой в операционной системе с графическим интерфейсом взаимодействия, поэтому реализует:

- возможность монтирования и демонтажа локальных и сетевых логических дисков для последующего их использования в операционной системе;
- поддержка визуализации процесса множественного выделения файлов и папок;
- обеспечение наглядности процесса копирования, перемещения, создания и удаления файлов и папок на локальных и сетевых накопителях информации;
- возможность создания меню пользователя для выбора набора предустановленных операций, которые подлежат автоматизированному выполнению;
- возможность автоматизированной фильтрации элементов файловой системы в операционной системе по виду (файл или папка), расширению: непосредственно выполняемые `[* .exe]cutable` – исполняемые, `[* .com]mand` – командные, `[* .bat]ch` – пакетные; открываемые посредством использования вспомогательного приложения `[* .doc]ument`, `[* .txt]textual`, `[* .bmp]bitmap`, `[* .reg]registry`.

Определение

Оболочкой называют программное обеспечение, которое устанавливается в определенной операционной системе с целью оптимизации доступа к файловой системе компьютера, а также для повышения эффективности работы пользователя.

Классификация оболочек для операционных систем с командным и графическим интерфейсом взаимодействия с пользователем представлена на рис. 6.8.

Оболочки для операционных систем с командным интерфейсом	Оболочки для операционных систем с графическим интерфейсом	Оболочки операционных систем портативных компьютеров
Norton commander	Windows commander	Norton commander
Volcov commander	Windows navigator	Far for Pocket PC
DOS navigator	FAR for Windows	Windows commander

Рисунок 6.8. Классификация оболочек для операционных систем с командным и графическим интерфейсом взаимодействия

Оболочки для операционных систем, которые поддерживают командный и графический интерфейс взаимодействия с пользователем устанавливаются непосредственно в самой операционной системе, но при этом имеется ряд особенностей:

- в операционной системе с командным интерфейсом взаимодействия – установка и запуск осуществляется непосредственно в операционной системе;
- в операционной системе с графическим интерфейсом взаимодействия – предусматривается возможность запуска оболочек двух типов: оболочек для операционных систем с графическим интерфейсом пользователя (запускаются как обычные приложения) и операционных систем с командным интерфейсом (запускаются только из режима эмуляции сеанса MS-DOS в отдельном окне или в полноэкранном режиме работы определенного пользователя).

6.2.4. Программное обеспечение для промышленных контроллеров

Промышленные контроллеры обеспечивают автоматизацию мониторинга и управления сложными объектами и процессами посредством специального:

- программного обеспечения – операционные системы специального назначения, процедуры и программы для реализации локального управления;
- аппаратного обеспечения – технические средства мониторинга (датчики), интерфейсы и преобразователи между входами устройств разного типа.

Определение

Промышленным контроллером называют аппаратно-программный комплекс со специальной архитектурой и оптимизированным набором компонентов, который предназначен для реализации наблюдаемости и управляемости сложного технологического процесса в различных сферах деятельности (промышленности).

Сфера практического использования контроллеров разного типа широка:

- складская деятельность – мониторинг состояния конвейера, который обеспечивает перемещение и доставку номенклатурных единиц сырья и товаров;
- охрана помещений – управление включением и выключением оборудования;
- производство – управление производственным циклом сборки сложной многономенклатурной продукции, который состоит из ряда технологических заделов;
 - автоматизированная сборочная линия – содержит последовательность роботов-манипуляторов реализующих выполнение определенной операции;
 - станки с числовым программным управлением – непосредственно реализуют расточку детали по заданной программе или алгоритму на основе чертежа.

Контроллер включается контур управления определенным образом (рис. 6.9).

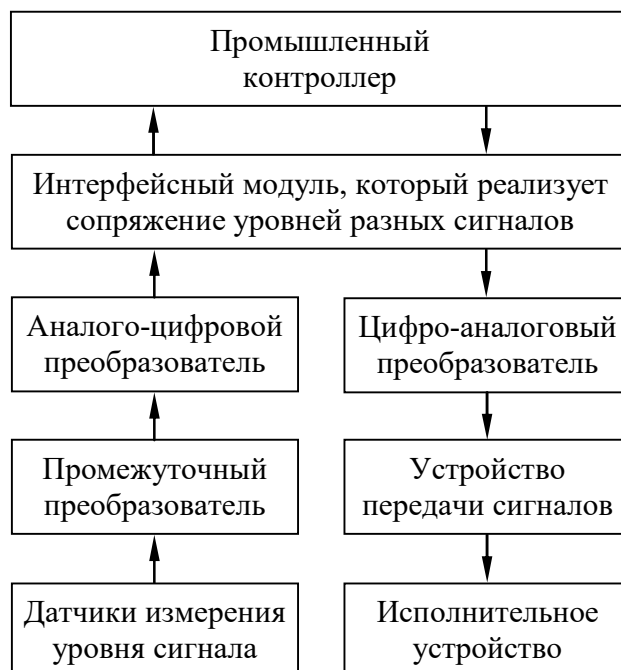


Рисунок 6.9. Схема включения промышленного контроллера для управления объектом или процессом

Определение

Датчиком называют техническое средство измерения значения уровня сигнала соответствующего определенной физической величине, который преобразуется в электрический сигнал для дальнейшего использования разными потребителями.

Иногда возникает существенная сложность при регистрации состояния наблюдаемого объекта, которая обусловлена низким уровнем мощности сигнала, поэтому специалистами в области автоматизации применяются разные усилители.

Определение

Усилителем (усилительным каскадом) называют техническое средство (последовательность устройств), которое обеспечивает увеличение номинального значения уровня эффективного сигнала поступающего на вход от источника (объекта).

Поскольку цифровой автомат (компьютер) является дискретным устройством, которое реализует непрерывную обработку данных, то необходимо разработать алгоритмическое и программное обеспечение для реализации задач мониторинга и управления сложными технологическими процессами и производствами продукции.

Промышленный контроллер выступает основным узлом на всех уровнях и технологических заделах сложного наукоемкого производства, которое позволяет собрать совокупность узлов и компонентов для осуществления сборки изделия.

Выделяют ряд категорий промышленных контроллеров исходя из их местоположения в контуре управления сложными технологическими объектами и системами, которые реализуют организационную, сырьевую и технологическую подготовку производства сложной многономенклатурной продукции в рамках нескольких заделов:

- контроллеры для реализации принципа супервизорного управления – используется в многоуровневых системах управления, которые реализуют управление сложными единичными, серийными или массовыми производствами;
- контроллеры для обеспечения управления цехом – реализуют мониторинг состояния и управление всем технологическим оборудованием в данном цехе;
- контроллеры управления автоматизированной сборочной линией – обеспечивают управление многоэтапным процессом сборки определенного изделия;
- контроллеры для реализации локального управления разными роботами-манипуляторами – позволяют реализовать контроль выполнения элементарной операции в рамках определенного технологического задела (этапа производства).

Контроллеры позволяют автоматизировать и существенно повысить эффективность производства, сократить транзакционные и временные издержки при выполнении различных операций в рамках последовательности технологических заделов.

Практическое использование контроллеров позволяет реализовать существенную экономию электроэнергии на освещении производственных цехов, общехозяйственных и общеадминистративных издержках, техническом и сервисном обслуживании сборочных конвейеров и линий, которые сопутствуют сложному многономенклатурному наукоемкому производству заданной ассортиментной группы продукции.

Количество отделов и цехов предприятия, а также совокупность номенклатурных единиц продукции в рамках одной или нескольких ассортиментных групп накладывает ограничения и позволяет выделить некоторые закономерности функционирования производственного цикла, информационные связи между разными технологическими заделами, поэтому перечисленные особенности существенно влияют на технические характеристики потенциального набора используемых промышленных контроллеров.

Структурная схема размещения совокупности контроллеров на различных уровнях управления сложным многономенклатурным производством системных плат и прочих компонентов и узлов персональных компьютеров представлена на рис. 6.10.

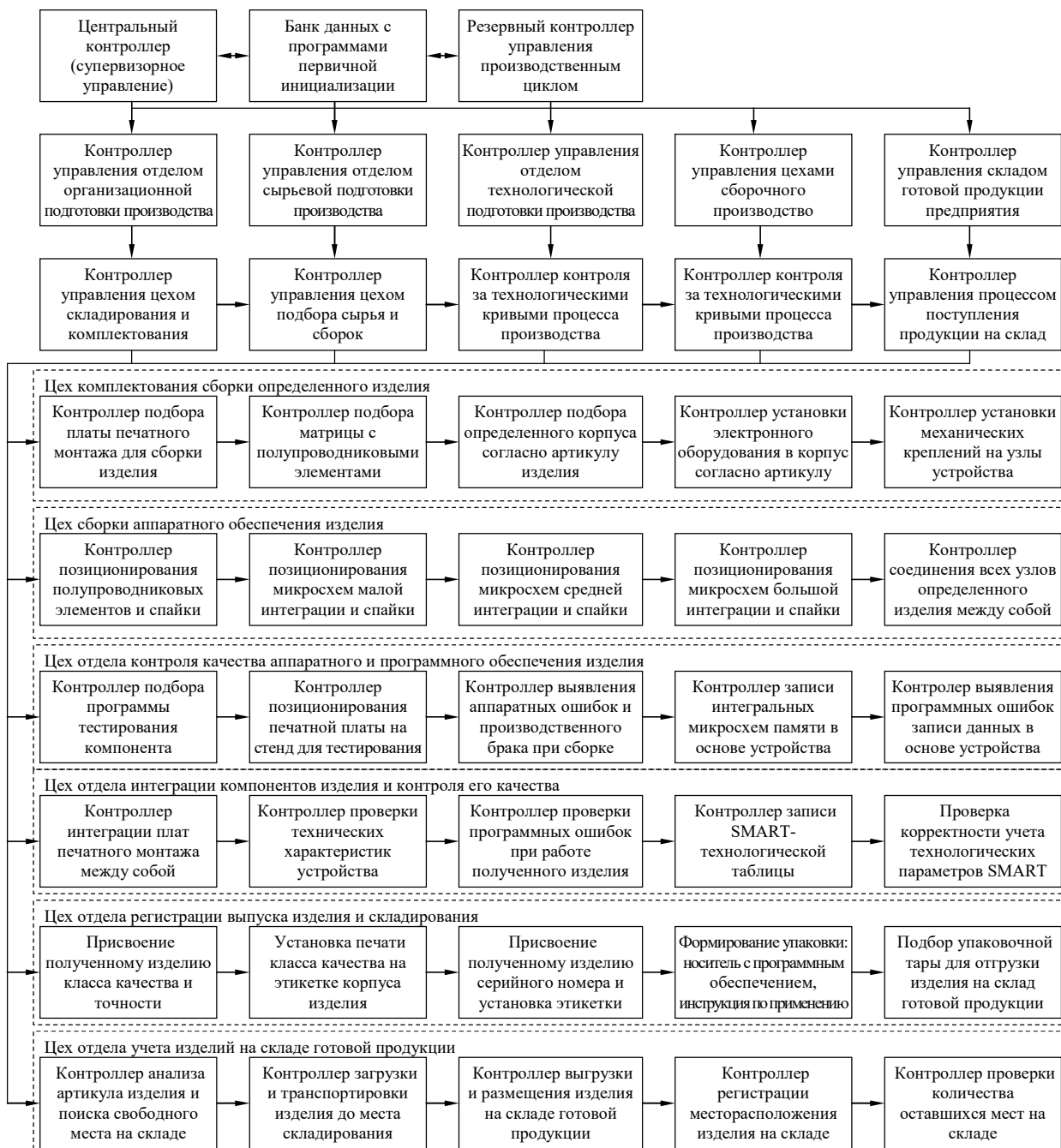


Рисунок 6.10. Схема использования промышленных контроллеров при реализации принципа супервизорного управления сложным производством
Со склада готовой продукции изделия отгружаются конечному потребителю.

