

Владимирский государственный университет  
*Кафедра АТП*

**ИНТЕГРИРОВАННЫЕ СИСТЕМЫ  
ПРОЕКТИРОВАНИЯ И УПРАВЛЕНИЯ**

*Конспект лекций*

**ОГЛАВЛЕНИЕ..... ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.**

<b>ОСНОВНЫЕ ПОНЯТИЯ АСУ. ....</b>	<b>4</b>
Предпосылки создания АСУ.....	4
Основные понятия АСУ. ....	4
Принципы управления. ....	5
Функции управления. ....	6
Организация структур управления.....	7
<b>ОСНОВЫ СОВРЕМЕННОГО ПРОИЗВОДСТВА. ....</b>	<b>9</b>
Типы производств. ....	9
Календарно-плановые нормативы ( серийное производство ). ....	10
Административная структура производства. ....	13
Планирование производства. ....	13
<i>Основные понятия.....</i>	<i>13</i>
<i>Основные системы планирования.....</i>	<i>15</i>
<i>Переменные системы управления производством. ....</i>	<i>17</i>
<b>ПОНЯТИЕ АВТОМАТИЗИРОВАННЫХ СИСТЕМ УПРАВЛЕНИЯ И ИХ КЛАССИФИКАЦИЯ.....</b>	<b>19</b>
Определение АСУ.....	19
Понятие большой и сложной системы. ....	19
Классификация АСУ.....	19
Принципы построения АСУ. ....	21
<b>СТРУКТУРА АСУ.....</b>	<b>23</b>
Общая схема структуры АСУ.....	23
Техническое обеспечение АСУ.....	24
<i>Общая схема технического обеспечения АСУ.....</i>	<i>24</i>
<i>Подсистема сбора и регистрации информации. ....</i>	<i>24</i>
Состав.....	24
Датчики. ....	24
Преобразователи информации.....	27
<i>Подсистема приема и передачи информации. ....</i>	<i>28</i>
<i>Подсистема хранения и обработки информации. ....</i>	<i>28</i>
Подсистема обработки информации.....	28
Подсистема хранения информации. ....	31
<i>Подсистема вывода и отображения информации. ....</i>	<i>34</i>
Мониторы. ....	34
Защитные экраны.....	34
Принтеры. ....	35
Плоттеры (графопостроители). ....	39
<i>Подсистема ввода информации.....</i>	<i>41</i>
Мышь. ....	41
Trackball. ....	42
Дигитайзер. ....	42
Сканер. ....	42
МАТЕМАТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ АСУ. ....	43
<i>Моделирование систем. ....</i>	<i>44</i>
Основные понятия.....	44
Этапы моделирования.....	44
Классификация моделей. ....	44
Основные принципы моделирования. ....	46
<i>Методы оптимизации. ....</i>	<i>48</i>
Основные понятия.....	48
Методы оптимизации. ....	49

<i>Программное обеспечение</i> .....	50
Классификация ПО .....	50
Обслуживающие программы .....	50
Характеристики ОС .....	51
III. РВ .....	55
IV. РАФОС .....	58
<i>Пользовательские программы</i> .....	60
<b>ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ</b> .....	61
<i>Общие понятия и состав</i> .....	61
<i>Внешнее информационное обеспечение</i> .....	61
Состав .....	61
Системы кодирования .....	62
Системы информации .....	63
<i>Внутреннее информационное обеспечение</i> .....	63
Состав .....	63
Требования к информации .....	63
Физическая структура размещения информации .....	65
Логические структуры .....	65
Структуры баз данных .....	67
Банки данных .....	70
<b>ОРГАНИЗАЦИОННОЕ ОБЕСПЕЧЕНИЕ АСУ</b> .....	70
<i>Введение</i> .....	70
<i>Функции заказчика и подрядчика</i> .....	70
<i>Функции исполнителя (разработчика)</i> .....	71
<b>ЮРИДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ АСУ</b> .....	71
<i>Введение</i> .....	72
<i>Оформление документов договорной деятельности при разработке</i> .....	72
Основные понятия .....	72
Договор состоит из разделов .....	72
Содержание технического задания .....	73
<b>ЭЛЕМЕНТЫ ПРОЕКТИРОВАНИЯ</b> .....	74
Стадии проектирования .....	74
Содержание технического проекта .....	74
Правила рисования информационных систем .....	75
<b>ЭКЗАМЕНАЦИОННЫЕ ВОПРОСЫ</b> .....	79

# ***Основные понятия АСУ.***

## ***Предпосылки создания АСУ.***

Одна из важнейших предпосылок лежит в сфере ***военного противоборства***. Времена когда управляет боем командир прошли. Человек уже не в состоянии достаточно быстро оценить обстановку с учетом всех известных данных. В качестве примера можно привести войну Аргентины с Англией за Мальвийские о-ва в 70-х годах. В этой войне был случай когда по флагману Английской эскадры под прикрытием истребителей с самолета была выпущена крылатая ракета которую практически невозможно засечь из-за того что она очень низко летит, а если ее и замечают, то человек в течении секунд которые у него есть не успевает принять решение. Но радары корабля заметив в последний момент ракету сообщили об этом в вычислительный центр и компьютер принял верное решение о развороте судна. Правда двигательная установка на корабле тогда частично отказала и ракета попала в вычислительный центр. Но этот случай показывает то, что тот, кто имеет лучше компьютеры и систему АСУ в целом, тот и побеждает в войне.

