

## **Содержание:**

# **Введение**

Актуальность исследования. Программное обеспечение (ПО) – совокупность специальных программ, облегчающих процесс подготовки задач к выполнению на ЭВМ и организующих прохождение их через машину, а также процедур, описаний, инструкций и правил вместе со всей связанной с этими компонентами документацией, используемых при эксплуатации вычислительной системы.

Обрабатывают информацию, управляют работой компьютера программы, а не устройства.

Новинки программного обеспечения уже давно доминируют над новыми аппаратными разработками. Комплект ПО по стоимости превосходит (иногда в несколько раз) стоимость компьютера адекватного класса.

Целью данной работы является аналитический обзор современного обеспечения программного обеспечения персонального компьютера, для достижения поставленной цели, были выделены следующие задачи:

- рассмотреть теоретические аспекты исследования программного обеспечения персонального компьютера;
- провести анализ программного обеспечения в ПК.

Объект исследования – программное обеспечение.

Предмет исследования - аналитический обзор современного обеспечения программного обеспечения персонального компьютера.

Структура работы состоит из введения, основной части, заключения и списка литературы.

Теоретической и методологической базой данной работы послужили труды российских и зарубежных авторов в области информатики, материалы периодических изданий и сети Интернет.

# Глава 1 Теоретические аспекты исследования программного обеспечения персонального компьютера

## 1.1 Понятие и типы программного обеспечения

Для эффективного использования компьютера должно соблюдаться соответствие между уровнем развития вычислительной техники и программного обеспечения. С одной стороны, ПО определяет функциональные возможности компьютера. С другой, установка конкретного ПО может быть ограничена конструктивными особенностями компьютера.

Назначение ПО:

- обеспечение работоспособности компьютера;
- облегчение взаимодействия пользователя с компьютером;
- сокращение цикла от постановки задачи до получения результата;
- повышение эффективности использования ресурсов компьютера.

Программное обеспечение позволяет:

- усовершенствовать организацию работы вычислительной системы с целью максимального использования ее возможностей;
- повысить производительность и качество труда пользователя;
- адаптировать программы пользователя к ресурсам конкретной вычислительной системы;
- расширить ПО вычислительной системы[1].

Максимальное использование возможностей вычислительной системы достигается, во-первых, за счет выделения каждому пользователю или задаче минимально необходимых ресурсов для своевременного и качественного решения его задач, во-вторых, за счет подключения к ресурсам вычислительной системы большого числа пользователей (в том числе и удаленных), в-третьих, путем перераспределения ресурсов между различными пользователями и задачами в зависимости от состояния системы и запросов на обработку.

Повышение производительности и качества труда пользователей происходит за счет автоматизации процедур расчетного и оформительского характера, реализуемых с помощью разнообразных средств программирования (алгоритмическим языком, пакетов прикладных программ) и удобных устройств ввода-вывода информации.

Адаптируемость программ пользователя к ресурсам конкретной вычислительной системы обеспечивается тем, что операционная система содержит средство обслуживания большого диапазона машинных конфигураций. Кроме того, операционная система позволяет создавать и легко настраивать существующие программы на различные устройства ввода-вывода.

Расширение существующего ПО предполагает наличие следующих возможностей:

- создание пользователем собственных программ и пакетов, реализующих как конкретные расчетные задачи, так и процессы управления отдельными устройствами и всей вычислительной системой в целом;
- дополнение существующего ПО программами, позволяющими расширять возможности операционной системы, работать с новыми типами внешних устройств, новыми вычислительными системами (компьютерами), в новых областях применения.

ПО ориентировано на использование вычислительных систем в различных сферах деятельности и должно обеспечивать своевременное и адекватное поставленным задачам решение. Это вызывает необходимость соблюдения ряда требований при разработке компонентов ПО, основными из которых являются:

- модульность;
- наращиваемость и развитие;
- надежность;
- предсказуемость;
- удобство и эргономичность;
- гибкость;
- эффективность;
- совместимость.

## **1.2 Установка программного обеспечения**

Основные принципы разработки современного программного обеспечения:

- параметрическая универсальность;
- функциональная избыточность;
- функциональная избирательность.

Программы на компьютер можно установить двумя способами:

Инсталляцией с дистрибутива

Простым копированием



Рис. 1. Структурная схема программного обеспечения ЭВМ

Рис. 1.1 – Структурная схема программного обеспечения ЭВМ[2]

Первый (низший) уровень иерархии занимает внутреннее программное обеспечение ПЭВМ, сохраняемое в ее постоянной памяти. С его помощью ПЭВМ выполняет основные функции, определяемые аппаратной структурой. Программы внутреннего ПО работают непосредственно с аппаратными модулями компьютера. Вследствие этого они функционально связаны с ними и при замене определенного аппаратного модуля требуется заменить и программу внутреннего ПО,

предназначенную для работы с ним.

Программы, обслуживающие аппаратные модули, называются драйверными программами или драйверами. Они позволяют при замене или подключении нового аппаратного модуля не производить изменений в других программах ПЭВМ, а только сменить драйвер соответствующего аппаратного модуля.

Внутреннее ПО представляет собой программный интерфейс, обеспечивающий взаимосвязь работы компьютера со всеми остальными программами. Доступ к программам внутреннего ПО производится только через систему программных прерываний.

Внутреннее ПО выполняет следующие основные функции:

- управляет широким набором периферийных устройств;
- осуществляет быструю проверку работоспособности ПЭВМ при ее включении;
- устанавливает отдельные аппаратные модули в исходное состояние;
- загружает программы ОС.

Основными элементами внутреннего ПО служат драйверы ввода-вывода, программа самопроверки и программа первоначальной загрузки. Внутреннее ПО взаимодействует, с одной стороны, с функциональными модулями ПЭВМ, а с другой стороны, реализует программный интерфейс операционной системы.

Программа самопроверки предназначена для проверки функциональных модулей ПЭВМ, т.е. установки схем компьютера в начальное состояние путем загрузки программных регистров необходимой информацией. При проверке отдельных функциональных модулей ПЭВМ в них могут быть обнаружены неисправности. Программа самопроверки сообщает пользователю об обнаруженных неисправностях с помощью сообщений на экране и (или) звукового сигнала.

При обнаружении ошибки проверку компьютера можно продолжить с помощью диагностических программ, загружаемых с дискеты. Если ошибка не нарушает работоспособности ПЭВМ, то по желанию пользователя ею можно пренебречь. Если в состав ПЭВМ включается новый функциональный модуль, то к общей программе самопроверки добавляется программа самопроверки данного модуля.