6.3. Прикладное программное обеспечение

Прикладное программное обеспечение позволяет реализовать автоматизацию решения частных задач пользователей и оптимизировать выполнение определенных видов производственной и непроизводственной деятельности всех сотрудников:

- графический дизайн с использованием растровой графики (Photoshop, Corel Draw);
- конструирование и строительный дизайн (AutoCad, ArchiCad);
- бизнес планирование и сетевые графики планирования (BP Win);
- бухгалтерский учет и аудит финансово-хозяйственной деятельности (Парус, 1С);
- рейтинг показателей эффективности функционирования организации по совокупности данных первичной и итоговой финансовой отчетности (Balanced Score Card);
- системы поддержки крупных производств (ERP – Enterprise Resource Planning);
- системы моделирования информационных процессов на основе стандартов IDEFx;
- система реализации финансового анализа и управления распределенным многономенклатурным производством и прогнозирования показателей рентабельности, ликвидности и безубыточности, платежеспособности (SAP R3);
- моделирование реакций и процессов (ChemLab, MathLab и набор тулбоксов);
- аналитически-численные расчеты на основе формальных моделей (MathLab);
- математические расчеты на основе численных моделей (Matematika, MathCad);
- математическая обработка апостериорных данных экспериментов посредством использования набора статистических методов анализа (Statistica, SPSS);
- пакеты программ офисного назначения для поддержки документооборота (MS Office);
- пакеты прикладных программ для автоматизации создания функциональных схем различных логических устройств (B&R Lad Diagram constructor);
- многоуровневые распределенные системы автоматизированного и открытого дистанционного обучения: электронные системы поиска информационных ресурсов, библиотеки, электронные учебники, диагностические модули для оценки уровня остаточных знаний контингента обучаемых (Open.Net);
- программное обеспечение для реализации перевода текста с одного национального языка на другой иностранный язык (Сократ, Prompt XP Family);
- интегрированные среды программирования на языках высокого уровня (Borland C++ Builder, Borland Delphi, Borland J++ Builder, ASP.Net Framework);
- среды низкоуровневого программирования, компиляторы и отладчики программного обеспечения для контроллеров и адаптеров (Assembler, Soft Ice).

Определение

Прикладным программным обеспечением называют совокупность программ и программных модулей, которые обеспечивают поддержку решения частных задач пользователя в определенной предметной области: автоматизация аналитически-численных и математических расчетов, дизайн, конструирование, проектирование, моделирование, бухгалтерский учет, финансовый анализ и аудит.

6.3.1. Пакеты прикладных программ офисного назначения

ИТ документооборота предполагает построение бизнес-процессов позволяющих построить модель отражающую особенности документооборота в организации.

Сложность создания, внедрения и практического использования средств автоматизации документооборота на предприятии зависит от ряда ключевых факторов:

- уровня вертикальной интеграции организационной структуры предприятия;
- степени диверсификации направлений деятельности предприятия при производстве товаров, выполнении работ, оказании различных услуг потребителям;
- набора и класса решаемых задач и выполняемых функций при осуществлении перечня уставных видов деятельности в определенном сегменте рынка.

Выделяют разные средства автоматизации документооборота на предприятии:

- пакеты прикладных программ для автоматизации документооборота в офисе, которые предназначены для использования в ИС различного назначения:
 - обработка текстовых документов и массивов документов на носителях;
 - прием/передача электронной почты по разным вычислительным сетям;
 - вычислительные процедуры для реализации обработки и анализа данных;
 - создание и редактирование деловой графики, презентаций, графических объектов, в частности статических и динамических схем и иллюстраций;
 - ведение БД больших объемов с использованием сложных структур данных: картотека и каталог, справочник и оперативные учетные данные (журналы);
 - использование информационных ресурсов сетей Интернет/интранет;
 - создание персональной информационной системы пользователя и т.п.
- пакеты прикладных программ для реализации автоматизации финансового анализа и бухгалтерского учета на предприятиях малого и среднего бизнеса;
- системы стратегического планирования ERP (Enterprise Resource Planning) – система позволяющая обеспечить комплексную автоматизацию процессов:
 - поиск сырьевой базы, новых технологий и средств для производства продукции;
 - учет поставок сырья и исходных ингредиентов для обеспечения производства;
 - складирование и размещение номенклатурных единиц продукции на складе;
 - отгрузка номенклатурных единиц продукции со склада в сборочные и производственные цеха для поддержки функционирования конвейера или линии;
 - контроль процесса расходования сырьевой базы на всех этапах и технологических заделах производства ряда номенклатурных единиц продукции;
 - результаты использования системы менеджмента качества при использовании конвейера для сборки узлов и агрегатов, а также анализ соответствия готовой продукции действующим требованиям и стандартам качества;
 - контроль отгрузки номенклатурных единиц на склад готовой продукции;
 - учет и контроль отгрузки выпущенной продукции со склада готовой продукции совокупности внешних контрагентов посредством сбытовой сети;
 - маркетинговые исследования конъюнктуры рынка для задач планирования.

При рассмотрении пакета прикладных программ офисного назначения MS Office 2000/XP можно выделить ключевую особенность и преимущественное достоинство – удобство интерфейса и простоту использования для конечного пользователя.

Возможность легкого освоения набора программных компонентов пользователями с разным уровнем владения средствами вычислительной техники: бухгалтерами, инженерами, журналистами, преподавателями, студентами, школьниками. Все без исключения программные компоненты и модули программы имеют удобный графический интерфейс, а также предоставляют мощные средства навигации для реализации большого количества задач из различных предметных областей.

Пакет Microsoft Office 2000/XP является дальнейшим шагом в развитии функциональных возможностей предыдущих версий пакета. Появляются дополнительные приложения и расширяется набор функций (табл. 6.1), но имеются технические ограничения при выполнении регламентированных задач пользователей (табл. 6.2).

Таблица 6.1

Сравнительная характеристика возможностей пакета прикладных программ MS Office 97 и 2000

Характеристики	MO97	MO2000
Помощник MS Office для работы со справочной системой	+	+
Создание Web-страниц в формате HTML, XML	+	+
Просмотр Web-документов с вложениями и скриптами	+	+
Панель MS Office для запуска программ и открытия документов	+	+
Возможность создания перечня контактов в Outlook Express	+	+
Язык программирования Visual Basic и элементы ActiveX	+	+
Редактирование документов в формате HTML посредством использования технологии Dynamic Data Exchange	+	+
Разработка Web-страниц с использованием множества профилей, схем и различных элементов оформления контента	–	+
Совместная работа пользователей над документами в локальных интрасетях и глобальной WWW (World Wide Web)	–	+
Просмотр Web-страниц поддерживается непосредственно из браузера или родственного приложения (MS Word, Excel, Access)	+	+
Манипулирование данными и их редактирование посредством использования данных разных приложений в редакторе Front Page	–	+
Страницы доступа к данным для связи с БД, просмотра, редактирования и ввода информации в обозревателе Explorer	–	+
Многоуровневый буфер обмена для хранения информации	–	+
Плавающие таблицы с поддержкой их перемещения в документе	–	+

**Технические ограничения пакета прикладных программ MS Office 2000
при совместной работе пользователей**

Параметр	Ограничение на значение
Наибольшее число пользователей, которые могут и способны одновременно открывать общий документ (книгу)	256
Наибольшее число страниц, строк или столбцов (ячеек) в документе (файле)	Ограничивается объемом памяти
Максимальное число записей, поддерживаемое журналом изменений документа	32000 (по умолчанию 30 дней)
Число одновременно объединяемых документов Word и книг Excel	Ограничивается объемом памяти
Количество ячеек, которые могут быть выделены в книге Excel	32767
Число цветов для обозначения изменений, вносимых разными пользователями	32

Обработка данных реализуется отдельными приложениями путем настройки команд меню и панелей инструментов, а также посредством создания новых объектов (шаблоны документов, стили форматирования, макросы, списки, таблицы) и программных модулей на языке программирования Visual Basic for applications.

На базе пакета прикладных программ MS Office 2000/XP создаются документы в ходе производственной и непроизводственной деятельности персонала на рабочих местах, которая направлена на выполнение разных должностных обязанностей.

Пакет прикладных программ офисного назначения MS Office 2000/XP включает набор компонентов реализующих автоматизацию создания, пересылки и форматирования различных видов документов и информационных ресурсов в ИС:

- текстовый редактор (процессор) Word – позволяет редактировать документы;
- система электронных таблиц Excel – поддерживает аналитические расчеты;
- система управления БД Access – создание БД и их инфологических схем;
- программа Power Point – подготовка и отображение мультимедиа-презентаций;
- программа Outlook Express – чтение и доставка электронной корреспонденции;
- система Publisher – настольная издательская система книжной продукции;
- редактор Front Page – позволяет создавать Web страницы Web-серверов;
- Info Path – система разработки форм документов и планировщик задач;
- программы конфигурирования – мастера конфигурирования и модули для настройки компонентов пакета прикладных программ MS Office 2000/XP:
 - мастер активации приложения – позволяет зарегистрироваться в Microsoft;
 - мастер сохранения настроек – позволяет перенести параметры конфигурации с одного компьютера на другой при переустановке пакета программ;
 - мастер восстановления приложений – реализует процедуру восстановления.

6.3.2. Прикладные программы для автоматизации бухгалтерского учета

Комплексные системы бухгалтерского учета обеспечивает автоматизацию процессов подготовки отчетности согласно требованиям бухгалтерского учета действующим на территории определенного государства. Нормативно-правовой основой для осуществления бухгалтерского учета является действующее законодательство в области административно-процессуального и налогового законодательства, учетная политика организации, разработанная модель бухгалтерского учета предполагающая наличие рабочего плана счетов, что позволяет определить совокупность подходов, методов и технологий постановки и ведения бухгалтерского учета, включая несколько важных задач: регистрация первичных документов-оснований разных финансово-хозяйственных операций, их стоимостное измерение, группировка и итоговое обобщение полученных результатов деятельности предприятия.

Ключевыми элементами учетной политики организации выступают:

- рабочий план счетов бухгалтерского учета на основе действующих норм;
- формы документов для формирования бухгалтерской и налоговой отчетности;
- порядок проведения инвентаризации активов и обязательств организации;
- методы оценки активов, пассивов, дебиторской и кредиторской задолженности;
- правила документооборота и технология обработки учетной информации;
- технологии контроля всех совершенных финансово-хозяйственных операций.

План счетов бухгалтерского учета выступает систематизированным перечнем синтетических счетов бухгалтерского учета первого и второго порядка позволяющих составлять совокупность форм отчетности для проведения внутреннего и внешнего аудита, а также для реализации возможности расчета системы нормативных показателей, обеспечивающих возможность проведения финансового анализа.

План счетов бухгалтерского учета содержащий исчерпывающий перечень счетов регламентируется действующим законодательством (для кредитных организаций – инструкция №205П-ФЗ) и является основой для формирования рабочего плана счетов обеспечивающего возможность разработки модели бухгалтерского учета.

В рабочем плане счетов бухгалтерского учета вводятся аналитические счета исходя из вида деятельности организации и особенностей реализуемых операций.

Счета предназначены для группировки и текущего учета различных финансово-хозяйственных операций, а также имеют существенные особенности строения:

- активные счета – предназначены для группировки и реализации текущего учета имеющихся средств по их составу и эффективности их размещения;
- пассивные – предназначены для реализации учета источников привлечения и образования средств (финансовых ресурсов) по их целевому назначению.

Каждая финансово-хозяйственная операция корреспондируется документом-основанием, который вносится в журнал документов-оснований и отражается посредством двойной записи в бухгалтерском балансе в зависимости от типа счетов (активный или пассивный), в любой момент возможен просмотр различных срезов.

При постановке и ведении бухгалтерского учета в кредитной организации все операции регистрируются в регистрах бухгалтерского учета в хронологической последовательности и группируются по соответствующим счетам. Компьютерные программы в зависимости от потребностей пользователя отражают различные срезы бухгалтерского баланса, отображают проводки соответствующие определенным хозяйственным операциям, обеспечивают возможность сторнирования при удалении и коррекции документов, а также позволяют использовать вспомогательные документы в процессе проведения финансового анализа. Соответствие реальных финансово-хозяйственных операций и корректность проведенных операций посредством регистров обеспечивают лица, подписавшие документы и вносящие сведения в БД, а также субъекты реализующие аудит. Содержание регистров бухгалтерского учета и финансовой отчетности является коммерческой тайной, поэтому создается служба внутреннего аудита, но иногда привлекаются внешние аудиторы.

В зависимости от последовательности использования и количества учетных регистров использующихся для регистрации операций выделяют различные формы отчетности, выступающие информационной основой для финансового анализа.

В кредитных организациях используется мемориально-ордерная форма счетоводства – до группировки однородных финансово-хозяйственных операций на счетах происходит оформление мемориальных ордеров, в которых указываются: краткое содержание операции, сумма операции, проводка включая задействованные счета.