Другая предпосылка лежит в сфере ***экономического развития*** страны. Дело в том что экономическая развитость страны зависит от множества факторов и далеко не последним из них является качество управления. В других странах уже давно все изменения в экономике и переходы между экономическими моделями сначала моделируются, благодаря чему удается избежать грубых ошибок в управлении страной. (Не следует это понимать как то, что внедрение АСУ очень выгодная вещь. Дело в том, что его внедрение может обойтись очень дорого, а существенного увеличения прибыли не будет. Но АСУ всегда внедряется прежде всего потому, что это ***удобно*** заказчику).

Еще одна предпосылка внедрения АСУ лежит в сфере ***культуры***.

В нашей стране необходимость внедрения АСУ заметил академик ***Глушков*** и ему удалось добиться постановления съезда о внедрении АСУ. В принципе, он является основателем науки в нашей стране.

## ***Основные понятия АСУ.***

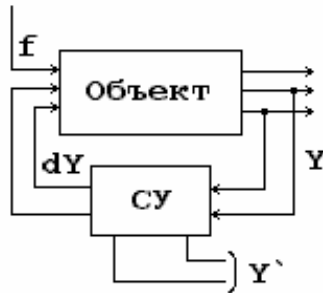
***АСУ*** - это автоматизированные системы управления. Разберем эти понятия.

***Автоматизированная система*** - это человеко-машинная система, т.е. человек - соавтор в выборе решений. Данная система создается и внедряется при невозможности иметь в данном случае ***автоматическую*** систему. То есть, автоматическая система не может отследить все возможные состояния объекта или присутствуют какие-либо помехи, которые данная система не в состоянии обработать.

Вообще, под ***системой*** понимают совокупность взаимосвязанных элементов обладающих некоторыми свойствами ***единства*** и ***целостности***.

Система всегда существует и при введении или исчезновении хотя бы одного из элементов система умирает или появляется новая система, - в этом заключается свойство *единства*. А свойство *целостности* заключается в том, что свойства системы в принципе не сводятся к совокупности свойств объектов.

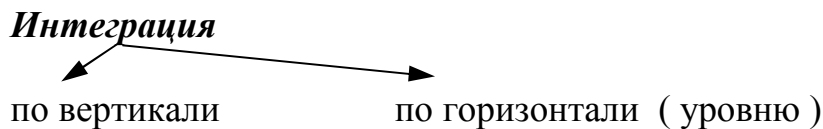
Под *управлением* понимают целенаправленное воздействие на объект для достижения им желаемого состояния (характеристик).



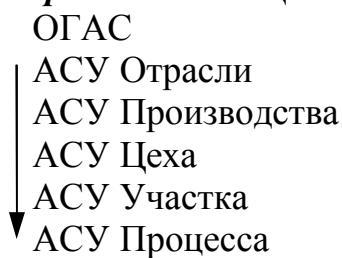
Под объектом понимаем все что угодно. Те воздействия которые заставляют объект менять состояние называют *входными*, а те переменные, которые характеризуют состояние объекта называют *выходными*. Часть воздействий на объект играют роль целенаправленных воздействий, - эту часть называют *управлением*. Часть воздействий мешают управлению, - их называют *возмущающими воздействиями*.

Часть переменных мы можем наблюдать, - они называются наблюдаемыми. На основании наблюдаемых переменных ( $Y$ ) и желаемых результатов ( $Y'$ ) вырабатывается *управляющее воздействие* ( $dY = Y' - Y$ ).

Иногда говорят про Интегрированные Системы Автоматического Управления (ИСАУ). Здесь нераскрытым словом является интеграция. Под *интеграцией* понимают взаимосвязанную деятельность разнородных подсистем.

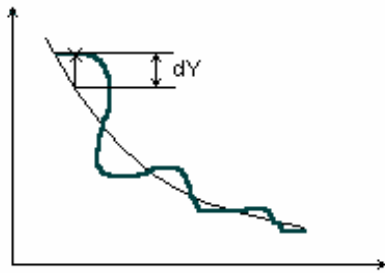


**Пример: ИНТЕГРАЦИЯ ПО ВЕРТИКАЛИ:**



### **Принципы управления.**

1. Управление *по отклонению*.



Управляющее воздействие вырабатывается на основе анализа отклонения от заданной траектории.

Различают два вида управления по отклонению, - это так называемые алгоритм мягкой и жесткой посадки.

Алгоритм жесткой посадки состоит в том, что при каждом отклонении рассчитывается такое воздействие чтобы систему вернуть на первоначальную расчетную траекторию.

Алгоритм мягкой посадки состоит в том, что при каждом отклонении происходит расчет новой траектории для достижения заданной цели.