При успешном завершении самопроверки ПЭВМ готова к работе. Управление через программное прерывание передается программе начальной загрузки. Эта программа предназначена для считывания в оперативную память остальных

компонентов операционной системы. При успешном выполнении этой операции управление передается только что считанной программе.

Драйверы ввода-вывода используются для обслуживания периферийных устройств ПЭВМ. Эти программы работают непосредственно с соответствующими контроллерами, что позволяет пользователю не знать физической организации конкретного устройства и работать только с командами драйвера, реализующими его обслуживание.

Драйверы имеют следующие особенности:

- открытую структуру, что позволяет добавлять в систему новые драйверы;
- гибкость организации доступа к драйверам через программные прерывания, что позволяет не фиксировать их в строго определенных областях памяти, быстро и легко их заменять;
- настраиваемую структуру, ориентирующую драйверные программы на определенный класс периферийных устройств, параметры которых размещены в специальных таблицах. Драйверы настраиваются на конкретные периферийные устройства при помощи изменений значений в этих таблицах;
- резидентное размещение в оперативной памяти, позволяющее использовать драйвер в любой момент времени из любой программы.

К основным драйверным программам относятся: драйвер жесткого диска, драйвер видеоадаптера, драйвер клавиатуры, драйвер печатающего устройства, системные драйверы (установка таймера, проверка конфигурации компьютера, определение емкости ОЗУ), дополнительные драйверы (драйвер связи и др.)[\[3\]](#).

Операционная система занимает второй (средний) уровень иерархии ПО. Она управляет ресурсами компьютерной системы, к которым относятся оперативная и внешняя память, устройства ввода-вывода и программы пользователя. ОС взаимодействует с компьютером через интерфейс внутреннего ПО. Это дает возможность ПЭВМ, имеющим аппаратные различия, работать с одной и той же операционной системой.

## **1.3 Состав и классификация программного обеспечения**

Состав ПО определяется кругом задач, которые пользователь предполагает решать с помощью компьютера.

По назначению, т.е. в зависимости от класса решаемых задач, ПО обычно разделяют на две основные группы: Общее (базовое) и прикладное.

Классификация программного обеспечения по функциональному назначению

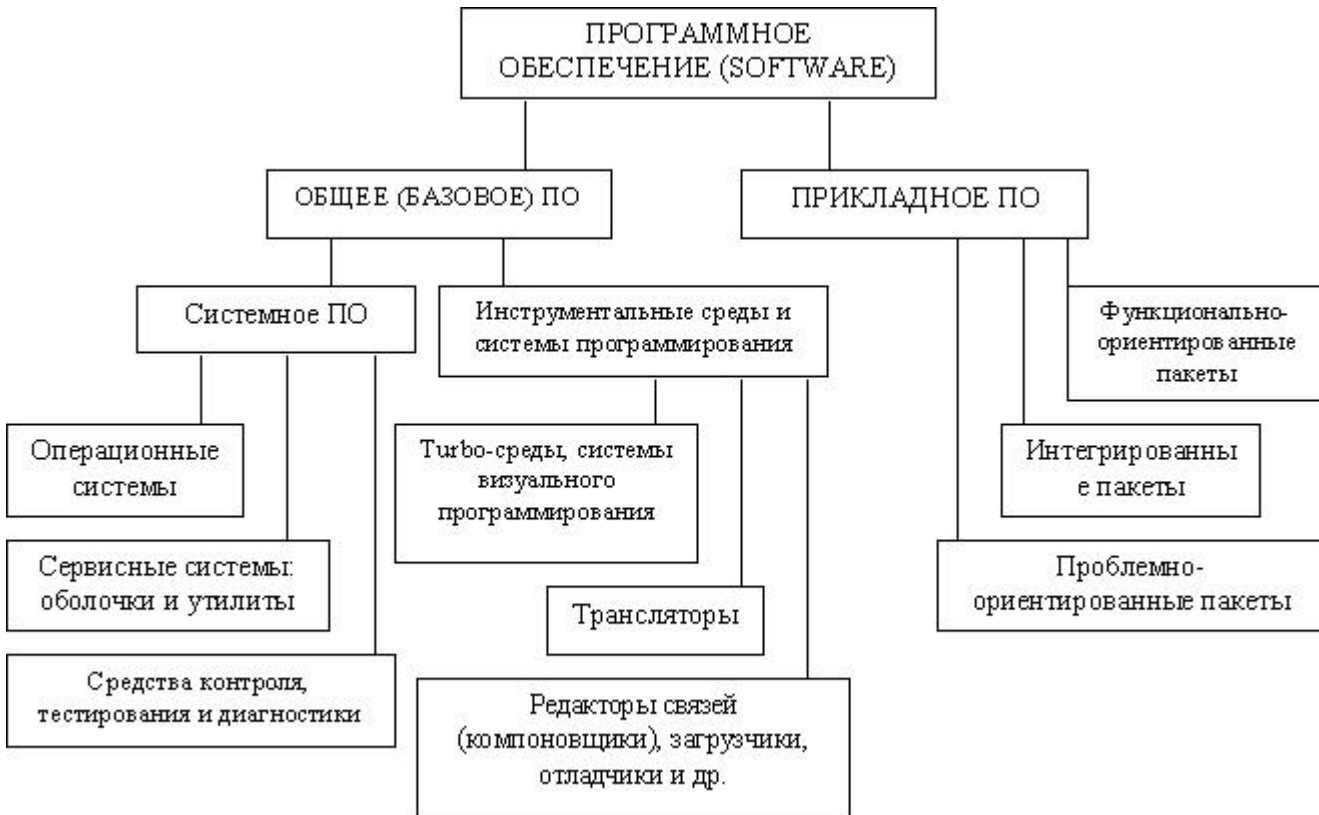


Рис. 1.2 - Схема общей классификации ПО[4]

Системное ПО – совокупность программ, обеспечивающих работоспособность компьютера; комплекс программ, которые осуществляют организацию вычислительного процесса и управление ресурсами компьютера.

Системы программирования (Инструментальное ПО) – совокупность программных средств, позволяющих разрабатывать программы.

Прикладное ПО – совокупность программ, предназначенных для решения задач из различных сфер человеческой деятельности.