При разработке ИТ и средств автоматизации бухгалтерского учета и аудита необходимо ориентироваться на действующие правила определяющие основные регистры бухгалтерского учета в кредитной организации, к которым относят:

- бухгалтерский журнал документов – предназначен для возможности регистрации всех операций и документов, проведенных за учетный день, он прилагается к ежедневному балансу и бухгалтерским документам операционного дня;
- кассовые журналы операций – обеспечивают возможность регистрации учетных операций по выдаче и приему наличной валюты, оформленный документ помещается в кассовые документы дня и прилагается к бухгалтерскому журналу;
- ведомость остатков по счетам – составляется ежедневно по окончанию операционного дня, ведется по счетам синтетического учета кредитной организации первого и второго порядка, балансовым, внебалансовым и лицевым счетам;
- ведомость остатков привлеченных и размещенных средств – ведется посредством использования средств автоматизации, ежедневно выводится на печать, в документе указываются: дата привлечения и размещения различных видов инструментов, номер счета второго порядка, остаток средств и срок погашения;
- оборотная ведомость по счетам кредитной организации – ведется в разрезе счетов первого и второго порядка, наименования разделов и счетов баланса, содержит входящий остаток на указанную дату, оборот по активным и пассивным счетам за отчетный период, исходящие остатки на отчетную дату.

В зависимости от способов обобщения и группировки информации о финансово-хозяйственных операциях на активных и пассивных счетах (табл. 6.3) выделяют:

- счета синтетического учета (1 пор.) – предназначены для отражения состояния движения средств, их источников в обобщенном виде и денежном выражении;
- счета аналитического учета (2 пор.) – открываются в расширение определенного синтетического счета, при этом сальдо синтетического счета выступает суммой всех аналитических счетов, а сумма оборота синтетического счета равна сумме всех оборотов аналитического счета по отношению к контрагентам.

Аналитический учет ведется путем открытия и ведения лицевых счетов по видам валют, клиентам, банкам-корреспондентам, видам ссуд, прочим признакам.

Лицевые счета ведутся на отдельных листах и карточках с указанием: даты совершения операции, учетного номера документа, шифра (кода) операции, номера корреспондирующего счета, суммы оборотов и остатки по дебету и кредиту.

Таблица 6.3

Структура счетов входящих в основу плана счетов

№ п.п	Структурная единица (элемент) кода	Количество знаков			
		Корреспондирующие счета по учету средств клиентов	Счета по учету кредитов	Бюджетные счета	Счета по учету доходов и расходов
1.	Номер разряда плана счетов	1	1	1	1
2.	Номер счета первого порядка, который в каждом разделе начинается с №01	2	2	2	2
	Общее количество знаков	3	3	3	3
3.	Номер счета второго порядка	2	2	2	2
	Общее количество знаков	5	5	5	5
4.	Код валюты или драгоценного металла	3	3	3	3
	Общее количество знаков	8	8	8	8
5.	Защитный ключ (требования Банка России)	1	1	1	1
	Общее количество знаков	9	9	9	9
6.	Номер структурного подразделения	4	4	4	4
7.	Символ бюджетной отчетности	-	-	3	-
8.	Символ отчета о прибылях и убытках	-	-	-	5
9.	Порядковый номер лицевого счета	7	7	4	2
10.	Итого знаков	20	20	20	20

Наиболее сложным объектом информатизации является кредитная организация, при этом среди особенностей ее плана счетов можно выделить следующие:

- балансовые счета (раздел А);
 - синтетические счета двух видов – включают счета первого и второго порядка;
 - группировка счетов по экономическому содержанию, типу финансово-хозяйственных операций, по назначению, срочности и ликвидности;
- счета доверительного управления (раздел Б);
 - если кредитная организация (банк) выступает доверительным управляющим, то в ее распоряжение поступает обособленное имущество для управления которым необходимо составлять самостоятельный баланс;
- внебалансовые счета (раздел В);
 - учитывают ценности и документы, принятые на хранение, инкассо, комиссию, а также состав и движение источников финансирования капитальных вложений, бланки строгой отчетности, бланки акций и прочее;
 - ценности и документы, которые ранее были отражены на балансовых счетах по внебалансовым счетам в последующем не отражаются;
 - принцип двойной записи относится к синтетическим счетам первого и второго порядка: активные счета корреспондируют со счетами 999.99, а пассивные операции корреспондируют со вспомогательным счетом 999.98;
 - аналитический учет организуется на лицевых счетах в имеющейся картотеке;
- срочные счета (раздел Г);
 - операции по сделкам купли-продажи денежных средств, иностранной валюты, драгоценных металлов, ценных бумаг, по которым дата заключения сделки не совпадает с датой расчета составляется отдельный баланс;
 - являются счетами-мемориалами, которые устанавливают объем заключенных сделок до наступления срока актуальности определенного платежа, а затем их данные для завершения сделки переносятся на балансовые счета, где отражается фактическое движение финансовых активов и инструментов;
 - позволяют обеспечить проведение сделок в наличной и безналичной форме, срочных сделок и реализовать учет нереализованных курсовых разниц;
- счета ДЕПО (раздел Д);
 - учитываются операции с эмиссионными ценными бумагами (акции и облигации), которые приняты на ответственное хранение от разных клиентов или для осуществления доверительного управления или других целей;
- отсутствуют парные по наименованию активно-пассивные счета;
- использование единых счетов для отражения операций как в национальной так и в иностранной валюте, а также отсутствует валютный раздел баланса;
 - в синтетическом учете парные операции отражаются только в рублях;
 - в аналитическом учете в разрезе кодов валют и в рублевом эквиваленте;

- счета подразделяются по типам клиентов с учетом их формы собственности, вида деятельности и типа хозяйствующего субъекта или контрагента;
 - резидент – ЮЛ находящиеся на территории определенной страны и представляющие интересы государства, прочие ЮЛ: финансовые, коммерческие, некоммерческие организации, а также частные предприниматели и ФЛ;
 - нерезиденты – прочие субъекты рынка ЮЛ, кредитные организации и ФЛ.

Для программной реализации различных методов ведения и поддержки бухгалтерского учета используются разные инструкции и следующие признаки счетов:

- тип сальдо для балансовых счетов (начальное и конечное сальдо по активному и пассивному счетам бухгалтерского баланса из плана счетов);
- периодичность использования счета и алгоритм закрытия определенного счета;
- периодичность формирования отчетности – ежемесячно, ежеквартально, раз в год;
- реализация валютного учета по балансовым счетам бухгалтерского баланса;
- структура кода счета – формируется для использования при помощи табл. 6.3;
- принадлежность к группе, подгруппе счетов или разделу согласно плана счетов.

Компьютерные системы бухгалтерского учета позволяют использовать различные рабочие планы счетов согласно виду деятельности, при этом программным способом устанавливаются наименования и соответствия между разными счетами плана счетов.

В бухгалтерские проводки включаются признак «Вид счета» (активный или пассивный), который соответствует определенным моделям бухгалтерского учета.

Бухгалтерский учет обеспечивает ведение синтетического и аналитического учета.

Определение

Синтетический учет – это обобщение данных об финансовом состоянии и имуществе, обязательствах и хозяйственных операциях в стоимостном выражении.

Детализация объектов учета на синтетических счетах осуществляется путем открытия субсчетов к ним: имеются в виду субсчета второго порядка (две цифры).

Субсчета обеспечивают возможность дополнительной классификации и анализа всех операций: вводится иерархия субсчетов для каждого синтетического счета.

Определение

Аналитический учет ведется на лицевых, материально-ордерных, личных и иных аналитических счетах бухгалтерского учета в ручном варианте или посредством средств вычислительной техники, а также обеспечивает группировку информации внутри каждого синтетического счета в стоимостном или в натуральном выражении.

Совокупность имеющихся значений по всем счетам аналитического учета позволяет сформировать сумму, которая записывается в синтетическом счете второго порядка, а совокупность сумм по всем синтетическим счетам второго порядка позволяет сформировать сумму синтетического счета первого порядка, в свою очередь, совокупность сумм по синтетическим счетам первого порядка позволяет сформировать итог по соответствующему разделу бухгалтерского баланса.

Объектами и срезами аналитического учета могут выступать ряд сущностей:

- основные средства производства: здания, сооружения, машины, оборудование;
- приборы и материалы, товары, нематериальные активы, малоценные и быстроизнашивающиеся предметы, производственные или внешние заказы;
- ЮЛ выступают поставщиками, покупателями, клиентами, заказчиками, дебиторами, кредиторами, с которыми систематически ведутся взаимные расчеты;
- материально-ответственные и подотчетные лица в пределах организации;
- акционеры как номинальные держатели ценных бумаг и учредители как совокупность владельцев организации (держатели контрольного пакета акций);
- документы – основания подтверждают факт финансово-хозяйственной операции;
- элементы организационной структуры – бизнес-единицы: отделы и подразделения;
- статьи расходов и элементы затрат, а также издержки обращения финансовых инструментов на рынке – комиссионное вознаграждение при их обращении.

Идентификатор счета определяет способ обработки учетной информации.

Структура данных отражающая кодификатор определенного счета включает:

- код плана счетов – предназначен для систематизации и поиска документа;
- раздел плана счетов – характеризуется наименованием и идентификатором, позволяет отразить совокупность операций с финансовыми инструментами и имуществом определенного вида, которые относятся к активу или пассиву;
- тип сальдо счета – отражает способ расчета конечного остатка с учетом начального сальдо предыдущего отчетного периода и оборотов по дебету и кредиту в текущем отчетном периоде на активных и(или) пассивных счетах;
- признак валютного учета – регламентирует наличие операции, которая имеет денежное выражение в валюте иностранного государства (EUR, USD, JPE);
- классификационный код счета содержит несколько идентификаторов;
 - идентификатор синтетического счета согласно плану счетов;
 - идентификатор синтетического субсчета 1-го уровня (3 цифры);
 - идентификатор синтетического субсчета 2-го уровня (2 цифры);
 - идентификатор аналитического счета согласно коду введенному бухгалтером.

В структуре кода счета можно отразить всю необходимую информацию, а его разрядность достаточно большая: в компьютерных программах ограничена 255 символами, чего вполне достаточно для представления информации отражающей признак счета. Компьютерные программы бухгалтерского учета позволяют использовать структуру кода для отбора и группировки информации разного типа и назначения.

В компьютерных системах бухгалтерского учета аналитическим счетам соответствуют справочники, картотеки, реестры документов, учетные регистры, а вся учетная информация о выполненных хозяйственных операциях представлена в БД в виде бухгалтерских проводок. Для каждого синтетического счета и субсчета, а также аналитического счета вводится начальное сальдо, которое выступает суммой имеющейся на начало первого учетного периода, а затем рассчитывается конечное сальдо.

Основу компьютерных систем бухгалтерского учета составляют классификаторы и кодификаторы технико-экономической информации разного рода и назначения.

Определение

Система классификации – совокупность правил и результат распределения заданного множества объектов на подмножества в соответствии с признаками сходства или различия посредством принципа наибольшего правдоподобия.

Определение

Кодификаторы – используются для кодирования и простоты поиска элемента множества, который отражает счет, номенклатурную единицу, документ и прочее.

Кодификатор используется в основе плана счетов и кодирует элементы плана счетов: синтетические счета первого и второго порядка, а также аналитические счета.

Выделяют различные методы классификации учетной информации (рис. 6.11):

- **иерархический метод** – между классификационными группами устанавливаются отношения подчинения, последовательной детализации свойств или типа: «базовый класс – подкласс – группа – подгруппа – вид – подвид – элемент» и т.п., при этом в иерархической классификации каждый объект попадает только в одну классификационную группу, а объединение группы одного иерархического уровня дает определенное множество объектов, что определяет глубину иерархии посредством разных классификационных признаков;
- **фасетный метод** – исходное множество объектов разбивается на подмножества в соответствии со значениями отдельных взаимно независимых фасетов, которыми выступают наборы значений каждого признака классификации, допускается, что один и тот же объект может входить в различные фасеты.