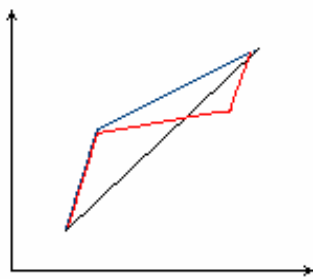


Рис.1

Применение того или иного алгоритма оправдано в разных случаях. Например когда корабль движется в море, он естественно отклоняется от первоначальной траектории. Применение в таком случае алгоритма жесткой посадки крайне нецелесообразно (смотри рис. 1). Красным цветом на этом рисунке показана траектория корабля при применении алгоритма

жесткой посадки ( корабль возвращаясь на траекторию, - проскакивает ее в силу инертности). Синим цветом на том же рисунке показана траектория корабля при применении алгоритма мягкой посадки. Как видно в данном случае применение алгоритма мягкой посадки позволяет сократить время движения, расход топлива и т.п. В другом случае когда самолет заходит на посадку применение алгоритма мягкой посадки (постоянный перерасчет траектории) может привести к тому что самолет должен будет садиться по вертикали вниз (смотри рис.2). В данном случае необходимо использовать алгоритм жесткой посадки.

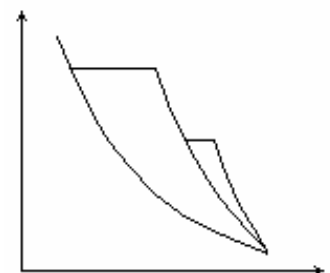


Рис.2

## 2. Управление *по возмущению*.

Иногда управляя по возмущению мы выигрываем во времени и можем избежать отклонения. Например, если человека кусает ядовитая змея и дав пострадавшему противоядие (не дожидаясь отклонения), мы тем самым можем избежать этого отклонения.

### **Функции управления.**

1. **Планирование.** Сводится к расчету заданной траектории (плана). Существует два ранее рассмотренных варианта движения объекта к заданной цели. Это алгоритм мягкой и жесткой посадки.
2. **Учет.** Это измерение фактического состояния объекта управления.

3. **Контроль.** Сопоставление фактического состояния объекта с заданным и вычисление отклонений.
4. **Анализ.** Под анализом понимается исследование отклонений в поведении объекта и выработка вариантов управляющих воздействий.
5. **Регулирование.**
  - выбор управленческого решения;
  - принятие решения;
  - реализация управляющего воздействия.

**Автоматизированной системой** - называется человеко-машинная система в которой на основе равноправного соавторства человек и компьютер выполняют функции управления.

### **Организация структур управления.**

Выделяют несколько видов связей, которые присутствуют между элементами в структурах управления.

**Линейные** связи - связи между элементами чисто административные.

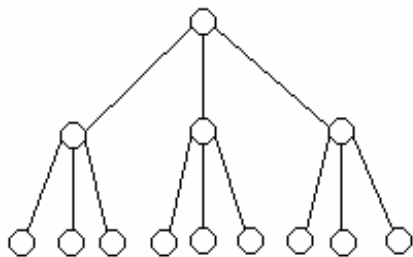
**Функциональные** связи - связи в рамках выполнения определенных функций (гл. бухгалтер - бухгалтер).

**Информационные** связи - связи по обмену информацией по одному уровню.

Выделяют несколько видов структур управления. Рассмотрим их ниже.

#### 1. **Линейная** структура управления.

Самая простая структура, - когда присутствуют только административные связи ( иерархическая структура ). У каждого элемента нижнего уровня только один элемент верхнего уровня, а каждый элемент верхнего уровня имеет хотя бы два элемента нижнего уровня, т.е. в этой системе отсутствуют функциональные и информационные связи. Эта структура практически никогда не используется.



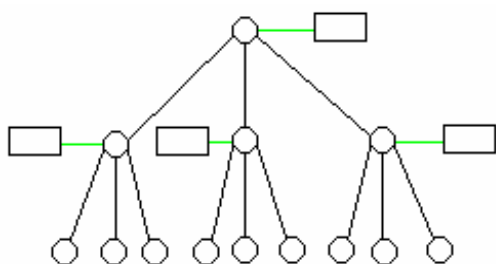
Пример: построение иерархии церкви, но до какого-то уровня.

«+» Легко решаются вопросы субординации, персональная ответственность каждого за его область деятельности.

«-» Каждый вышестоящий элемент должен полностью быть в курсе всех событий и являться специалистом во всех областях касающихся деятельности.

#### 2. **Линейно - штабная** структура.

**Штаб** - совет специалистов, по разным узким областям, они всего лишь советники и ответственности не несут.



У каждого элемента верхнего уровня появляется советующий орган (штаб).

«+» Возможность принятия более квалифицированного решения.

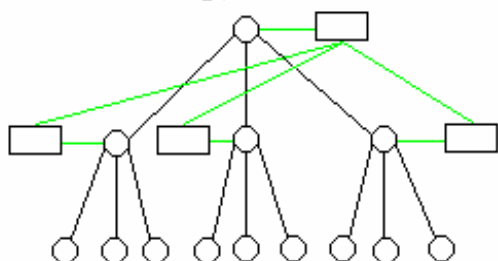
«+» Вопросы субординации решаются также легко как в линейной структуре.

«-» Советующий орган ни за что ответственности не несет (не перед кем не отчитывается).

— - Административные связи

— - Функциональные связи

### 3. Линейно - функциональная структура.



Это наиболее распространенный вид структуры.