# Глава 2 Анализ программного обеспечения в ПК

## 2.1 Стандартные программы в Windows

**Стандартные программы в Windows** - набор полезных и популярных программ, поставляемых вместе с Windows. Служат для различных целей. Стандартными программами ос Windows являются: Блокнот - простой текстовый редактор, Paint - простой и удобный графический редактор, Калькулятор, WordPad, Подключение к удаленному рабочему столу, Проводник, Служебные программы и другие. К Служебным программам относятся: Дефрагментация диска, очистка диска, восстановление системы, монитор ресурсов, панель управления, сведения о системе и другие.

**Блокнот** — небольшая стандартная программа в составе Windows для просмотра и редактирования текстовых файлов (как правило \*. TXT). Блокнот можно использовать для многих целей, например, для создания небольших текстовых заметок, так как он открывается, как правило, быстрее чем офисные программы.

**Paint** — Простой графический редактор, стандартная программа, входящая в состав всех версий Windows. Он также часто упоминается как MS Paint или Microsoft Paint. Позволяет создавать рисунки, картинки и т.д. или редактировать изображения. Кроме того, его можно использовать для сохранения графических файлов в различных форматах.

В стандартных программах присутствует инструкция по работе с программой - встроенная справка. Для вызова справки можно нажать клавишу F1 или значок знак вопроса в правом верхнем углу программы, если он присутствует.

Где найти в Windows 10, 8.1, 8

В Windows 10:

Нажать "Пуск" - "Все приложения" - "Стандартные - Windows"

В Windows 8:

Каждый, кто начинает работать с новой Windows 8 или 8.1 заметит, что интерфейс меню пуск совершенно отличается от предыдущих версий Виндовс и пользователю требуется некоторое время, чтобы привыкнуть к нему. Любой, кто использует



Стандартные программы знает, как открыть их в предыдущих Windows, но это немного отличается в Windows 8 в связи с изменением Меню Пуск. Если вы хотите открыть стандартную программу и вы откроете меню Пуск, то заметите, что пункт стандартные программы исчез, и появился стартовый экран[5].

Есть несколько способов открыть Стандартные программы в Windows 8, например:

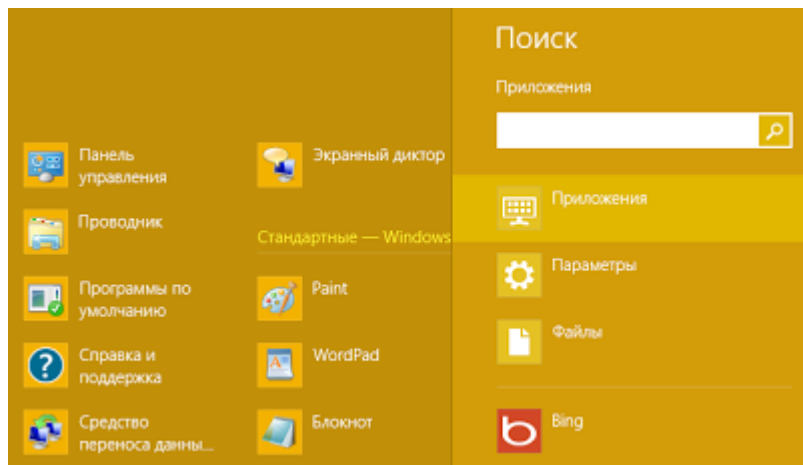


Рис. 2.1 – Способы открытия стандартных программ в Windows 8

1. В любом окне виндовс или в окне Пуск подвести мышь в правый верхний угол вашего экрана или рабочего стола, всплывет вертикальное меню: Поиск, Общий доступ, Пуск, Устройства, Параметры. Нажать значок Лупы - кнопка поиск - откроется окно Приложения, где все ваши программы.

3.а Ведите бегунок нижнего скроллбара вправо - увидите группу

Стандартные программы

Там и есть ваши искомые программы.

3.б Можно не водить бегунок и не искать во всем списке программ, а с правой стороны данного окна в строке поиска написать, например, Блокнот или notepad и слева увидите результат поиска в программах - блокнот.

## 2.2 Операционная система

Все разнообразие ПО (программного обеспечения) по большому счету делят на системные и прикладные программы. Первая группа обеспечивает работу второй на имеющемся «железе» (процессоре, дисках, оперативной памяти, устройствах

ввода вывода). Операционные системы (ОС) относятся к системному ПО. Одной из задач ОС является реализация алгоритмов работы с аппаратным обеспечением. Может возникнуть вопрос: зачем это нужно? Ведь если подумать, каждая прикладная программа может включать код, обеспечивающий обращение к «железу». Однако, это только бы усложнило жизнь программистам и раздуло бы ПО до больших размеров. И что самое грустное — в прикладных программах было бы много одинакового кода, отвечающего за реализацию низкоуровневых команд (обращений к железу). Кроме того, как решить проблему совместной работы разных программ на одном компьютере — еще один вопрос. Поэтому операционные системы и другое системное ПО вполне обоснованно занимают отведенную им роль посредника между прикладным ПО и аппаратным обеспечением компьютера.



Рис. 2.2 – Упрощенная схема логической структуры компьютера

Даже в своем историческом развитии операционные системы зародились именно как набор программ и библиотек для управления операциями ввода и вывода.

Этими достаточно универсальными программами далее пользовались остальные программисты, которым уже не нужно было ломать голову как запрограммировать считывание данных с дискеты или вывод текста на принтер. Они просто вызывали функцию из подключенной библиотеки, а она делала всю работу (в ней уже был заложен код работы с физическими устройствами).

С течением времени операционная система все более усложнялась, на нее возлагали новые функции. Компьютеры становились мощнее, потребовалась одновременно запускать определенное множество программ на выполнение процессору. ОС стала решать задачи эффективного распределения ресурсов «железа» между работающими программами. С одной вычислительной машиной стали одновременно работать несколько пользователей. ОС стала следить за правами каждого и защищать данные. В результате современные ОС включают в себя множество различных функций[6].

По своему строению операционная система представляет комплекс программ и модулей. Выделяют понятие ядра операционной системы. Программное обеспечение ядра защищено от вмешательства пользователей и программистов. К ядру прикладные программы обращаются с помощью запросов на выполнение того или иного действия с аппаратным обеспечением. Эти запросы называются системными вызовами и представляют собой специальные команды.