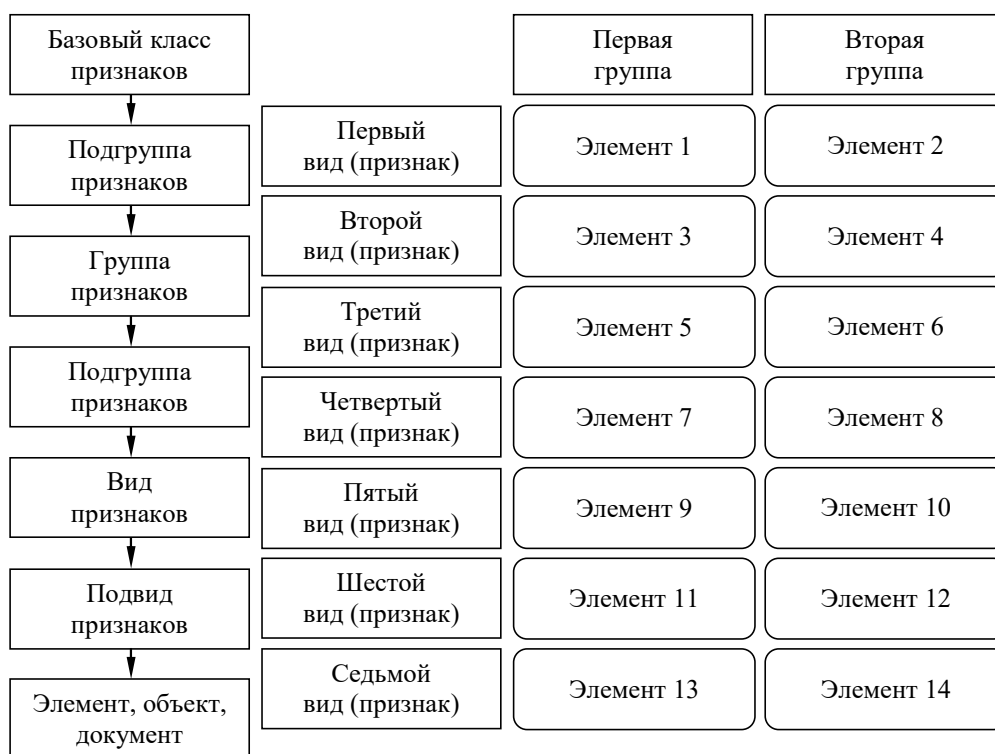


Рисунок 6.11. Иерархический и фасетный метод классификации

Допустимы несколько способов кодирования идентификаторов счетов соответствующих различным объектам учета, которые используются в ходе финансово-хозяйственной деятельности организационной структуры предприятия или организации:

- последовательное – иерархическая схема (примером выступает план счетов);
- параллельное – фасетная схема, обеспечивающая присвоение объектам учета или классификационным группам условных обозначений из двух частей.

Характеристиками любого кода используемого при классификации выступают:

- используемый алфавит – ограниченное счетное множество символов кода для представления (кодирования) исходного набора символов (данных), в качестве которых выступают цифры, буквы, символы, штрихи, цвета и прочее;
- длина кодовой последовательности и система обозначений, которая используется при кодировании совокупности объектов деятельности организации;
- метод кодирования – задает систему правил и способов представления (кодирования) различных объектов, которые используются в бухгалтерском учете:
 - классификационный (иерархический) – обеспечивает формирование составного кода, каждая часть которого разделяется точкой, используется при кодировании счета в плане счетов соответствующего объекту учета;
 - регистрационный (идентификационный) – задает однозначное соответствие между наименованием объекта учета и его кодом, предназначен для повышения секретности при работе пользователя с финансовой информацией.

К кодам экономической информации предъявляется особый ряд требований:

- структура кода включающая минимально необходимое количество элементов с учетом возможности расширения множества кодируемых объектов;
- учет специфики программных и технических средств обработки данных;
- высокая помехозащищенность кода и возможность использования условных обозначений (символы, геометрические фигуры различной формы и цвета).

Классификаторы и кодификаторы имеют различные сферы действия:

- локальные (внутрисистемные) классификаторы – используются только в рамках ИС определенного кредитного учреждения или предприятия (организации);
- отраслевые классификаторы – применяются во всех ИС, которые функционируют в определенной отрасли народного хозяйства или производства;
- региональные, республиканские, городские и областные классификаторы – используются для всех ИС в определенном географическом регионе (области);
- общероссийские классификаторы – используются в ИС организаций и обеспечивают расчеты и формирование отчетов на территории РФ или субъектов РФ;
- международные классификаторы – применяются для стандартизации ИС организаций, которые оперируют с объектами учета на международном уровне.

Чем выше уровень определенного классификатора, тем более общими являются заложенные в нем признаки объектов и тем шире номенклатура различных объектов, подлежащих учету и контролю посредством методов бухгалтерского учета и аудита.

Компьютерные системы бухгалтерского учета прошли большой исторический путь становления и развития, поскольку они изменялись параллельно с совершенствованием информационных технологий, программных и технических средств автоматизации обработки потоков разнородной информации, методов и средств разработки, концепций построения информационных систем разного профиля и назначения.

В связи с возрастающими общественными потребностями рынок компьютерных систем бухгалтерского учета начал формироваться примерно с конца 80-х г.г. Сегодня существует большое число разных программных средств автоматизации бухгалтерского учета: от локальных программных средств бухгалтерского учета до полнофункциональных компьютерных систем бухгалтерского учета в составе ИС крупного вертикально и горизонтально интегрированного предприятия (организации).

При рассмотрении определенного предприятия любого размера как объекта информатизации можно выделить большое количество признаков, которые определяют направления внедрения средств автоматизации финансового и бухгалтерского учета:

1. Основные характеристики объекта информатизации и системы управления, которые влияют на особенности автоматизации процесса бухгалтерского учета.
 - отраслевая специфика бухгалтерского учета, которая находит отражение в моделях и алгоритмах учета, элементах учетной политики, поскольку существуют законченные отраслевые решения как для отдельных учетных функций, так и для ИС предприятия занимающегося определенной деятельностью;
 - для предприятий находящихся в одной отрасли производства существует специфика видов деятельности, технологических процессов производства продукции, выполнения работ и оказания услуг, поэтому существуют типовые схемы движения информационных потоков в пределах его подразделений и отделов;
 - при разработке инфологических схем БД и структур данных в распределенных хранилищах информации для компьютерных систем бухгалтерского учета крупного предприятия необходимо учитывать номенклатуру используемых материалов, малоценных и быстроизнашивающихся предметов, набор разных видов и групп технологического оборудования, профессий рабочих и служащих, контрагентов хозяйственных операций (поставщиков и покупателей);
 - интенсивность информационных сообщений, состав используемых систем классификации и кодирования важнейших видов информации, формы документов существенно зависят от видов и масштабов деятельности предприятия;
 - тип (ручное, автоматизированное) и характер (единичное, серийное, массовое) производства оказывает существенное влияние на состав нормативно-справочной информации в БД, формы первичных документов, порядок и периодичность учета затрат и результатов производственного процесса (деятельности);
 - организационная структура управления предприятием, число и территориальное распределение подразделений, наличие централизованной бухгалтерии, бухгалтерии обособленных подразделений определяет требования к ИТ сбора, передачи и хранения данных, конфигурации программных систем.

2. Общая характеристика компьютерных систем бухгалтерского учета, которые используются для автоматизации и интенсификации при реализации задач учета:

- созданная и принятая модель бухгалтерского учета и учетной политики, соответствие международным стандартам финансового и бухгалтерского учета;
- возможность адаптации системы: выбор и конфигурирование функций, состава и структуры данных в БД, ввод форм входных и выходных документов;
- структура разработанного рабочего плана счетов, структура и элементы кода счета, глубина аналитического учета, виды первичных учетных регистров: бухгалтерских проводок, хозяйственных операций и первичных документов;
- типовая нормативно-справочная база системы, состав типовых хозяйственных операций, отражаемых посредством принятой модели бухгалтерского учета;
- инструментальные средства создания и развития – экранных и печатных форм первичных документов, форм внешней и внутренней бухгалтерской отчетности, формы управленческой и статистической отчетности, а также интерфейса программного обеспечения для разных категорий пользователей;
- связь с другими системами масштаба предприятия и внешними информационными системами предполагает использование общих форматов обмена данными;
- методология и технология проектирования, внедрения, практического использования и сопровождения разных средств автоматизации задач учета;
- необходимость обеспечения совместимости аппаратной и программной платформы с информационной и технологической архитектурой системы;
- инструментальные средства разработки приложений для модернизации;
- программные средства для администрирования БД и программной системы;
- требования к квалификации персонала, который эксплуатирует систему;
- стоимость и затраты на реализацию проекта по созданию системы учета.

Анализ различных систем показал, что существуют стандартные подходы к автоматизации учета хозяйственных операций осуществляемых в организации:

- непосредственный ввод учетных данных в регистр (журнал, книгу) хозяйственных операций в виде проводок посредством встроенных справочников;
- контроль правильности вводимых бухгалтерских проводок с помощью заранее подготовленного списка типовых шаблонов «корректных проводок»;
- фильтрация списка бухгалтерских проводок в учетном регистре с целью их выборки, редактирования, копирования, сортировки по дате, документам;
- ввод хозяйственных операций, которые содержат шаблоны бухгалтерских проводок, которые открыты для настроек, ведения регистра (журнала) хозяйственных операций, формируемых на основании типовых операций;
- автоматическое заполнение регистра бухгалтерских проводок посредством использования справочников содержащих единицы измерения и стоимость;
- работа с шаблонами типовых первичных документов, создание регистра (журнала) документов, которые отражают набор хозяйственных операций.

В отличие от классических методов и правил ведения бухгалтерского учета первичные документы соответствующие некоторым хозяйственным операциям можно оформлять до фактического совершения операции, а затем проводить их.

Использование систем электронного документооборота позволяет существенно повысить эффективность управления организационной структурой предприятия.

Если в системе ведется валютный учет, шаблон проводки содержит реквизиты кода валюты, валютного курса, на основании которых выполняется перерасчет валютной суммы проводки и хозяйственной операции. При этом в системе должен вестись международный справочник, который содержит виды валют и курсы валют.

В системе обеспечивается возможность независимого ведения курсов валют в справочнике курсов валют и оформляемых документах или хозяйственных операциях (Парус и 1С). При этом в этих системах автоматически ведется международный справочник видов валют, курсов валют, обновляемый непосредственно через порталы банков и бирж Англии, США, Франции, России посредством сети Интернет.

Аналитические и стратегические исследования и мероприятия показали, что для малого предприятия самый оптимальный вариант – несетевая, централизованная БД; для среднего – сетевая, файл-сервер, централизованная БД или сетевая, клиент-сервер, централизованная БД; для больших – сетевая, клиент-сервер, распределенная БД в зависимости от уровня интеграции вычислительной системы и организации.

Раз уж было упомянуто об уровне интеграции, то системы бухучета различаются по полноте учетных функций и широте охвата инфраструктуры предприятия:

- системы для автоматизации отдельных участков бухгалтерского учета;
- комплексные системы для автоматизации всех участков бухгалтерского учета;
- комплексные системы бухгалтерского учета с расширением функций бухгалтерского учета, например, торговые системы, складские системы, системы управления продажами, системы автоматизации торгово-закупочной деятельности;
- системы с возможностью модернизации платформы программного средства и конфигурации позволяющей реализовать функции управления предприятием.

Существующие системы позволяют реализовать широкий набор функций и задач при автоматизации бухгалтерского учета и финансового анализа на предприятии:

- разработка, распространение и продажа готовых программных продуктов реализующих системы для автоматизации документооборота на предприятии;
- консалтинговые услуги по проектированию информационной среды, разработке модели бухгалтерского учета, внедрению и сопровождению программных систем, выбору программных продуктов для создания ИС предприятия;
- создание информационно-правовых систем для расширения функций учета;
- создание и распространение справочной и методической литературы для эффективного использования систем пользователями с разной подготовкой;
- организация ряда учебных центров подготовки обслуживающего персонала;
- создание информационных ресурсов с методическим обеспечением системы.

В ходе проведенного анализа удалось представить рейтинг современных программных средств, обеспечивающих автоматизацию бухгалтерского учета (табл. 6.4).

Таблица 6.4

Рейтинг современных программных систем автоматизации бухгалтерского учета

Место (позиция)	Название фирмы-изготовителя	Рейтинг
1	1С: Предприятие	91
2	Интеллект-сервис: БЭСТ	78
3	Парус	77
4	Галактика	75
5	Диасофт	72
6	R-Style software lab	70
7	SAP/R3	66
8	Инфин	63
9	Инфософт	60
10	Омега	58

Исходя из отбора программ по рейтингу бесспорным лидером среди разработчиков является фирма 1С, продукты которой впоследствии стали бестселлером.

Продукция 1С обладает широким набором функций и реентерабельностью, пользователи имеют возможность получения информационно-технологического сопровождения, образовательных и консультационных услуг со стороны представителей разветвленной франчайзинговой сети занимающейся распространением и сопровождением программных продуктов и систематических обновлений к ним.

В целом, было установлено, что фирмы-разработчики программ предлагают широкую номенклатуру программных продуктов, учитывающих потребности пользователей. Ряд фирм-разработчиков создает программные продукты единой серии под общей торговой маркой. Они предназначены для предприятий различных масштабов и разных специалистов, которые заняты постановкой учета на основе типовых элементов учетной политики и используют типовые программные решения.