«+» Появилась возможность закрепить персональную ответственность с низу до верху в рамках выполнения каждой функций.

«-» У нижестоящих элементов в

рамках разных функций появилось много начальников и вопросы субординации усложнились.

«+» Высокая загрузка персонала и оборудования.

### 4. Матричная структура.

По таблице видно кому чьи услуги требуются на 1 апреля.

Апрель, 1	Машинистка	Копировальщик	Конструктор	Расчетчик
Руководитель А	+			
Руководитель Б		+	+	+
Руководитель В	+			
Руководитель Г			+	

Иногда специалистов объединяют в бюро (бюро машинисток, бюро копировальщиков и т.п.) в бюро свой начальник. Знаком "+" отмечены те руководители, кому требуется услуги специалистов на данное число.

«+» Позволяет максимально загрузить, использовать трудовые ресурсы.

«-» Чрезвычайно сложны вопросы субординации.

**Особенность:** система может существовать только в высокоорганизованных структурах НИИ, конструкторских бюро и т.п.



# ***Основы современного производства.***

## ***Типы производств.***

Производство бывает трех типов:

- ***дискретное*** - в каждую единицу времени выпускается целое количество изделий;
- ***непрерывное*** - в каждый момент времени продукция может выражаться некоторым целым числом;
- ***дискретно-непрерывное*** - одна величина непрерывна а другая дискретна: кабель - бухты, молоко - бутылки, бумага - рулоны и т.п.

Большинство предприятий дискретного типа ( в Перми, т.к. большинство предприятий - предприятия ВПК ).

Дискретные предприятия делят по коэффициенту серийности на:

- ***массовое*** производство
- ***серийное*** производство
- ***единичное*** производство.

***Коэффициент серийности*** - определяет сколько технологических операций (в среднем) выполняется на одном станке (количество операций делится на количество оборудования).

### 1. ***Массовое*** производство (коэффициент серийности = 1).

Номенклатура изделий стабильна и ограничена, оборудование специализировано и уникально (на каждом станке выполняется одна операция).

«+» Самые маленькие затраты, высокая технология.

«-» Монотонность ритма, отсутствие творчества в работе, очень высокие затраты на переход с одного выпускаемого изделия на другое.

### 2. ***Единичное*** производство.

Требуется универсальное оборудование, которое позволяет выполнять любые виды работ, персонал с высокой квалификацией.

«+» Отдельные образцы удается сделать качественнее, чем в массовом производстве.

«-» Дорогое оборудование, высокая себестоимость изделий, отсутствие сложившейся технологии.

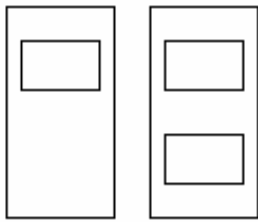
### 3. ***Серийное*** производство.

Различают: ***крупносерийное***, ***среднесерийное*** и ***мелкосерийное*** производство. На одном и том же оборудовании выполняется несколько технологических операций. Оборудование время от времени перенастраивается, что позволяет изготавливать детали партиями.

«+» Сокращается время переналадки, увеличивается концентрация производственного оборудования и номенклатура изделий.

«-» Сильно усложняется управление производством.

Цеха обычно объединяют в производства. Цеха бывают специализированные по технологическому признаку и по номенклатуре.



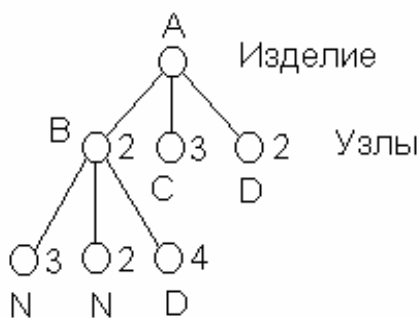
- По технологии (кузнечный, токарный, термический цеха).
- По номенклатуре (изготовление шестерен, валов, лопаток).

Детали изготавливают по технологии состоящей из трех стадий :

- Заготовительная (: заготовка)
- Механообрабатывающая (: деталь)
- Сборочная стадия (: узел, изделие).

**Изделие** - готовая продукция.

В изделие обычно входят узлы и подузлы в узлы входят детали а детали изготавливаются из заготовок.



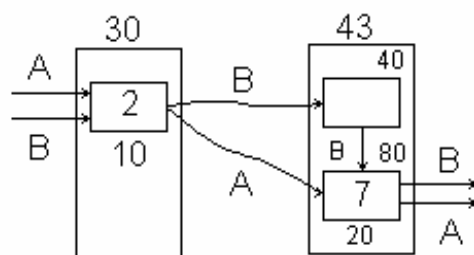
**Применяемость** - количество деталей применяемых при сборке узла.

Применяемость различают прямую и полную. **Прямая** это сколько деталей нужно для сборки данного узла а **полная** это сколько нужно всего.

Все современное производство имеет маршрутную технологию обработки.

**Технологический маршрут** - совокупность технологических операций, необходимых для изготовления готовой детали.

**Технологическая операция** - совокупность технологических переходов, выполняемых на одном оборудовании без смены режущего инструмента (токарная, фрезерная).