Итак, операционная система выполняет две основные задачи:

1. облегчает (или даже предоставляет возможность) пользователям и программистам использование аппаратного обеспечения. Например, операционная система дает возможность абстрагироваться от того как на самом деле происходит обработка данных на жестком диске, а работать с понятием файла.
2. обеспечивает эффективное использование аппаратного обеспечения. Поскольку на современных вычислительных машинах одновременно запускаются далеко не одна программа, то ОС отвечает за распределение памяти, регистров процессора и др. между запущенными программами в каждый момент времени. ОС определяет оптимальное распределение этих ресурсов во времени (использование процессора программами по очереди) и пространстве (загрузка в разные части оперативной памяти разных программ).

На сегодняшний день наиболее популярными являются операционные системы семейства Windows, которые являются проприетарным (коммерческим) продуктом

корпорации Microsoft.

Свою «родословную» Windows начинают от операционной системы DOS и первоначально представляли собой надстраиваемые над ней оболочки (Windows запускался из под DOS), увеличивающие возможности DOS и облегчающие неподготовленному пользователю работу с компьютером. Уже более поздние версии (начиная с Windows NT) представляли собой полноценные операционные системы.

Преимуществом Windows считается дружелюбный для пользователя интерфейс. Из недостатков отмечают ненадежность системы.

Операционная система UNIX оказала большое влияние на развитие мира операционных систем, заложив основы работы современных ОС. Изначально UNIX был системой для разработки ПО. В основном в UNIX работали программисты (да и вообще в 70-е годы мало кто другой работал с вычислительными машинами).

UNIX развивался на нескольких фундаментальных идеях. Например, одна небольшая задача должна решаться одной небольшой программой, а сложные задачи должны быть решаемы комбинацией простых программ.

В UNIX большое внимание уделено распределению ресурсов компьютера между пользователями. Эта система является мультитерминальной (каждый пользователь работает с компьютером с помощью своего терминала).

Не смотря на то, что Unix-подобные системы уступают по популярности Windows, они работают на больших типах компьютеров.

Linux представляет собой множество Unix-подобных операционных систем (дистрибутивов), которые чаще всего являются свободно распространяемыми.

Одной из уникальных особенностей систем GNU/Linux является отсутствие единого географического центра разработки. Linux и программы для нее пишутся миллионами программистов, рассредоточенных по всему миру.

Это операционная система также создавалась на основе ядра UNIX.

Является продукт компании Apple для ее же компьютеров Macintosh.

Считается надежной и удобной. Но в отличие от Windows не так популярна.

Современные операционные системы позволяют производить обмен данными как внутри одной программы, так и между различными программами. При этом процесс копирования или переноса осуществляется через Буфер обмена.

Буфер обмена – это область оперативной памяти, которая служит для временного хранения данных.

Данные в Буфер обмена можно поместить только один раз, а извлекать (вставлять) неограниченное число раз. В Буфере обмена могут храниться только данные, помещенные в него последней операцией Копировать (Вырезать). Каждая новая операция Копировать (Вырезать) удаляет старые данные из Буфера обмена и помещает в него новые.

Технология OLE (Object Linking and Embedding) - это технология встраивания и связывания объектов. Технология OLE включает следующие возможности:

1. Возможность совмещать в одном документе объекты с различными форматами данных (текст, рисунки, таблицы, базы данных и т.д.).
2. Приложение, содержащее встроенный или связанный объект, хранит сведения о формате данных этого объекта и приложениях работающих с ним.
3. Приложение, содержащее объект, осуществляет функции по отображению, перемещению и копированию объекта внутри документа и между приложениями. При этом поддерживается целостность объекта.
4. Возможность автоматически вызывать приложения для редактирования объектов или импортировать функции редактирования из этого приложения.

Технология OLE может быть использована в двух вариантах:

- установление связи с объектом;
- внедрение объекта.

Установление связи позволяет связывать данные из различных приложений таким образом, что данные одного приложения отображаются в другом, и при изменении данных в одном приложении они автоматически изменяются в другом.

Если данные, связанные с приложением невозможно представить в виде текста или рисунка, например, звук или видеофильм, то они отображаются в документе в виде пиктограммы (значка). Активизация данной связи (двойной щелчок на пиктограмме) приведет к проигрыванию звукового сигнала или видеоролика.

Для данных представимых в наглядном виде (текст, рисунок и пр.) активизация связи приведет к запуску приложения, в котором можно будет произвести редактирование объекта.

Связывание осуществляется следующим образом. Первоначально создается исходный документ. Затем с помощью некоторой программы создается объект, который должен быть помещен в исходный документ. Этот объект сохраняется в файле. Эта процедура обязательная, без нее невозможно установление связи. После этого в программе с исходным документом выполняется команда по связыванию объекта с документом.

Преимущества связывания:

- происходит автоматическое изменение объекта в документе, при изменении оригинала объекта;
- исходный документ имеет небольшой размер.

При внедрении объект существует только в единственном экземпляре и только там, где он внедрен, т.е. в самом документе (а не в отдельном файле, как при связывании). Для внедрения объекта можно использовать те же команды, что и при обмене данными, или использовать специальную команду для внедрения объекта.

Преимущества внедрения:

- легкая переносимость документа на другой компьютер;
- возможность редактирования объекта непосредственно в документе.

По типу пользовательского интерфейса различают текстовые (линейные), графические и речевые операционные системы.

Пользовательским интерфейсом называется набор приемов взаимодействия пользователя с приложением. Пользовательский интерфейс включает общение пользователя с приложением и язык общения.

Линейные операционные системы реализуют интерфейс командной строки. Основным устройством управления в них является клавиатура. Команда набирается на клавиатуре и отображается на экране дисплея. Окончанием ввода команды служит нажатие клавиши Enter. Для работы с операционными системами, имеющими текстовый интерфейс, необходимо овладеть командным языком данной среды, т.е. совокупностью команд, структура которых определяется синтаксисом

этого языка.

Первые настоящие операционные системы имели текстовый интерфейс. В настоящее время он также используется на серверах и компьютерах пользователей.

Такие операционные системы реализуют интерфейс, основанный на взаимодействии активных и пассивных графических экранных элементов управления. Устройствами управления в данном случае являются клавиатура и мышь. Активным элементом управления является указатель мыши — графический объект, перемещение которого на экране синхронизировано с перемещением мыши. Пассивные элементы управления — это графические элементы управления приложений (экранные кнопки, значки, переключатели, флажки, раскрывающиеся списки, строки меню и т.д.).

Примером исключительно графических ОС являются операционные системы семейства Windows. Стартовый экран подобных ОС представляет собой системный объект, называемый рабочим столом. Рабочий стол — это графическая среда, на которой отображаются объекты (файлы и каталоги) и элементы управления.