Фирма «Парус» предлагает ряд программных решений с разными функциями:

- «Парус "Предприятие"» – полнофункциональная система автоматизации бухгалтерского учета и управления на предприятии малого и среднего размера;
- «Парус "Система управления"» – система автоматизации управления для промышленных предприятий, выпускающих широкую номенклатуру продукции;
- «Парус "Аналитика"» – аналитическая программа для анализа результатов финансово-хозяйственной деятельности торгово-закупочных предприятий;
- «Парус "Бюджет"» – система автоматизации управления учетом для бюджетных организаций, которые осуществляют разные виды деятельности;
- «Парус "Страхование"» – система автоматизации управления для страховых компаний, которые осуществляют страхование, сострахование и перестрахование.

Фирма «Интеллект-Сервис» предлагает ряд основных программных продуктов:

- БЭСТ-4 – полнофункциональная, многопользовательская и много-валютная система оперативного (торгового, складского) и бухгалтерского учета, ввод данных в которую осуществляется на основе первичных документов, она хорошо адаптируется и вливается в специфику конкретного предприятия;
- БЭСТ-4 «Магазин» – для автоматизации предприятий и сетей розничной торговли, обеспечивает управление товарооборотом от поступления товаров на складе до их реализации, поддерживает оперативный и бухгалтерский учет, взаимодействие с торговым оборудованием (касса, весы, штрих-сканер);
- БЭСТ «Анализ» – программа анализа товарооборота, закупок и цен товаров за различные периоды времени, маржинальной прибыли по различным видам товаров, поддерживается обмен информацией с БД других модулей;
- БЭСТ «Компания» – автоматизация управления крупным предприятием торговли направлена на оперативное управление торговыми потоками, включая учет закупок, запасов, продаж товаров, расчетов по обязательствам, автоматизированы все функции бухгалтерского и налогового учета, а для управления БД используются крупномасштабные СУБД типа MS SQL-Server;
- БЭСТ «Маркетинг» – программа автоматизации маркетинговых исследований, поддерживает функции: определение типа конкурентной среды и целевых сегментов рынка, выработка рекомендаций по проведению рекламных компаний, поддержке и стимулированию сбыта, финансовое планирование, формирование продаж, расчет бюджета проводимой рекламной компании;
- БЭСТ «Офис» – полнофункциональная система управления малым и средним предприятием, обеспечивает планирование движения денежных средств, доходов и расходов, учет и анализ особенностей хозяйственной деятельности;
- БЭСТ «План» – программа формирования календарных сбытовых и производственных планов для предприятий торговли, производства и сферы услуг, расчета издержек и себестоимости продукции и услуг, анализа эффективности вариантов планов, прогноза результатов работы на основе отчетных данных;
- БЭСТ-ПРО – комплексная автоматизация системы управления предприятием производственного типа, торговли или сферы услуг, обеспечивающая полный управленческий цикл, включая ведение договоров, контроль взаиморасчетов, планирование производства и сбыта, учет затрат на производство продукции, выполнение работ и услуг, управление снабжением, а также расчет зарплаты.

Продукты компании «Интеллект-Сервис» обеспечивают возможность автоматизации ведения бухгалтерского, финансового и управленческого учета в организациях малого, среднего и большого размера, занимающихся выпуском широкого ассортимента и номенклатуры продуктов, работ и услуг ориентированных на удовлетворение потребностей дифференцированного контингента потребителей.

Фирма R-Style Software Lab специализируется в основном на создании разных средств автоматизации кредитных организаций, в частности банков и депозитариев:

- RS-Bank – позволяет создать автоматизированную компьютерную систему, которая обеспечивает деятельность кредитных учреждений любого масштаба и с любым методом реализации управления организационной структурой;
- RS-Dealing – система для управления финансовыми ресурсами банков;
- RS-Loans – система автоматизации обслуживания и кредитования ФЛ;
- RS-Retail – программный комплекс для автоматизации банковских услуг;
- RS-Balance – автоматизация бухгалтерского и финансового учета результатов хозяйственной деятельности предприятий малого и среднего размера.

Программный комплекс 1С:Предприятие используется в нескольких режимах:

- конфигуратор – создание новой или настройка типовой конфигурации 1С, которая обеспечивает автоматизацию ведения бухгалтерского учета в фирме;
- рабочий режим – обеспечивает выполнение штатных учетных функций, расчетов и формирование отчетности в процессе функционирования системы;
- отладчик – отладка и поиск ошибок в программных модулях, которые написаны на встроенном языке программирования используемом в 1С:Предприятие;
- монитор пользователей – оперативный анализ процесса работы пользователей всех имеющихся категорий в сетевой версии программы 1С:Предприятие.

Каждая категория пользователей работает в определенном режиме программы:

- режим конфигуратора предназначен для работы программистов в системе, осуществляющих настройку, внедрение и сопровождение программы;
- режим отладчика используется программистами, которые занимаются модификацией существующих или созданием новых программных модулей;
- рабочий режим обеспечивает эксплуатацию программы бухгалтерами, менеджерами склада, работниками отдела кадров, кассирами и товароведом;
- режим монитора пользователей используется администратором системы для анализа активности пользователей разных категорий и уровня подготовки.

Развитие программы осуществлялось в рамках двух основных направлений:

- совершенствование алгоритмического обеспечения и инструментальных средств для разработки программного обеспечения, реализующего базовую платформу 1С: Предприятие, обеспечивающую запуск разных конфигураций;
- разработка типовых конфигураций для автоматизации определенных сегментов ведения бухгалтерского учета в различных предметных областях.

Со временем сложилось сообщество пользователей программы «1С: Предприятие», которое насчитывало сотни тысяч человек. Появилась развитая франчайзинговая сеть обеспечивающая распространение и сопровождение базовых программных продуктов и типовых конфигураций к ним, а также занимающаяся обучением и консультированием пользователей с разным уровнем подготовки и родом занятий.

Рассмотрим имеющиеся конфигурации программы «1С:Предприятие»:

- «1С: Предприятие 7.7 с поддержкой MS SQL Server» – расширение системы для реализации работы с большими БД в архитектуре «клиент-сервер»;
- «1С:Аспект» – автоматизация торгового учета на предприятиях оптовой и мелкооптовой торговли, в т.ч. с использованием упрощенной схемы учета;
- «1С:АФСП» – анализ финансового состояния предприятия на основе накопленных и текущих данных в рамках стандартной бухгалтерской отчетности;
- «1С: Правовые базы данных» – юридические БД на разных информационных носителях CD-ROM и DVD-ROM (1С:Гарант, 1С:Кодекс, 1С:Эталон, 1С:Аспект);
- «1С: Бухгалтерия» – универсальная система для ведения учета любой сложности на предприятиях разных видов деятельности и форм собственности;
- «1С: Строительство» – типовое решение для крупных строительных организаций, обеспечивающее управление своими активами и обязательствами;
- «1С: Зарплата и кадры» – обеспечивает автоматизацию процесса расчета размера зарплаты и оптимизацию ведения кадрового учета на предприятии;
- «1С: Налогоплательщик» – программа подготовки данных на носителях информации на магнитных и оптических дисках для передачи в государственные налоговые инспекции согласно закону «О подоходном налоге с ФЛ»;
- «1С: Основные средства» – автоматизация управления основными средствами: зданиями, сооружениями, оборудованием (сборочными линиями и конвейерами);
- «1С: Платежные документы» – позволяет реализовать автоматизацию документооборота платежных поручений и требований на разных предприятиях;
- «1С: Предприятие: Комплексная конфигурация» – обеспечивает реализацию автоматизации бухгалтерского учета и аудита, расчета размера зарплаты сотрудников, кадрового учета, торгово-закупочной деятельности и складского учета;
- «1С: Бухгалтерия для бюджетных организаций» – позволяет реализовать автоматизацию учета и контроля за расходованием средств, подготовить финансовую отчетность для налоговых органов и Федерального Казначейства;
- «1С: Войсковая часть» – реализует повышение эффективности бухгалтерского учета в учреждениях подведомственных Министерству обороны РФ;
- «1С: Торговля+Склад» – автоматизирует процесс осуществления торгово-закупочной деятельности и складского учета на профильном предприятии.

Деятельность компании 1С является диверсифицированной поскольку наряду с разработкой, распространением и сопровождением комплексных систем бухгалтерского учета она занимается созданием прикладного программного обеспечения ориентированного на широкий круг пользователей с разным уровнем подготовки:

- игры, симуляторы, тренажеры и обучающие программы (электронные учебники);
- энциклопедии, словари, справочники, рубрикаторы, буклеты и брошюры;
- системы для обеспечения возможности автоматизированного перевода произвольного текста с иностранного на национальный язык, а также наоборот.

Остановимся на конфигурации «1С:Предприятие 8: Управление торговлей» и «1С:Предприятие 8: Управление персоналом», которая обеспечивает автоматизацию выполнения торгово-закупочной деятельности и управления персоналом:

1. Управленческий учет – для контроля и анализа финансово-хозяйственной деятельности:
 - постановка и ведение финансового учета в масштабе крупного предприятия;
 - контроль остатков и резервов сырья и товаров на складах готовой продукции;
 - мультивалютный учет разных товаров на складах и в производственном цикле;
 - мультивалютный учет взаиморасчетов с покупателями и поставщиками товаров;
 - мультивалютный учет наличных средств в кассе и на расчетном счете фирмы;
 - формирование сводных финансовых и управленческих отчетов для управления.
2. Финансовый учет – для реализации планирования статей доходов и расходов:
 - отдельный финансовый учет по каждой фирме или определенному контрагенту;
 - автоматическое формирование бухгалтерских проводок для типовых операций и типовых форм документов, которые используются в организации;
 - учет НДС и налога с продаж, а также формирование книги закупок и продаж;
 - стоимостной анализ и учет хозяйственных операций в рублях и валюте;
 - формирование бухгалтерской отчетности: баланс, отчет о прибылях и убытках.

Если непосредственно рассмотреть диалоговое окно Конфигуратора программы 1С:Предприятие, то допустимо открытие вкладки меню «Конфигурация» – «Открыть конфигурацию», которое содержит панель меню из нескольких элементов:

- метаданные – создание и настройка объектов, а также данных конфигурации;
- интерфейсы – создание пользовательских интерфейсов для работы разных пользователей в программе 1С:Предприятие в режиме обычной работы;
- пользователи – определение перечня объектов и разграничение прав доступа пользователей разных категорий для обеспечения работы с конфигурацией.

Для обеспечения ввода перечня объектов и их сведений в справочники (номенклаторы), а также реализации использования различных форм документов необходимо осуществить конфигурирование объектов представленных в иерархическом дереве окна программы 1С: Предприятие запущенной в режиме «Конфигуратор».

Для облегчения работы пользователей разрабатываются разные интерфейсы, учитывающие специфику работы пользователей разного уровня подготовки которые выполняют определенный набор функций согласно должностным инструкциям.

Для обеспечения разграничения прав доступа к информационной БД программы 1С: Предприятие обеспечивается ввод перечня параметров пользователей и устанавливаются спецификаторы доступа на отдельные объекты конфигурации.

Непосредственное конфигурирование компьютерных систем бухучета на базе 1С: Предприятие состоит в адаптации модели бухгалтерского и оперативного учета посредством настройки имеющейся типовой конфигурации на основе особенностей ведения учета в организации, перечня объектов учета и отражающих их сведений.

При конфигурировании информационной базы необходимо учитывать:

- особенности организационной структуры и методы управления предприятием в терминах 1С:Предприятие: фирма, контрагент, подразделение, пользователь;
- определение бизнес-процессов: учет кадров, складской учет, торговля, посредническая деятельность, услуги, соответствующие системе управления;
- определение задач всех подразделений для реализации учета в программе 1С:Предприятие и их распределение по разным исполнителям – пользователям;
- настройка параметров типовой конфигурации в соответствии с используемой в организации учетной политикой и принятой методологией и моделью учета;
- конфигурирование форм первичной отчетности и организация документооборота;
- создание номенклаторов и справочников, классификаторов и кодификаторов экономической информации для реализации аналитического учета объектов;
- внедрение и модификация типовых форм и способов представления бухгалтерской, статической, оперативной отчетности принятой в организации;
- разработка структур данных и модификация значений свойств характеризующих объекты учета в иерархическом дереве при работе с конфигурацией;
- разработка и модификация процедур обработки данных, а также событий инициированных нажатием кнопок расположенных на различных формах программы 1С:Предприятие согласно требованиям указанным в описании;
- разграничение прав доступа к элементам информационной базы при работе пользователей разных категорий, включая администраторов и сотрудников;
- обеспечение возможностей архивирования и резервного копирования данных пользователей и метаданных конфигурации программы 1С: Предприятие;
- обеспечение систематического получения обновлений каждой конфигурации;
- реализация переноса данных пользователя при переходе на новую версию.