**Технологическая линия** - совокупность оборудования на технологическом маршруте.

**Переход** - совокупность элементарных действий без смены направления обработки

(проточка отверстия).

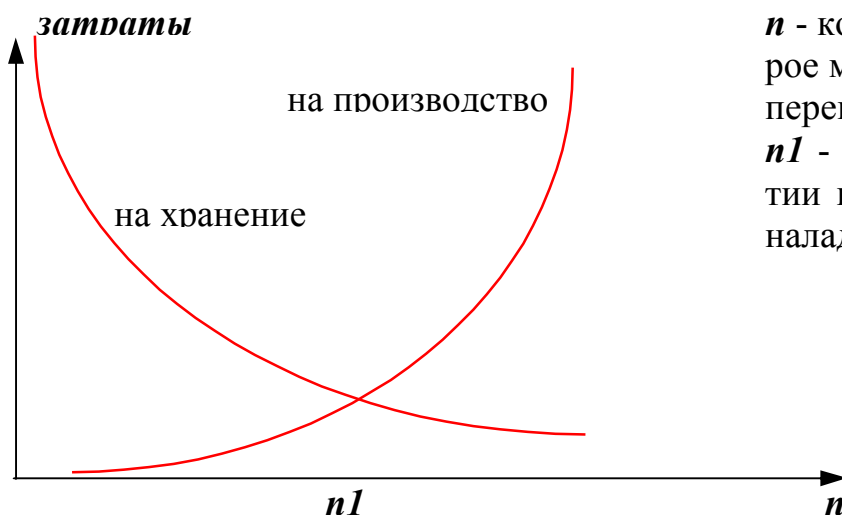
**Элементарные действия** - установка, снятие детали и т.д.

### **Календарно-плановые нормативы (серийное производство).**

Это нормативы, которые привязаны к определенному сроку. К ним относится, например, размер партии. На их основе составляется график работ. Плановые периоды: год, месяц, декада, неделя.

### 1. *Размер партии деталей.*

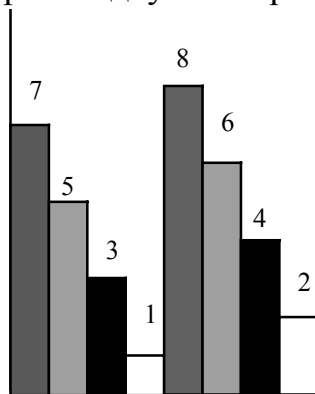
При увеличении размера партии уменьшаются удельные затраты на производство одной детали, но растут затраты на их хранение ( складские помещения, краны, АСУ учета и т.д.).



$n$  - количество деталей которое мы производим с одной переналадки станка.  
 **$n1$**  - оптимальный размер партии исходя из затрат на переналадку и хранение

**Размер партии деталей** - количество деталей обрабатываемое на одной технологической операции без переналадки и смены режущего инструмента.

**Оптимальная партия** - партия, при которой одновременно малы затраты на переналадку и на хранение.



Поскольку на разных производственных этапах оптимальные размеры партий разные, то получается, что некоторое количество деталей все время лежит в ожидании следующего этапа. Это связанные оборотные средства. Для того чтобы избежать такого эффекта на производстве приняты следующие ограничения:

1. Размеры партии должны убывать по технологическому маршруту.
2. Размеры партии должны быть кратны.

### 2. *Норма задела.*

**Незавершенное производство** - все детали и заготовки, которые не прошли технологический маршрут до конца или технологическую обработку на данной операции.

Незавершенное производство может быть подсчитано в разных величинах.

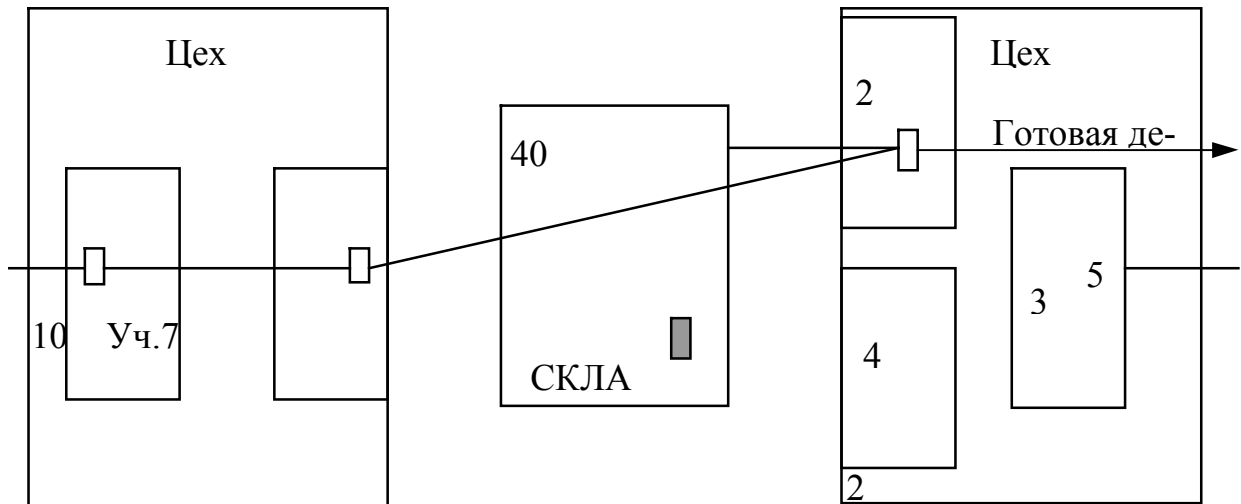
**Задел** - незавершенное производство выраженное в натуральных единицах измерения.