В графических операционных системах большинство операций можно выполнять многими различными способами, например через строку меню, через панель инструментов, через систему окон и др. Поскольку операции выполняются над объектом, предварительно он должен быть выбран (выделен).

Основу графического интерфейса пользователя составляет организованная система окон и других графических объектов, при создании которой разработчики стремятся к максимальной стандартизации всех элементов и приемов работы.

Окно — это обрамленная прямоугольная область на экране монитора, в которой отображаются приложения, документ, сообщение. Окно является активным, если с ним в данный момент работает пользователь. Все операции, выполняемые в графических ОС, происходят либо на Рабочем столе, либо в каком-либо окне.

В случае SILK-интерфейса (от англ. speech – речь, image – образ, language – язык, knowledge – знание) – на экране по речевой команде происходит перемещение от одних поисковых образов к другим.

Предполагается, что при использовании общественного интерфейса не нужно будет разбираться в меню. Экранные образы однозначно укажут дальнейший путь

перемещения от одних поисковых образов к другим по смысловым семантическим связям.

Текстовые редакторы — это программы, предназначенные для работы с текстом. Они позволяют читать, вводить, редактировать текст, сохранять его в файл, выполнять поиск по тексту и другое.

Главное отличие текстовых процессоров от текстовых редакторов: текстовые процессоры позволяют оформлять текст, а также вставлять нетекстовые объекты (изображения, таблицы, диаграммы, видео и аудио) в редактируемый документ. В текстовых процессорах оформление текста называется его форматированием.

Изначально, где-то в 70-е годы XX века, текстовыми процессорами называли специальные электронные устройства, предназначенные для работы с текстом. Позже, когда компьютеры стали более универсальными машинами, так стали называть программы-приложения.

Текстовые редакторы предназначены для работы с так называемым «плоским текстом». Такой текст не включает в себя элементы форматирования и нетекстовые объекты. В остальном по возможностям работы с текстом редакторы почти не уступают текстовым процессорам, а в чем-то превосходят их. Например, многие текстовые редакторы могут подсвечивать синтаксис текста, представляющего собой код на языке программирования. Причем подсветка кода работает для множества языков программирования.

Простейшие текстовые редакторы представляют собой одну программу. Более сложные могут включать дополнительные программы, библиотеки, утилиты. В таком случае текстовый редактор представляет собой уже не программу, а приложение, как и в случае с текстовыми процессорами.

Наиболее известным и простым текстовым редактором является «Блокнот», который поставляется вместе с операционной системой Windows. В GNU/Linux распространены более сложные текстовые редакторы. Их множество. В качестве примеров можно назвать более простой Gedit и более сложный Geany.

В определенном смысле текстовые процессоры можно считать разновидностью текстовых редакторов.

Текстовые процессоры позволяют форматировать (оформлять) текст и вставлять в него нетекстовые объекты (от таблиц и картинок до видеофайлов). Следовательно,



созданные в текстовых процессорах документы хранят не только текстовые данные, но также команды разметки текста на специальном языке, ссылки на связанные объекты или сами объекты в виде двоичных кодов.

Каждый текстовый процессор сохраняет документ в своем формате. Поэтому документ, созданный в одном приложении, может некорректно отобразиться, если его открыть в другом приложении.

Текстовые процессоры имеют более сложный интерфейс. Помимо строки меню, всегда есть панели инструментов, содержащие кнопки с пиктограммами, выпадающие списки и др. Нажатие на эти кнопки приводит к вызову определенных команд. Все возможные в текстовых процессорах команды содержатся в пунктах меню. На панелях инструментов обычно продублированы наиболее востребованные команды.

Видимость панелей инструментов настраивается. Так во многих текстовых процессорах по-умолчанию отображены только две панели «Стандартная» и «Форматирование».

У текстовых редакторов также могут быть панели инструментов. Однако они есть не у всех редакторов. В то время как панели инструментов есть у всех текстовых процессоров.

На сегодняшний день наиболее популярные текстовые процессоры входят в состав пакетов приложений. Так, MS Word входит в состав пакета MS Office, а LibreOffice Writer входит в состав пакета LibreOffice. Офисные пакеты приложений обычно включают текстовый и табличный процессор, приложение для создание презентаций, систему управления базами данных и др.

Потребность ввода графиков, схем, диаграмм, рисунков в текст или создание отдельного файла привела к появлению графических программ. Графические редакторы представляют собой программы и приложения, позволяющие создавать и модифицировать графические объекты.

Векторная графика создает изображения, составленные из различных геометрических фигур, которые описываются на специальном языке. Также векторные графические редакторы предоставляют возможность выбора толщины и цвета линий, заливки, шрифтов и др.

Существует также класс программ, позволяющих просматривать изображения в режиме слайдов, накладывать спецэффекты.

Деловая графика обеспечивает отображение информации, хранящейся в табличных процессорах, базах данных и других структурах в виде двух- или трехмерных графиков, круговой диаграммы, линейных графиков и др.

Информационные технологии научной графики предназначены для обслуживания задач оформления научных отчетов, содержащих математические, химические и прочие формулы, задач картографии и др.

Когнитивные компьютерные средства — это комплекс виртуальных устройств, программ и систем, реализующих комплексную обработку зрительной информации в виде образов, процессов, структур. Когнитивная графика позволяет представить в виде зрительных образов различные математические формулы и закономерности для доказательства сложных теорем, открывает новые возможности для познания законов функционирования сознания. Средства когнитивной графики связаны со многими новейшими информационными технологиями, включая гипертексты и мультимедиа.

Большинство графических процессоров удовлетворяет стандарту WIMP. Панель содержит меню действий, линейки инструментов и цветов.

Графические процессоры для обработки растровой графики предназначены для работы с фотографиями и включают в себя набор средств по кодированию фотоизображений в цифровую форму.

Графические процессоры для обработки векторной графики предназначены для профессиональной работы, связанной с художественной и технической иллюстрацией, с последующей цветной печатью. Они занимают промежуточное положение между пакетами для систем автоматизированного проектирования (САПР) и настольными издательскими системами (НИС).

Программные пакеты для работы с деловой графикой являются конструкторами графических образов деловой информации, призванными в наглядной и динамичной форме представлять результаты некоторого аналитического исследования. Такие программы достаточно просты в работе и интуитивно понятны.

Для подготовки буклетов, оформления журналов и книг предназначены специальные издательские системы. Они позволяют готовить их и печатать на принтерах или выводить на фотонаборные автоматы сложные документы высокого качества.