Важно отметить, что 1С:Предприятие использует технологию объектно-ориентированного программирования при использовании внутреннего языка для модификации процедур и компонентов типовой конфигурации, которая используется на предприятии. Встроенный язык программирования позволяет быстро создавать, модифицировать и модернизировать различные программные модули.

Конфигуратор оперирует с классами объектов метаданных: константы, справочники, документы, журналы, отчеты и т.п. Каждый объект метаданных характеризуется набором свойств (параметров) и методов обработки данных (процедур).

Процесс конфигурирования включает несколько существенных элементов:

- выбор состава объектов метаданных и на ввод в режиме «Конфигуратор»;
- создание новых объектов метаданных на основе имеющихся классов и типов;
- уточнение свойств всех объектов метаданных, которые имеются в иерархии;
- уточнение методов и процедур обработки свойств объектов метаданных;
- создание новых методов и процедур обработки всех свойств объектов метаданных, которые непосредственно используются в определенной конфигурации.

Аудиторская деятельность направлена на проведения экономического анализа и контроля уставной деятельности организации, установление достоверности имеющейся финансовой отчетности и проверку соответствия зарегистрированных в финансовой отчетности и фактически совершенных операций согласно действующему налоговому и административному законодательству на территории государства.

Аудиторская деятельность профессиональных аудиторов дифференцируется:

1. По способу проведения аудиторской проверки определенной организации:
 - обязательный – проводится согласно требованиям государственных органов;
 - инициативный – иницируется руководством организации для получения независимого экспертного заключения о соответствии сведений представленных в финансовой отчетности фактическому состоянию объектов учета.
2. По особенностям проведения аудиторской проверки состояния организации:
 - камеральная – финансовая отчетность доставляется представителями предприятия в офис организации осуществляющей аудиторскую деятельность;
 - выездная – проводится силами внешней аудиторской организации или уполномоченных государственных органов непосредственно в организации.
3. По характеру проведения анализа экономического состояния и полученных результатов финансово-хозяйственной деятельности за отчетный период:
 - внутренний – осуществляется отдельным подразделением организации;
 - внешний – реализуется независимыми аудиторами и ассоциациями аудиторов.
4. По периодичности проведения аудиторской проверки в организации выделяют:
 - первичный – по первому обращению или для вновь созданной организации;
 - повторный – реализуется систематически или повторно при потребности.

Нормативной документацией регламентирующей внешний и внутренний контроль и аудиторскую деятельность выступают действующие положения об организации внутреннего контроля в кредитной организации и на промышленном предприятии.

Основные требования к постановке аудиторской деятельности следующие:

- организационная структура формируется в зависимости от объема и содержания проводимых операций, методов их контроля, числа и квалификации персонала обеспечивающего выполнение различных функций в организации;
- соответствие перечня должностных инструкций и набора операций обеспечивающих модификацию информационной базы программы «1С: Предприятие» при осуществлении проведения документов определенного типа;
- высокий уровень профессиональной работы сотрудников и соответствие должностным требованиям, которые установлены в данной организации;
- независимость службы внутреннего аудита и контроля от других подразделений и отделов организации, наделение ее исключительными полномочиями;
- систематическое отражение результатов финансово-хозяйственной деятельности организации посредством использования различных форм отчетности.

Типовая последовательность этапов аудиторской проверки организации:

1. Предварительная оценка возможности принятия заказа на проведение аудита:
 - предварительная оценка степени риска проведения аудиторской проверки;
 - определение целей, задач и коллектива аудиторов осуществляющих проверку;
 - определение условий, обязательств, пожеланий и требований клиента.
2. Оформление заказа, разработка плана и программы аудиторской проверки:
 - изучение специфических особенностей информационной среды организации и разработанных в ней методов и инструкций ведения документооборота;
 - изучение сфер влияния и зон ответственности службы внутреннего контроля;
 - изучение особенностей бухгалтерского учета на основе принятой модели;
 - предварительные аналитические процедуры по секторам ответственности;
 - планирование направлений и результатов аудиторской проверки организации;
 - подготовка и обсуждение с клиентом плана проведения внешнего аудита;
 - формирование предварительного плана для проведения внешнего аудита.
3. Проведение аудиторской проверки на территории организации-заказчика:
 - анализ сегментированной отчетности предоставленной службой внутреннего аудита;
 - анализ предыдущих заключений, которые выданы внешними аудиторами;
 - изучение форм финансовой отчетности и проверка соответствия зарегистрированных и фактически выполненных финансово-хозяйственных операций;
 - определение направлений и степени существенности искажений исходя из выявленного перечня ошибок в предоставленной финансовой отчетности;
 - формализация полученных результатов и корректировка плана аудиторской проверки для расширения перечня направлений или углубления анализа.
4. Реализация плана аудита и подготовка отчета по результатам проведения проверки:
 - осуществление процедур контроля и проверки полученных результатов;
 - проведение аналитических процедур и оценка полученных результатов;
 - обзорная проверка содержания финансовой отчетности формально и по существу.
5. Подготовка заключительного отчета по результатам аудиторской проверки:
 - проведение дополнительных обзорных процедур в ходе аудиторской проверки;
 - проведение официальных встреч с руководством проверяемой организации;
 - подготовка обобщающего меморандума по результатам аудиторской проверки;
 - аудиторское заключение, которое содержит выявленные нарушения и результаты (безусловно отрицательное, положительное с оговорками, положительное).
6. Обсуждение аудиторского заключения о фактическом состоянии организации:
 - подведение итогов аудиторской проверки и выдача письменного заключения;
 - согласование с клиентов степени существенности выявленных замечаний;
 - определение возможных методов и технологий устранения выявленных ошибок;
 - выдача основных рекомендаций по совершенствованию принятой модели организации финансового и бухгалтерского учета на данном предприятии.

Информационные элементы в содержании аудиторского заключения:

- вводная часть содержит общие сведения об аудиторе и состав комиссии;
- аналитическая часть (основная часть) общие сведения об организации;
- итоговая часть содержит мнение аудитора о достоверности годового отчета:
 - положительное аудиторское заключение, явно выраженных нарушений нет;
 - положительное заключение с оговорками, незначительные нарушения;
 - отрицательное аудиторское заключение, имеются существенные нарушения;
 - отказ от выражения мнения о достоверности финансовой отчетности.

При рассмотрении кредитной организации состав и основные направления аудиторской деятельности включает ряд секторов, которые подлежат проверке:

- аудит внутренней документации, которая используется в организации;
- аудит активных операций, характеризующих направления размещения средств;
- аудит пассивных операций, отражающих источники привлечения средств;
- аудит последовательности формирования финансовых результатов и корректности схемы подготовки финансовой отчетности в налоговый орган;
- аудит валютных операций исчисляемых в национальной и иностранной валюте;
- аудит выполнения правил и нормативов обязательного резервирования активов.

В процессе осуществления аудиторской проверки на предприятии возникает необходимость распределения задач аудиторов каждый из которых специализируется на определенных сегментах и объектах бухгалтерского учета и отчетности.

В процессе выездной или камеральной проверки реализуется распределенный анализ разных направлений деятельности организации, при этом каждый из аудиторов осуществляет сбор и накопление сведений характеризующих состояние имеющихся объектов учета. После этого осуществляется консолидация собранных данных и выявленных несоответствий, а также формируется мотивированное заключение о причинах отклонений заявленных значений показателей содержащихся в финансовой отчетности относительно их предельно допустимых значений, которые регламентированы действующим законодательством и приняты в организации.

Формируется перечень рекомендаций для руководства предприятия, которые необходимо устранить в установленный аудиторской фирмой срок. Руководство организации формирует задание имеющимся службам и подразделениям для устранения выявленных замечаний и несоответствий, осуществляется сверка и коррекция финансовой и бухгалтерской документации хранящейся в организации.

Непосредственно после устранения выданного перечня замечаний обеспечивается повторный анализ финансовой отчетности и подготавливается проект аудиторского заключения представителями компании осуществляющей аудиторскую проверку. Аудиторское заключение подписывается всеми аудиторами осуществляющими проверку различных сегментов учета в рамках разных направлений деятельности и утверждается руководителем организации занимающейся внешним аудитом.

6.3.3. Программное обеспечение для защиты информации и информационных систем

Защита информации выступает одним из важнейших направлений деятельности государственных органов и ЮЛ, которая связана с комплексом организационных и технико-технологических мероприятий обеспечивающих разграничение прав доступа и предотвращение несанкционированного использования информационных ресурсов, программных комплексов и вычислительных сетей в основе ИС.

Для обеспечения безопасности данных Международная организация стандартизации (ISO) разработала документ, который определяет методы безопасности при работе пользователя, группы пользователей и организации в информационной сети:

- предотвращение возможности чтения сообщений передаваемых по E-Mail;
- защита передаваемых данных от анализа посторонних пользователей;
- использование методов расширенного (криптографического) кодирования;
- обнаружение изменений в разнородных потоках информационных сообщений;
- организационные, аппаратные и программные методы защиты информации;
- определение искажений, появившихся в блоках данных на носителях информации.

С безопасностью данных связано резервное копирование, которое защищает данные и программы от появления ошибок и стирания во время отказов, возникающих в информационной системе или сети. Резервное копирование предполагает систематическое копирование данных содержащихся в разных файлах на носителе.

Существуют три способа реализации резервного копирования: случайный (вероятностный), систематический (через определенные промежутки времени) и многоступенчатый (создается до трех поколений копий резервируемого тома данных).

Конфиденциальность определяет методы и средства обеспечения секретности сведений государственной и предпринимательской деятельности. Задачей конфиденциальности является такая организация работы систем и сетей, при которой доступ к данным и программам разрешается только тем пользователям и программам, которые имеют на это необходимые полномочия и спецификаторы доступа.

Определение

Дополнение данных (внесение избыточности) – заключается в наращивании последовательности информационных элементов в разных местах и происходит за счет добавления символов, которые приносят свойства обнаружения и коррекции ошибок, а в результате данные интерпретируются только ограниченному кругу пользователей посредством использования набора специальных программных средств.

Определение

Криптография – способ изменения исходных и машинных кодов программ, а также данных посредством использования правил кодирования и кодовых таблиц.

Определение

Хеширование – символы исходного текста подвергаются обработке посредством определенной вычислительной криптографической процедуры, которая называется хеш-функцией для реализации кодирования информационных элементов.

Хэш-ключи и хэш-функции для кодирования – выступают правилами, определяющими однозначное соответствие между исходным и кодовым алфавитом при реализации процедуры шифрования в ИС при работе с данными на предприятии:

- открытый – ключ известен ограниченному кругу лиц или устройств;
- закрытый – секретный динамически создаваемый ключ на основе алгоритма.

Существуют два основных способа шифрования данных для передачи в ИС:

- симметричный – кодирование и декодирование данных осуществляются посредством идентичной процедуры с использованием секретного ключа;
- асимметричный – кодирование осуществляется различными процедурами и алгоритмами, при этом допустимо использование двух и более ключей.

Определение

Идентификация – классификация объекта согласно паттерну значений признаков.

Основные виды идентификаторов разграничения доступа и кодирования:

- пароль – последовательность символов, снимающая ограничение доступа;
- реквизиты магнитной или компьютерной (чип) карточки – считывание параметров записанных на магнитной поверхности посредством кард-ридера;
- электронная подпись – способ подтверждения подлинности определенного файла;
- цифровые системы идентификации по биометрическим параметрам человека:
 - идентификация по голосу – специфические особенности голоса человека;
 - отпечатки пальцев – считывание капиллярной сетки на эпителии пальцев;
 - радужная оболочка глаза – сканирование рельефа поверхности сетчатки.

Аутентификация предполагает проведение процедуры идентификации и предполагает установление подлинности пользователя или программного модуля обращающегося к информационному ресурсу предполагающему ограничение доступа.

В современных ИС два основных вида процедуры аутентификации в системе:

- инициированная источником – источник при реализации процедуры информационного обмена выступает первичным инициатором проверки подлинности;
- инициированная потребителем – в начале сеанса обмена информацией потребитель информации осуществляет генерацию запроса о необходимости проверки подлинности сведений об источнике и корректности информации.