Величина задела предназначенная для ликвидации сбоя ритмичности называется **страховым заделом**. Страховые заделы подразделяются на цеховые и складские.

Цеховые предназначены для ликвидации боев ритмичности по внутренним причинам. Складские предназначены для ликвидации сбоя ритмичности на межцеховом уровне.

**Норма задела** - величина задела, минимально необходимого для бесперебойной работы.

Задел часто измеряют во временном измерении (трудоемкость или на сколько хватит задела).



**Оборотный задел**

$$O = 0.5(n1 - n2)$$

где

n1 - партия поставки,

n2 - партия обработки (потребления).

Оборотный задел возникает вследствие несогласованности партий и представляет собой оборотные средства.

Методы снижения оборотного задела:

1. Размеры партий по технологическому маршруту не должны убывать.
2. Размеры партий должны быть кратны.

**Цикловой задел** - количество деталей пролеживающих в производстве в течение длительности производственного цикла.

### 3. Длительность производственного цикла.

**Длительность производственного цикла** - время необходимое для изготовления детали, узла, заготовки на технологической линии. Или ее части т.е. различают длительность производственного цикла общую в цехе и на участке.

$$T_{умех} = \frac{\sum_{i=1}^n n_i * t_{wki}}{K_{cm}}$$

$t_{wki}$  - штучно-номенклатурное время, длительность обработки детали

$K_{см}$  - коэффициент сменности.

$n_i$  - количество деталей

$n$  - количество операций на технологическом маршруте

$$T_u = T_{umex} + T_{nep}$$

$T_u$  - длительность производственного цикла,

$T_{nep}$  - время перерывов.

Причем на реальных производствах  $T_{nep} \gg T_{umex}$

Перерывы состоят из:

$$T_{nep} = T_{mn} + T_{my} + T_{mu} + T_{mp} + T_{mc} + T_{bc} + T_{есм} + \dots$$

$T_{mn}$  - межоперационные перерывы

$T_{my}$  - межучастковые

$T_{mu}$  - межцеховые

$T_{mp}$  - транспортные

$T_{mc}$  - межсменные

$T_{bc}$  - внутрисменные

$T_{есм}$  - естественные.

**Опережение по запуску** - временной интервал равный длительности производственного цикла, на который сдвинут момент выпуска.

На основе всех рассмотренных величин в 1. 2. и 3. составляется план производства.

## ***Административная структура производства.***

Существующая система управления производством:

- предприятия, объединения
- заводы
- производства
- цеха имеют специализацию: либо предметную (цех валов, шестерен), либо технологическую (термический, гальванический).

## ***Планирование производства.***

### ***Основные понятия.***

Планирование разделяют на:

- ***перспективное*** (планирование на время от 1-5 лет);

- *оперативное* (сутки, квартал).

Планы бывают:

- *номенклатурные* (количество деталей и т.п.);
- *объемные* (стоимость, емкость и т.п.).

В нашей стране многократно вводили ошибочные показатели производства. Например одно время был введен стоимостной показатель, что привело к исчезновению дешевых товаров но дорогих товаров много не надо и рынок насытился. Что делать дальше? В результате многочисленных подобных экспериментов появилась идея что производительность нужно измерять следующим образом:  $V = T + \Delta \text{НЗП}$

V - валовой продукт (показатель)

T - товар

$\Delta \text{НЗП}$  - изменение незавершенного производства.

Такой показатель учитывает всю проделанную работу и сданную продукцию и незавершенную. Но незавершенное производство бывает разное например в строительстве можно только вбить сваи а можно только не врезать замки.

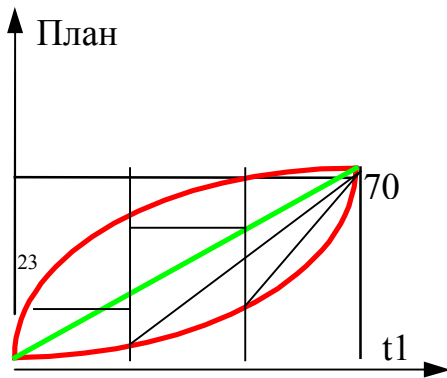
T  $\left\{ \begin{array}{l} \text{Рубли} \\ \text{Штуки} \end{array} \right.$  Товар стали учитывать как в рублях так и в штуках.