Настольные издательские системы (НИС) — это программы, предназначенные для профессиональной издательской деятельности, позволяющие осуществлять электронную верстку широкого спектра основных типов документов.

Предусмотренные в программных пакетах данного типа средства позволяют:

- компоновать (верстать) текст;
- использовать всевозможные шрифты и полиграфические изображения;
- осуществлять редактирование на уровне лучших текстовых процессоров;
- обрабатывать графические изображения;
- обеспечивать вывод документов высокого качества;
- и др.

Известными пакетами среди издательских систем для компьютеров являются PageMaker, QuarkXPress, Scribus и др.

Имеются два основных вида издательских систем. Издательские системы первого вида очень удобны для подготовки небольших материалов с иллюстрациями, графиками, диаграммами, различными шрифтами в тексте (например, газет, небольших журналов). Типичный пример такой системы — Aldus PageMaker.

Издательские системы второго вида больше подходят для подготовки объемных документов, например книг. Одна из таких систем — Ventura Publisher (Corel Ventura) — управляет меню и может считывать тексты, подготовленные с помощью других текстовых редакторов (например, Microsoft Word), сохраняя при этом параметры форматирования, заданные этим редакторами.

Основная операция издательских систем — верстка (размещение текста по страницам документа, вставка рисунков, оформление текста различными шрифтами и т.д.). Редактирование текста в издательских системах менее удобно, чем в текстовых редакторах. Поэтому бывает, что документы готовят в два этапа: сначала набирают текст в текстовом процессоре, а затем считывают его издательской системой и осуществляют окончательную подготовку документа.

Основные функции издательских систем: использование сотен видов шрифтов (начертаний и размеров символов текста), которые отображаются на экране так же, как при печати; изменение и корректировка рисунков и диаграмм; формирование таблиц; выравнивания; работа с формулами и др.

Большинству пользователей для выполнения издательских работ может быть вполне достаточно возможностей текстового процессора, в котором есть элементы цветовыделения и средства графических редакторов.

Множество задач носят учетно-аналитический характер и требуют табличной компоновки данных с подведением итогов по различным группам и разделам данных, например при составлении баланса, справок для налоговых органов, всевозможных финансовых отчетов и т.п. Для хранения и обработки информации, представленной в табличной форме, используются электронные таблицы.

Программные средства для проектирования электронных таблиц называют также табличными процессорами. Они позволяют не только создавать таблицы, но и автоматизировать обработку табличных данных. Кроме того, с помощью электронных таблиц можно выполнять различные экономические, бухгалтерские и инженерные расчеты, а также строить разного рода диаграммы, проводить сложный экономический анализ, моделировать и оптимизировать решение различных хозяйственных операций и многое другое.

Функции табличных процессоров весьма разнообразны и включают:

- создание и редактирование электронных таблиц;
- их оформление и печать;
- создание многотабличных документов, объединенных формулами;
- построение диаграмм, их модификацию и решение экономических задач графическими методами;
- работу с электронными таблицами как с базами данных (сортировка таблиц, выборка данных по запросам);
- создание итоговых и сводных таблиц;
- использование при построении таблиц информации из внешних баз данных;
- решение прогностических экономических задач путем подбора параметров;
- решение оптимизационных задач;
- статистическая обработка данных;
- разработка макрокоманд и др.

Табличные процессоры различаются в основном набором выполняемых функций и интерфейсом.

Электронная таблица представляет собой двумерный массив строк и столбцов, одни из которых обозначаются цифрами, а другие — буквами. Место пересечения строки и столбца называется ячейкой. Адресом (идентификатором) ячейки служат буква, указывающая столбец, и цифра, указывающая номер строки.

Существует абсолютная и относительная адресация ячеек. Абсолютная адресация применяется более широко. При относительной адресации в верхней строке состояния указывается приращение со знаком от начала искомой ячейки.

Система управления базами данных или сокращенно СУБД - это программное обеспечение объединенное со специализированным языком, предназначенное для работы с базами данных.

База данных - это какие-либо данные определенной тематики, каким-либо образом организованные и хранящиеся во внешней памяти компьютера (например, на жестком диске). Эти данные могут обновляться, извлекаться, осуществляться по ним поиск и другие операции. Основная цель организации баз данных - это удобство манипуляции с данными. Именно это "удобство" и должна обеспечивать СУБД.

СУБД бывают разными. Кроме того, специалисты по отношению к ним могут быть трех типов:

- Разработчики СУБД.
- Разработчики баз данных на конкретных СУБД.
- Пользователи баз данных и СУБД.

Пользователи, пользуясь СУБД, вносят в базы данных информацию, ищут ее, выполняют различные запросы и др. Другими словами, пользователи используют базу данных по ее прямому назначению, а программным обеспечением, с помощью которого они работают, выступает СУБД.

Ряд СУБД представляет собой приложения, с которыми можно работать, не зная языка программирования для баз данных. Примерами таких приложений являются Microsoft Access и LibreOffice Base. Другие СУБД предполагают более серьезный уровень профессиональной подготовки. Примером такой СУБД является MySQL.

СУБД классифицируются по ряду признаков: по модели данных (реляционные, иерархические и др), по способу доступа. Реляционные базы данных на сегодняшний день являются наиболее распространенными.

Реляционная система управления базами данных должна позволять создавать таблицы с данными, создавать запросы, вносить данные и изменения и др.

## **2.3 Особенности программного обеспечения промышленных компьютеров**

Специфика промышленных применений наложила свой отпечаток и на используемое программное обеспечение промышленных компьютеров.

Первым требованием является надежность программного обеспечения. Действительно, одно дело, когда «зависает» редактор текста в офисе, а другое дело, когда неправильно работает программа, управляющая ядерным реактором или космическим кораблем. В конечном счете, многие новации последнего времени, типа структурного программирования, объектно-ориентированных языков появились в результате стремления писать все более сложные программы с меньшим количеством ошибок.

Вторым требованием является быстрое реагирование на какие-либо внешние события или изменения в параметрах управляемых процессов. Системы, работающие в соответствии со вторым требованием, называются системами реального времени. Разумеется, понятие «быстрый» является относительным. Типовое время реагирования на внешние воздействия, необходимое современным промышленным системам, составляет десятки микросекунд. В то же время существует много задач, где инерционность протекающих процессов позволяет реагировать с задержкой в сотни миллисекунд. Строго говоря, отождествление понятий «реальное время» и «быстрый» не является верным. Например, если ваша система регистрирует уровень грунтовых вод, то и одно измерение в час соответствует требованиям реального времени для этого процесса.