Определение

Верификация – процедуры проведения анализа учетной записи (логин и пароль) или данных, которые предназначена для поддержки определения подлинности.

Определение

Проверка целостности – неразрушимая обработка структуры и содержания данных.

Определение

Экранирование – преобразование данных для защиты их целостности.

Проблемы использования ИТ наиболее четко начинают проявляться в процессе информатизации определенной сферы деятельности современного общества.

6.4. Пакеты утилит и сервисные программы

Пакеты утилит и сервисных программ решают различные задачи, в частности:

- тестирование аппаратного обеспечения современного компьютера (ЭВМ):
 - проверка центрального процессора – регистровой памяти, математического сопроцессора, кэш памяти, набора интегрированных инструкций, тактового генератора, часов реального времени и встроенных функций процессора;
 - проверка материнской (системной) платы – тестирование компонентов:
 - проверка северного (North chip) и южного (South chip) чипов – микросхемы большой степени интеграции в основе современного компьютера;
 - тестирование работы микросхемы BIOS (Basic Input Output System);
 - тестирование микросхем памяти SIMM (слот устарел) и DIMM (RIMM);
 - тестирование кэш памяти второго уровня (для некоторых материнских плат);
 - проверка контроллера шины EPC1 и внутреннего видео-адаптера;
 - тестирование контроллера шины AGP и внутреннего видео-адаптера;
 - тестирование шины PCI и внутренних устройств компьютера (ЭВМ);
 - тестирование шины ISA и всех внутренних устройств компьютера;
 - проверка корректности функционирования контроллера FDD и накопителя информации на гибких магнитных дисках (FDD – Floppy Disc Drive);
 - тестирование контроллера IDE и накопителя информации на жестких магнитных дисках (HDD), лазерного накопителя информации на оптических дисках (CD-ROM, DVD-ROM), а также внутреннего электронного накопителя информации (EDD – Electronical Disc Drive) компьютера;
 - тестирование шины USB и внешних периферийных устройств: внешний (Electronical Drive) и внешний портативный накопитель информации на электронных дисках (Flash Drive), а также принтер, сканер и модем;
 - проверка всех внешних периферийных устройств компьютера (ЭВМ):
 - тестирование порта LPT и матричного, струйного или лазерного принтера;
 - тестирование порта COM и внешнего модема (Dial Up, xDSL, WiFi);
- диагностика программного обеспечения современного компьютера (ЭВМ):
 - диагностика микропрограммы BIOS и проверка контрольной суммы;
 - проверка выгружаемых и невыгружаемых компонентов операционной системы;
 - проверка структурных изменений системного реестра операционной системы;
 - проверка поверхности локальных и сетевых логических дисков ЭВМ.

Определение

Пакетом утилит и сервисных программ называют набор программ для ЭВМ, которые позволяют реализовать конфигурирование программной среды операционной системы для оптимизации периода исполнения различных программ пользователя, а также реализовать мониторинг системных конфликтов и неполадок, диагностику аппаратного и программного обеспечения определенного компьютера, архивирование и резервное копирование данных на заданных локальных и сетевых носителях информации, проверку поверхности логических дисков.

6.4.1. Пакеты диагностических утилит

Диагностические программы и утилиты имеют очень важное значение в выявлении конфликтов в аппаратном и программном обеспечении, которое находится под управлением операционной системы и позволяют реализовать ее диагностику.

Определение

Пакетом диагностических программ называют совокупность программ, которые реализуют мониторинг конфликтов аппаратного и программного обеспечения находящегося под управлением сетевой или локальной операционной системы.

Существует большое количество диагностических утилит, но можно выделить несколько наиболее популярных программ, которые получили самое широкое распространение среди квалифицированных специалистов и конечных пользователей:

- Scan Disc – утилита, которая встроена в операционную систему Windows и реализует тестирования поверхности логических дисков накопителей информации;
- Scan CD – утилита проверки поверхности оптического диска CD-ROM и DVD-ROM;
- Check Disc – утилита, которая предназначена для быстрого тестирования поверхности логических дисков на накопителях информации различного типа;
- Norton Disc Doctor – утилита для диагностики накопителей информации на гибких (Floppy Disc Drive) и жестких магнитных дисках (Hard Disc Drive);
- Check It – простейшая утилита для диагностики аппаратного и программного обеспечения, которая предназначена для операционной системы MS-DOS и MS Windows, обеспечивает тестирование центрального процессора, микросхем памяти, системной платы, накопителей информации на гибких и жестких магнитных дисках, накопителя информации на оптических дисках, электронного накопителя информации, звукового адаптера и аудио-адаптера;
- MS Diagnostics – утилита, которая позволяет протестировать оборудование;
- AMI Diagnostics – утилита для реализации диагностики различных компонентов, которые относятся к аппаратному и программному обеспечению ЭВМ;
- PC Doctor – утилита для реализации тестирования аппаратного и программного обеспечения компьютера (ЭВМ) в операционной системе MS-DOS;
- PC Check – утилита для диагностики аппаратных и программных компонентов компьютера, которая предназначена для операционной системы MS-DOS;
- PC Diagnostics – утилита для тестирования аппаратного и программного обеспечения современного компьютера для операционной системы MS Windows;
- Modem Doctor – позволяет определить модель, протестировать контрольную сумму ROM (Flash EEPROM), RAM, AT-команды, индицировать состояние & и S – регистров, а также узнать версию микропрограммы и обеспечить диагностику буферов приема и передачи данных, реализовать контрольную посылку и прием;
- Sandra Delux – современная многофункциональная утилита под операционную систему MS Windows, которая реализует диагностику перечисленных компонентов аппаратного и программного обеспечения в автоматическом режиме.

Пакет диагностических утилит Sisoft Sandra Delux включает совокупность программных компонентов, которые непосредственно после инсталляции позволяют реализовать автоматическую и автоматизированную диагностику компонентов ЭВМ:

- центрального процессора – позволяет выявить аппаратные ошибки при работе:
 - нарушения в работе тактового генератора и часов реального времени;
 - ошибки при работе математического сопроцессора центрального процессора;
 - наличие нечитаемых ячеек данных в буферной памяти процессора ЭВМ;
- модулей памяти, которые реализуют оперативное запоминающее устройством:
 - проверка EPROM-чипа при записи и чтении данных на модуль памяти;
 - проверка скорости чтения данных с микросхем модуля памяти (линейное последовательное и псевдослучайное чтение логических нулей и единиц);
 - выявление скорости записи данных на микросхемы модуля памяти (линейная последовательная запись блоков данных и псевдослучайная запись логических нулей и единиц в рамках нескольких циклов с повторами);
 - режим автоматического многократного тестирования модуля памяти, который предполагает: цикл записи блоков данных определенного объема разными способами, а также последующее линейное и псевдослучайное чтение логических состояний на проверяемом модуле памяти компьютера;
- накопителя информации на гибких магнитных дисках (Floppy Disc Drive):
 - проверка головок чтения информации в режиме чтения данных с дискеты;
 - проверка головок записи информации в режиме записи данных с дискеты;
 - проверка совместного использования головок чтения-записи данных;
 - тестирование режима защиты от записи информации (переключатель на дискете);
- накопителя информации на жестких магнитных дисках (Hard Disc Drive):
 - тестирование контроллера накопителя информации на жестких магнитных дисках;
 - проверка механизма вращения оси (шпинделя), которая содержит магнитные диски с поверхностями со свойствами ферромагнетика (стресс тест);
 - определение объема и последующая проверка внутренней кэш памяти;
 - определение скорости поиска, чтения и записи информации при использовании контрольного пакета данных разного размера (512 Кб, 1Мб, 10Мб);
 - тестирование чтения-записи информации с поверхности магнитного диска;
 - проверка данных SMART-таблицы, которая отражает количество старто-стопов (start/stop number), ошибок чтения и записи (read/write errors), отклонений скорости вращения (mechanics errors), а также прогнозирование даты выхода из строя накопителя информации жестких магнитных дисках;
- электронного накопителя информации (Electronical Disc, Flash Disc) и электронных карт памяти внешнего электронного накопителя (Memory Stick, Secure Digital):
 - проверка корректности чтения информации в режиме чтения данных;
 - проверка корректности записи информации в режиме записи данных;
 - проверка совместного использования режима чтения-записи данных;
 - тестирование режима защиты от записи информации (переключатель на карте);

- лазерного накопителя информации на оптических дисках (CD-ROM, DVD-ROM):
 - тестирование контроллера лазерного накопителя на оптических дисках;
 - тестирование механизмов привода (двигатель вращения оси, двигатель перемещения головок чтения-записи информации на оптическую поверхность);
 - определение логического объема и проверка внутренней кэш памяти;
 - тестирование головок чтения-записи в режимах чтения, стирания и записи;
 - определение скорости поиска, чтения и записи информации при использовании контрольного пакета данных разного размера (512 Кб, 1Мб, 5Мб, 10Мб);
- аудио-адаптера, который реализует запись и воспроизведение аудио-потока:
 - тестирование контроллера аудио-адаптера, определение типа, серии и модели;
 - диагностика музыкального сопроцессора или синтезатора набора звуков;
 - процедура тестирования воспроизведения информации по всем и отдельным каналам аудио-карты на разных частотах в интервале от 300 Гц до 45000 Гц;
- видео-адаптера, который обеспечивает запись и воспроизведение видео-потока:
 - тестирование контроллера видео-адаптера на разном разрешении и глубине цвета;
 - тестирование видео-адаптера в режиме захвата видео-потока от камеры или внешнего устройства с предустановленной разной глубиной цвета, разрешающей способностью и частотой дискретизации (framerate);
 - диагностика аппаратных и программных алгоритмов сжатия видео-данных;
- сетевого адаптера, который позволяет реализовать сетевое взаимодействие ЭВМ:
 - тестирование контроллера сетевой карты и внутренних буферов памяти;
 - проверка коммутации сетевого адаптера и штекера кабеля передачи данных;
 - проверка приема пакетов данных определенного или произвольного размера;
 - проверка передачи пакетов данных фиксированного или случайного размера;
 - проверка протоколов передачи данных серии TCP/IP, Netbios, IPX/SPX;
- матричного, струйного или лазерного принтера, который реализует печать информации на носители разного вида (бумага, картон, металл, стекло, пленка):
 - тестирование контроллера принтера и порта передачи данных (LPT, USB);
 - проверка внутреннего процессора принтера, определение серии, модели, языка передачи данных для операционной системы MS-DOS и MS Windows (PCL6);
 - выявление объема и тестирование внутреннего буфера памяти принтера;
 - печать тестовой страницы с использованием разной разрешающей способности;
- сканера, который обеспечивает ввод растрового графического изображения:
 - тестирование контроллера сканера и порта передачи данных (LPT, USB);
 - проверка процессора сканера, определение модели и технических параметров;
 - определение объема и тестирование буфера внутренней памяти сканера;
 - сканирование растрового графического изображения на разном разрешении;
- внешнего модема, который позволяет принимать и передавать данные:
 - тестирование контроллера внешнего модема и порта подключения (COM, USB);
 - проверка процессора модема, выявление серии, типа, модели, набора команд;
 - тестирование внутренней памяти и системы внутренних AT-команд модема.

6.4.2. Пакеты сервисных программ

Пакеты сервисных программ позволяют выполнить оптимизацию программного окружения локальной и сетевой операционной системы и всего аппаратного и программного обеспечения, которое находится под ее непосредственным управлением.