Обычно сначала составляется перспективный план на 3-5 лет. Такое планирование делают обычно в объемных показателях, но часто и в стоимостных. План детализируется до уровня изделий с целью анализа плана удовлетворению рыночной стратегии. Стоимостной показатель плох тем, что есть инфляция (мы хотим сделать продукции на 50 млн., а сколько это будет потом?). **Выход** - это расчет в ценах прошлых лет, а затем перевод в текущие. Различия в планировании у нас и на западе связаны с тем, что у нас господствует теория прибавочной стоимости Маркса, а на западе теория полезности. Затем проводят оперативное планирование. Например нам надо на 200 млрд. сделать 80 двигателей.

$$\overline{80 * K_i = P_i}$$

$P_i$  - количество i-х деталей       $K_i$  - применимость i-й детали

После этого всем цехам спускают решение кому сколько чего надо сделать. В цехах решают обратную задачу - задачу разузлования. Решили, что могут выполнить деталей на 73 двигателя. Затем делят план на четыре для разбивки по кварталам и округляют в меньшую сторону, - получают планы по кварталам (это планирование в объемных показателях).



При реализации экспоненциального управления вероятность выполнения плана на много больше.

При линейном управлении мы не сможем выполнить план. В процессе выполнения начинается корректировка плана (в министерство мы можем

	t цикла	n	План
A	20 дней	100 шт.	1000 шт.
B	10 дней	300 шт.	5000 шт.
C	5 дней	50 шт.	2000 шт.
D	30 дней	80 шт.	8000 шт.

только 68 т.к. 1-е, 2-е, 3-е ...)

В производство запускается то, что требует соседний цех. Такая система называется *подетальной системой оперативного управления*. При такой системе рассчитываются объемы

по каждой детали, после чего начинают составлять номенклатурный план с помощью методов сетевого планирования.

### Основные системы планирования.

#### 1. Система *показного* планирования.

Ис-

ходя из общей производственной мощности, собирается портфель заказов. Составляется календарный план выпуска заказов. Зная длительность цикла по каждой номенклатурной единице, определяют время запуска. На одно и тоже время и оборудование могут претендовать разные заказы. В этом случае разносят сроки запуска, сдвигая по некоторым позициям сроки запуска к началу планового периода. Но в этом случае заказ, который сдвинут на более ранний срок начинает конфликтовать с другим заказом; эта процедура повторяется до тех пор, пока не удастся разработать бесконфликтный план.

«-» Такой план

без организационных перерывов разработать практически невозможно. Поэтому, набирая портфель заказов мы должны это учитывать и оставлять некоторый резерв мощности (либо лишние станки, либо лишнее рабочее время).

#### 2. *Сутко-комплектная* система планирования.

На серийных предприятиях работает как правило эта система. Проводится разузлование плана (вычисляется план по каждой детали, величина плана по детали делится на длину планового периода).

$$\frac{\Pi_i}{t} = \frac{C}{n_i} \quad \text{- суточная потребность в детали } i\text{-го наименования.}$$

$\Pi_i$  - план  
 $t$  - периодичность

Определяется величина обеспеченности сборки деталями.

$$\frac{O_i}{C/n_i} \quad \begin{array}{l} \text{- обеспеченность} \\ O_i \text{ - сколько деталей на складе (остаток)} \end{array}$$

	1	2	3	4	5	6	7	...	31	На складе	Суточная потребность
A	X	X	X	X	X	X				600	100
B	X	X								800	400
C	X									200	150
D	X	X								400	200

Запускаем в производство ту деталь, которую хватит на наименьший промежуток времени. Неудобство в том, что это не система управления - это сигнальная система, которая сообщает лишь об отклонении (управление это мера воздействия которую мы должны вычислить по отклонению).

**Сутко-комплект** - совокупность деталей всех наименований в размере их суточной потребности (условное изделие).

«+» Наглядность, минимум расчета. Полная децентрализация системы, при сохранении синхронной работы участка.

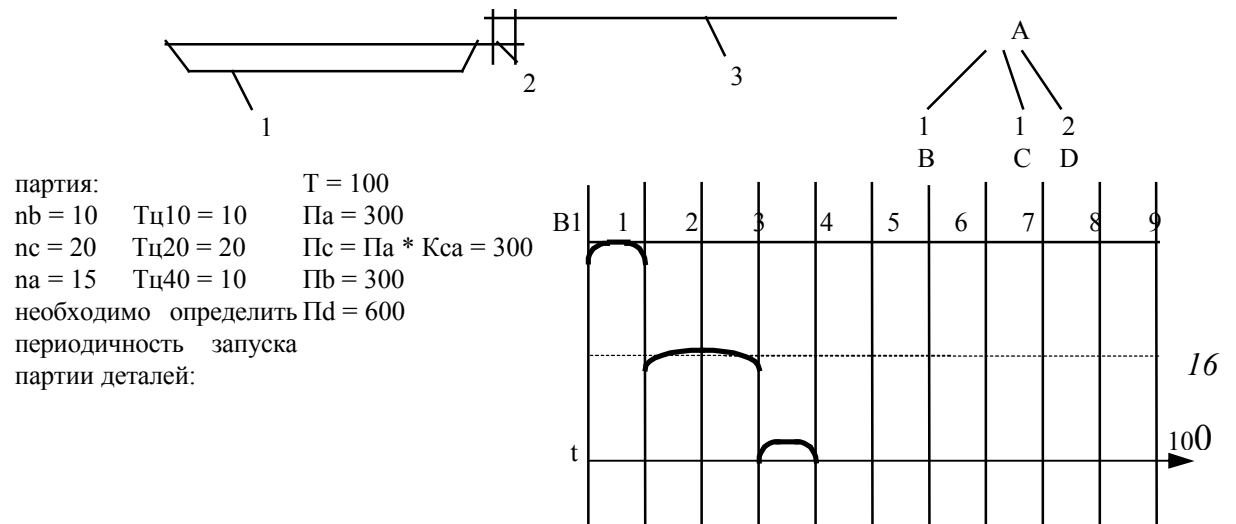
«-» Система является сигнальной, т.е. только сигнализирует о том какую деталь надо запускать в производство, но не дает ответа на вопрос какой должна быть мера управления.