Классическое и строгое определение понятия работы системы в реальном масштабе времени вытекает из требований теоремы Котельникова, которая определяет максимально допустимый период считывания информации о процессе и реагирования на изменения в этом процессе, исходя из наивысшей скорости изменения процесса (частотных свойств измеряемого сигнала - ширины его

спектра).

И третьим требованием, часто предъявляемым к программному обеспечению систем управления, является многозадачность. Это требование проистекает из подчас чрезвычайно сложной и многоуровневой природы управляемых процессов, когда необходимо одновременно реализовать сложные алгоритмы управления различными аспектами или частями реального объекта. Каждая задача выполняет свою долю работы по управлению объектом, и все они делят между собой ресурсы вычислительной системы в зависимости от своего приоритета, от внешних и внутренних событий, связанных с конкретной задачей. Другим тесно связанным с многозадачностью понятием является многопоточность. Хотя терминология здесь не совсем устоялась, под многопоточностью, как правило, понимают возможность выполнения в рамках одной задачи нескольких независимых процессов (поток команд), которые, в отличие от задач, пользуются общими участками кода и данных[7].

Названным требованиям должны удовлетворять все уровни программного обеспечения системы:

1. Базовая система ввода/вывода (BIOS);
2. Операционная система и драйверы (ОС);
3. Собственно прикладные программы.

BIOS осуществляет непосредственный интерфейс между аппаратурой и программным обеспечением верхних уровней. Основная опасность при обращении к BIOS - это возможность запрета прерываний на достаточно долгое время, в результате чего может быть пропущена важная информация от быстродействующих датчиков или телекоммуникационных устройств. Вот почему, с одной стороны, есть фирмы, предлагающие BIOS, ориентированные на приложения «жесткого» реального времени, а с другой стороны, многие операционные системы минимизируют взаимодействие с BIOS или не обращаются к ней вообще.

Операционная система выполняет базовые функции по интерфейсу с оператором, запуску программ, распределению памяти, поддержке файловой системы и т. п. Сегодня существует широкий выбор операционных систем, разработанных специально для применения в системах "жесткого" реального времени. Эти ОС часто так и называют операционными системами реального времени. Среди них наиболее известны такие ОС, как QNX, OS 9000, OnTime, VxWorks, iRMX, VRTX,

Nucleus и другие.

Важной проблемой использования программного обеспечения является его переносимость. Существует два подхода к переносимости программного обеспечения: первый - это поддержка одной ОС одновременно нескольких аппаратных платформ, второй - это обеспечение стандартного интерфейса между прикладными программами и ОС.

При проектировании ОС реального времени все чаще используется идеология микроядра, которая увеличивает надежность программного обеспечения и позволяет использовать только те компоненты операционной системы, которые необходимы в каждом конкретном случае. Например, микроядро одной из самых распространенных операционных систем QNX имеет размер менее 10 Кбайт. Модуль, ответственный за файловую систему, запускается как одна из задач и может быть легко удален. Все драйверы также функционируют как независимые задачи. То есть, если в вашей встроенной системе не используются файловые операции или отсутствуют интерфейсы с какими-либо внешними устройствами, вы можете просто не включать в состав операционной системы, функционирующей в вашем изделии, соответствующие модули, высвобождая тем самым память для более эффективного выполнения приложений[8].

Разработчики программного системного обеспечения поставляют не просто ядро операционной системы, а функционально законченный комплекс средств разработки и выполнения приложений реального времени.

В качестве примера приведем состав комплекса OnTimeRTOS-32 для x86 совместимых систем. Он состоит из пяти компонентов:

- RTTarget-32 - компактная операционная система, включающая все средства для запуска и выполнения приложений Win32, созданных стандартными системами разработки для Windows;
- RTKernel-32 - быстрый и компактный планировщик задач реального времени;
- RTFiles-32 - файловая система для 32-разрядных x86 совместимых систем обеспечивает файловый ввод/вывод в реальном масштабе времени;
- RTIP-32 - определяет сетевые возможности RTOS. Компонент содержит TCP/IP протоколы, необходимые для Ethernet и последовательных коммуникаций;
- RTPEG-32 - графическая объектно-ориентированная библиотека C++ содержит полный набор элементов для создания интерфейсов в стиле Windows-приложений.



Применение OnTimeRTOS-32 обеспечивает время переключения между задачами в 1 мкс.

Немного особняком стоит программное обеспечение, предназначенное для эксплуатации на верхнем уровне систем автоматизации, например, в диспетчерских пультах управления сложными агрегатами. Главными функциями таких программ (они получили общее наименование SCADA- систем - Supervisor Control And Data Acquisition) являются отображение технологического процесса в виде мнемосхем, сигнализация об аварийных ситуациях, ведение системного журнала, обеспечение общего управления процессом со стороны оператора и т. п. Многие подразделения АСУ крупных предприятий, как правило, имеют свои наработки в этой области. В то же время есть фирмы, специализирующиеся на разработке универсальных

SCADA программ, таких как Genesis (Iconics), FixDmacs (Intellution), In Touch (Wonderware), TraceMode (Adastr Research Group)[\[9\]](#).

В последнее время получает распространение программное обеспечение класса MES (Manufacturing Execution Systems), основной задачей которого является оптимизация управления производством в целом, включая планирование запасов комплектующих, расписание технологических процессов и т.д. Системы MES, с одной стороны, взаимодействуют с системами SCADA, образуя единую систему управления производством, с другой стороны, они часто интегрируются в систему планирования и управления предприятием в целом.

## **Заключение**

Программное обеспечение – это программа, которая управляет работой компьютера или выполняет какие либо расчеты или действия. Это могут быть внутренние команды, управляющие оборудованием или программа, выполняющая какие либо действия в ответ на вводимые с клавиатуры команды. Программное обеспечение ПК может быть с открытым исходным кодом или являться собственностью компании разработчика.

Между программным и аппаратным обеспечением есть существенна разница. Аппаратные средства материальны, т.е. являются частью компьютера, и в большинстве случаев выполняют физическую работу, в то время, как программное обеспечение управляет работой аппаратных средств. Компьютер можно

адаптировать к выполнению новых задач установкой нового оборудования. Аппаратное обеспечение включает в себя монитор, центральный процессор, клавиатуру, мышь. Программное обеспечение включает в себя различные программы для обработки текста, операционные системы, игры и другое.