Выделяют большое количество сервисных программ, которые подразделяются на:

- программное обеспечение для подавления вирусной опасности (антивирусы):
 - сетевой экран (Firewall) – реализует мониторинг сетевой активности и блокирование исходящих и входящих соединений на уровне слотов и сокетов;
 - антивирусный монитор (Antiviral monitor) – резидентно загружается после запуска операционной системы и обеспечивает подавление вирусной опасности;
 - антивирусные сканеры (Antiviral scanner) – реализует сканирование логических дисков накопителя информации по требованию пользователя;
 - мониторы реестра операционных систем (Reg Optimizer) – индицирует несанкционированное изменение разделов и ключей системного реестра;
 - программы для извлечения определенных типов вирусов и троянских программ (Antispyware) – осуществляют поиск и удаление вредоносного кода;
 - программы для комплексной защиты информационных систем (Complex Antiviral Software) – осуществляют комплексную защиту операционной системы;
- программы для обслуживания накопителей информации на разных носителях:
 - программы разметки накопителей информации (PowerQuest Partition Magic);
 - программы для создания образов логических дисков (Power Quest Drive Image, Power Quest Drive Copy, Norton Drive Image, Norton Drive Copy);
 - программы для восстановления данных с логических дисков после их неудачного форматирования (DOS Unformat Utility, Norton Unformat);
 - программы для восстановления данных с логических дисков после случайного удаления файлов и папок (DOS Undelete Utility, Norton Undelete);
 - программы для восстановления данных с логических дисков после повреждения файлов и папок (Final Data Recovery, Ontrack Easy Recovery);
- программы для комплексного обслуживания аппаратуры и программ для ЭВМ:
 - утилиты для комплексного обслуживания компьютера (Norton Utilities);
- программы для реализации криптографического кодирования данных ЭВМ:
 - программные системы для криптографического кодирования данных на накопителях информации и носителях, которые содержат важные данные;
 - программные системы хранения учетных записей (accounts) (Password Case);
 - программы для хранения паролей банковских карт (Password Encrypt);
- программное обеспечение для охранных и пожарных систем организаций:
 - микропрограммное и программное обеспечение систем контроля доступа;
 - программы систем идентификации по биометрическим параметрам;
- архиваторы – позволяют реализовать компрессию информации на основе алгоритмов сжатия данных (Arc, Ace, Win Rar, Win Zip, Win Arj, 7Zip, UAE, Tar Gz).

Определение

Пакетом сервисных программ называют программное обеспечение, которое позволяет реализовать мониторинг активности технического средства (принтер, сетевой адаптер, сканер) или архивирование и резервное копирование данных на сетевом или локальном накопителе информации, который используется на компьютере (ЭВМ).

Пакет утилит Norton Utilities (Norton System Works) включает совокупность различных программных компонентов (модулей), которые реализуют быструю диагностику и обслуживание аппаратного и программного обеспечения компьютера:

- Norton Scan Disc – программа для сканирования файловой системы логического диска с целью выявления ошибок в ее структуре, а также файлов и папках;
- Norton Disc Doctor – позволяет выявить механические повреждения в аппаратуре и ошибки в микропрограмме накопителей информации на разных носителях;
- Norton Disc Defragmenter – реализует процесс дефрагментации информационных элементов файловой системы для оптимизации работы операционной системы;
- Norton System Monitor – отражает состояние процессора, оперативной памяти, накопителя информации на жестких магнитных дисках, перечень процессов;
- Norton Registry Monitor – отслеживает состояние реестра и индицирует изменения;
- Norton Go Back – реализует съем и сохранение критических областей операционной системы для отката внесенных деструктивных изменений в ее основе;
- Norton Antivirus – подавляет вирусную опасность обусловленную внутренними угрозами в операционной системе и внешними угрозами при работе в сети;
- Norton Mail Security – реализует мониторинг почтового трафика входящих и исходящих электронных писем на наличие деструктивных вложений (файлов);
- Norton Ghost – реализует возможность создания системного загрузочного меню при инсталляции и использовании нескольких операционных систем на ЭВМ;
- Norton Partition Manager – реализует возможность разметки накопителя информации: создание основного и дополнительного раздела и логических дисков;
- Norton Wipe Info – реализует функцию шредера (надежного удаления) папок и файлов, которые расположены в основе файловой системы компьютера (ЭВМ);
- Norton UE32 – позволяет сжать файлы посредством использования UUE-формата;
- Tutorial System – содержит интегрированную справочную систему, которая позволяет реализовать обучение пользователя использованию пакета утилит;
- Norton Automatic Update – компонент позволяющий реализовать автоматическое или ручное обновление компонентов пакета утилит Norton System Works;
- Norton Definst – вспомогательная утилита для проверки корректности установки кумулятивного пакета обновлений ручным или автоматическим способом;
- Norton PassMark Performance Test – вспомогательная утилита позволяет отображать состояние аппаратного и программного обеспечения компьютера (ЭВМ).

Пакет утилит Norton System Works реализует комплексную диагностику ЭВМ.

Заключение

Процессы глобальной интеграции в политике, экономике, обществе и интенсификация роста разнородных источников и потребителей информации обуславливают проблему информационного кризиса, который проявляется в информационной лавине и голоде, что, в свою очередь, инициирует создание, распределение и использование информационных ресурсов, продуктов и услуг посредством современных достижений в области информационных и коммуникационных технологий и средств автоматизации: аппаратное, программное и алгоритмическое обеспечение.

Информатизация разных сфер деятельности постиндустриального общества позволяет обеспечить существенный прирост эффективности производственной и непроизводственной деятельности, оптимизировать выполнение совокупности операций, достичь сокращения себестоимости продукции, временных и транзакционных издержек, а также глубже изучить закономерности информационных процессов, сформировать правовые основы деятельности субъектов информационной индустрии.

Информационные ресурсы различного вида и назначения концентрируют агрегат накопленных знаний по совокупности предметных областей, при этом они выступают основой для создания информационных продуктов и оказания услуг.

Внутренние и внешние потребительские свойства информации существенно влияют на ее полезность по отношению к определенной предметной области.

Информационное взаимодействие имеет очень сложную структуру и существенные специфические отличия по отношению к определенной сфере получения, обработки и передачи данных посредством использования средств автоматизации.

Цифровой автомат является дискретным устройством по принципу функционирования, а условия аппаратной реализуемости алгоритмического обеспечения обуславливают необходимость разработки аналого-цифровых и цифро-аналоговых преобразователей для дискретизации непрерывного аналогового сигнала в дискретный аналог для последующего представления в форме данных подлежащих обработке.

Арифметические основы цифровых автоматов сводятся к рассмотрению особенностей преобразования целых чисел и правильных дробей из одной системы счисления в другую, а также правил выполнения простейших арифметических операций: со снижением номинала основания системы счисления возникает проблема переполнения разрядной сетки при представлении числа, а с его ростом наблюдается существенное повышение сложности переноса в старший разряд и заема значащей единицы при использовании методов сложения, вычитания, умножения и деления.

Логические основы цифровых автоматов сводятся к рассмотрению совокупности простых и сложных логических операций и функций, а также особенностям их представления в аналитическом, табличном и схематическом виде (временная диаграмма, функциональная схема или электрическая принципиальная схема). Немаловажное значение имеют законы алгебры логики как правила эквивалентного преобразования и структурной декомпозиции сложных логических выражений

к простым. Рассмотрены устройства комбинаторной логики обладающие памятью состояния и простейшая реализация бистабильной ячейки способной хранить 1 бит информации: асинхронный RS-триггер, синхронный RS-триггер и тактируемый T-триггер.

Зарождение и развитие информационных систем связывают с началом исследований и разработок закономерностей процессов информационного обмена в разных сферах деятельности постиндустриального общества, поскольку способствует созданию внедрению и использованию аппаратного, программного и алгоритмического обеспечения на основе информационных и коммуникационных технологий: ИТ получения данных, ИТ обработки данных, ИТ хранения данных, ИТ передачи данных.

Классическая архитектура ЭВМ предложена в Принстонском университете США коллективом ученых под руководством Джона фон Неймана и характеризуется несколькими ключевыми принципами: микропрограммного управления, наличия сегментированной памяти со стандартизированной структурой адресов, наличие оператора условного перехода для смены линейной последовательности выполнения команд для реализации ветвления алгоритмической структуры, которые на современном этапе дополняются принципом открытой архитектуры для обеспечения гарантированной возможности интеграции аппаратного и программного обеспечения посредством разработки, внедрения и последующего использования совокупности технологических стандартов на интерфейсы и программное обеспечение.

Современная архитектура ЭВМ и ИС представлена обеспечением разного вида:

- алгоритмическое обеспечение – включает совокупность математического обеспечения цифровых автоматов: принципы функционирования компонентов архитектуры ЭВМ, эффективные и оптимальные алгоритмы сжатия и передачи данных, методы криптографической защиты и шифрование данных;
- аппаратное обеспечение – представлено совокупностью устройств и компонентов расположенных внутри системного блока компьютера и подключаемых к нему посредством использования стандартизованного набора портов;
- программное – включает различные локальные и сетевые многопользовательские и однопользовательские операционные системы, а также пакеты прикладных программ разного вида, диагностические программы и утилиты.

Рассмотрены внутренние элементы архитектуры современных компьютеров, в частности: системная плата и блок питания, центральный процессор, модули памяти, накопитель информации на гибких и жестких магнитных дисках, лазерный накопитель информации на оптических дисках, электронный накопитель информации.

Представлены внешние периферийные элементы и устройства ЭВМ: системный блок и блок питания, монитор, клавиатура, манипулятор (мышь, тачпад, трекбол), принтер, сканер, модем, стереофоническая система объемного воспроизведения.

Таким образом, стремительное развитие информационных и коммуникационных технологий обусловило их проникновение во все сферы деятельности информационного общества независимо от рода деятельности и предметной области.

Перечень литературных источников

1. Александров В.В. Информатика: Энциклопедический словарь для начинающих/ Под общ. ред. Д.А. Поспелова. – М. : Педагогика-пресс, 1994. – 349 с.
2. Бауэр Ф.Л., Гооз Г. Информатика. Вводный курс: в 2-х частях,- М.: Мир, 1990. – 1066 с.
3. Башмаков А.И. Разработка компьютерных учебников и обучающих систем. – М.: Филин, 2003. – 613 с.
4. Брой М. Информатика. Основополагающее введение. - М.: ДИАЛОГ-МИФИ, 1998. – 682 с.
5. Ваулина Е.Ю., Рычков В.Н. Термины современной информатики: программирование, вычислительная техника, Интернет. Англо-русский, русско-английский словарь. – М.: Изд-во Эксмо, 2004. – 640 с.
6. Ершов А.П. и др. Основы информатики и вычислительной техники. Пробный учебник для 10-11-х классов. - М.: Просвещение, 1985. – 95 с.
7. Информатика. Базовый курс: Под. ред. С.В. Симоновича,- СПб.: Питер, 1999. – 639 с.
8. Информатика: Базовый курс. / Под ред. Н.В. Макаровой. - СПб.: Питер, 2001. – 336 с.
9. Пройдаков Э.М., Теплицкий Л.А. Англо-русский толковый словарь по вычислительной технике, Интернету и программированию. – 2-е изд., испр. И доп. – М.: Издательство торговый дом «Русская редакция», 2000. – 448 с.
10. Каймин В.А. Информатика: Учебник / В.А. Каймин; Междунар. акад. информатизации. – М.: Высшее образование: Агентство учебной литературы "РеСК", 1998. – 334 с.
11. Каймин В.А. Информатика: учебник / В. А. Каймин. - Москва: Проспект, 2007. – 270 с.
12. Информатика: Учеб. пособие и сборник задач с решениями / Под общ. ред. Каймина В. А., Муравья Л. А. – М.: Бридж, 1994. – 194 с.
13. Каймин В.А. Основы компьютерной технологии. – М.: Финансы и статистика, 1992. – 207 с.
14. Кураков Л.П. Информация как объект правовой защиты / Л.П. Кураков, С. Н. Смирнов. – М.: Гелиос, 1998. – 238 с.
15. Новиков Ф.А. Яценко А.Д. Microsoft Office 2000 в целом. – СПб.: БХВ, 2000. – 728 с.
16. Осипов Г.С. Приобретение знаний интеллектуальными системами: Основы теории и технологии. – М.: Наука, 1997. – 109 с.
17. Попов Э.В. Общение с ЭВМ на естественном языке. – М.: Наука, 1982. – 360 с.
18. Семакин И.Г. и др. Информатика. Учебник для 7-9 классов. - М.: БИНОМ, 1998.
19. Симонович С.В. Общая информатика: новое издание – М.: Питер, 2007. – 431 с.
20. Стинсон К., Зихерт К. Microsoft Windows 2000 Professional. Справочник профессионала / Пер. с англ. – М.: Издательство ЭКОМ, 2003. – 816 с.
21. Таненбаум Э. Архитектура компьютера. – СПб.: Питер, 2002
22. Таненбаум Э. Современные операционные системы. – СПб.: Питер, 2002.
23. Топоркова О.М. Информатика: Учебн. пособ. – Калининград: КГТУ, 2001.
24. Фигурнов Э.В. IBM PC для пользователя. - М.: ИНФРА-М, 1999.
25. Фигурнов В.Э. Windows для начинающих и опытных. – М.: Инфра-М, 2006.
26. Шафрин Ю.А. Основы компьютерной технологии. Учебное пособие для учащихся старших классов. - М.: АБФ, 1996.

Информатика
Учебник

Ветров Анатолий Николаевич, 2008 ©
Санкт-Петербург, 2008