Есть два основных типа программного обеспечения: системные программы и приложения. Первый тип используется для управления аппаратными средствами, в то время как второй тип используется для всех остальных задач. К системному программному обеспечению относятся операционные системы, такие как Windows или Linux, драйвера, которые позволяют компьютеру взаимодействовать с другими устройствами, как принтеры или управлять внутренними аппаратными средствами. Есть множество различных видов приложений, такие как игры, медиаплееры, текстовые редакторы, антивирусные программы и приложения для создания новых программ.

Аппаратные средства понимают только машинный код, который представлен в двоичной системе исчисления. Программное обеспечение в некотором смысле выступает переводчиком между человеком и аппаратными средствами, преобразуя все команды человека в бинарный код. Для создания программы программисты пишут исходный код программы, состоящий из читабельных команд, которые затем другой программой, компилятором, преобразуются в двоичный машинный код. В результате этих действий получается компьютерная программа.

Созданные программистами программы могут распространяться как с открытым исходным кодом, т.е. с исходным текстом программы, который можно отредактировать, так и без него, скрывая исходный текст программы. Программы с открытым исходным кодом в основном распространяются бесплатно и могут быть доработаны и исправлены любым человеком, в то время, как программы защищенные авторским правом, и распространяемые без исходного текста, могут исправить или переделать только авторы программы или компании разработчики этой программы.

## **Список литературы**

1. Гашков С.Б., Применко Э.А., Черепнев М.А. Криптографические методы защиты информации. – М.: Академия, 2015. – 304 с.
2. Грибунин В.Г., Чудовский В.В. Комплексная система защиты информации на предприятии. – М.: Академия, 2012. – 416 с.

3. Гришина Н.В. Комплексная система защиты информации на предприятии. – М.: Форум, 2015. – 240 с.
4. Емельянова Н.З., Партыка Т.Л., Попов И.И. Защита информации в персональном компьютере. – М.: Форум, 2012. – 368 с.
5. Защита информации в системах мобильной связи. Учебное пособие. – М.: Горячая Линия - Телеком, 2015. – 176 с.
6. Комплексная система защиты информации на предприятии. Часть 1. – М.: Московская Финансово-Юридическая Академия, 2012. – 124 с.
7. Корнеев И.К, Степанов Е.А. Защита информации в офисе. – М.: ТК Велби, Проспект, 2014. – 336 с.
8. Максименко В.Н., Афанасьев, В.В. Волков Н.В. Защита информации в сетях сотовой подвижной связи. – М.: Горячая Линия - Телеком, 2014. – 360 с.
9. Малюк А.А, Пазизин С.В, Погожин Н.С. Введение в защиту информации в автоматизированных системах. – М.: Горячая Линия - Телеком, 2011. – 146 с.
10. Малюк А.А. Информационная безопасность. Концептуальные и методологические основы защиты информации. Учебное пособие. – М.: Горячая Линия - Телеком, 2014. – 280 с.
11. Маньков В.Д, Заграничный С.Ф. Методические рекомендации по изучению "Инструкции по применению и испытанию средств защиты, используемых в электроустановках". – М.: НОУ ДПО "УМИТЦ "Электро Сервис", 2011. – 132 с.
12. Петраков А.В. Основы практической защиты информации. Учебное пособие. – М.: Солон-Пресс, 2015. – 384 с.
13. Северин В.А. Комплексная защита информации на предприятии. – М.: Городец, 2012. – 368 с.
14. Сурис М.А., Липовских В.М. Защита трубопроводов тепловых сетей от наружной коррозии. – М.: Энергоатомиздат, 2013. – 216 с.
15. Хорев П.Б. Методы и средства защиты информации в компьютерных системах. – М.: Академия, 2012. – 256 с.
16. Хорев П.Б. Программно-аппаратная защита информации. – М.: Форум, 2012. – 352 с.
17. Шаньгин В.Ф. Комплексная защита информации в корпоративных системах. – М.: Форум, Инфра-М, 2015. – 592 с.
18. Петров, И.В. Стандартные языки и приемы прикладного программирования [Текст] / И.В. Петров. – М.: СОЛОН-Пресс, 2014. – 230 с.
19. Целищев, Е. AutomatiCS 2011: разрабатывать КИПиА просто и эффективно. Часть V. Подключение к многоканальным приборам [Текст] / Е. Целищев, А. Глянцева // САПР и Графика. – 2012. – № 2. – С. 76-78.

20. Шалыто, А.А. SWITCH-технология автоматный подход к созданию программного обеспечения «реактивных» систем [Текст] / А.А. Шалыто, Н.И. Туккель // Программирование. - 2013. - № 2. - С. 88-99.
1. Гашков С.Б., Применко Э.А., Черепнев М.А. Криптографические методы защиты информации. - М.: Академия, 2015. - с.34 [↑](#)
2. Защита информации в системах мобильной связи. Учебное пособие. - М.: Горячая Линия - Телеком, 2015. - с.75 [↑](#)
3. Максименко В.Н., Афанасьев, В.В. Волков Н.В. Защита информации в сетях сотовой подвижной связи. - М.: Горячая Линия - Телеком, 2014. - с.73 [↑](#)
4. Малюк А.А. Информационная безопасность. Концептуальные и методологические основы защиты информации. Учебное пособие. - М.: Горячая Линия - Телеком, 2014. - с.14 [↑](#)
5. Маньков В.Д, Заграничный С.Ф. Методические рекомендации по изучению "Инструкции по применению и испытанию средств защиты, используемых в электроустановках". - М.: НОУ ДПО "УМИТЦ "Электро Сервис", 2011. - с.75 [↑](#)
6. Малюк А.А, Пазизин С.В, Погожин Н.С. Введение в защиту информации в автоматизированных системах. - М.: Горячая Линия - Телеком, 2011. - с.45 [↑](#)
7. Шалыто, А.А. SWITCH-технология автоматный подход к созданию программного обеспечения «реактивных» систем [Текст] / А.А. Шалыто, Н.И. Туккель // Программирование. - 2013. - № 2. - С. 88-99. [↑](#)
8. Целищев, Е. AutomatiCS 2011: разрабатывать КИПиА просто и эффективно. Часть V. Подключение к многоканальным приборам [Текст] / Е. Целищев, А. Глянцева // САПР и Графика. - 2012. - № 2. - С. 76-78. [↑](#)
9. Петров, И.В. Стандартные языки и приемы прикладного программирования [Текст] / И.В. Петров. - М.: СОЛОН-Пресс, 2014. - с.72 [↑](#)