В результате получили систему пропорционального управления.

Составляется сменно-суточное задание того что надо делать цеху. Такой по каждой детали на месяц вперед называют графиком запуска-выпуска.

A	3 дня	5 дней	7 дней	15 дней
	10 операций	20 операций	30 операций	40 операций

Задача составления графика *запуска/выпуска*.





1. Рассчитывается периодичность запуска по формуле.
  2. Определяются даты запуска деталей каждого наименования на каждой операции (1-ой), зная время запуска каждой партии деталей и длительность цикла на этой операции можно определить время выпуска каждой партии деталей.
- График позволяет задать длительность производственного цикла.

### ***Переменные системы управления производством.***

Единственная переменная, которой можно управлять - ***интенсивность работы на операции.***

***Пути управления:***

1. ***Интенсивный*** путь - путем увеличения производительности,
2. ***Экстенсивный*** путь - фактическая производительность не меняется, а меняется расчетная производительность,
3. ***Комбинированный*** путь.

Формы управления: моральное, материальное, физическое воздействия или сверх урочное время.

Особенностью является то что мы управляем по отклонению, а чтобы ее можно было указать необходимо:

1. Полностью автоматизированное складское хозяйство с оперативным учетом.
2. Необходимо рассчитывать график запуска выпуска, т.е. суточные задания  $\Rightarrow$  должна быть компьютерная сеть.

Наблюдение производится за объемом незавершенного производства (сколько деталей есть вообще), объемом выпущенных деталей. В некоторых случаях вводится еще одна характеристика - ***график запуска выпуска*** деталей, который позволяет контролировать обработку детали. Составляется по расходно-приходным накладным.

***Расходно-приходная накладная*** - документ, в котором отмечаются приходящие на обработку и уходящие с обработки партии деталей. По нему можно узнать все, что касается готовых деталей.

Чтобы учесть количество незаконченных деталей, обычно раз в месяц проводят тотальный обыск, результатом которого является инвентаризационная ведомость остатка. Это дорогостоящая операция, которая наполняется специальными людьми.

$$\Delta \text{НЗП} = \sum_{k=1}^m \text{Приход(сколько потупило на операцию)} - \text{Расход(сколько передано на следующую операцию)}$$

m - дата.

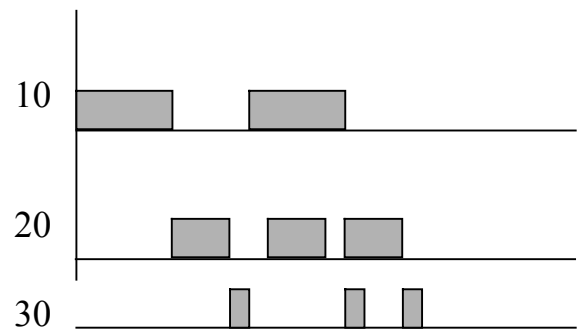
В течение месяца по ведомостям. Ошибка в этой формуле носит интегральный характер т.е. в течении времени накапливается.

$$\text{НЗП} = \text{НЗП}_0 + \sum \Delta \text{НЗП} - \text{расчетный остаток}$$

Чтобы построить график запуска-выпуска деталей, систему управления необходимо централизовать. Вводится диспетчер, который полностью отслеживает все технологические маршруты и контролирует обработку деталей по графику.

Составление графика:

№ операции	10	20	30
Размер партии	100	50	50
План	3000	-	-
Суточная потребность	30	-	-
Планов. период	100	-	-
Период запуска	3	1,5	1,5
Время цикла	2	1	0,5



Если сократить длительность производственного цикла до длительности технологического, то сокращается время простоя детали.

# ***Понятие автоматизированных систем управления и их классификация.***

## ***Определение АСУ.***

*АСУ* - называется система, объединяющая комплекс технических средств, экономико-математические методы, математическое, лингвистическое, информационное, организационное, юридическое и методическое обеспечение, а так же коллектив людей объединенных общей целью. Управляющее решение принимается в соавторстве человека с машиной. Системы делятся на примитивные элементарные (для них строятся автоматические системы управления) и большие сложные.

## ***Понятие большой и сложной системы.***

Большой и сложной называется система, обладающая следующими свойствами:

1. ***Свойство не наблюдаемости.*** Большая и сложная система не наблюдаема с позиции одного наблюдателя (каждый видит лишь часть экономист видит экономический срез, социолог социологическую среду и т.п.)
2. ***Большая и сложная система всегда эргодична.*** Т.е. как элемент всегда включает человека.
3. ***Большая и сложная система всегда стохастична.*** Действует по случайным законам (она не детерминирована).
4. ***Большая и сложная система всегда дискретна.*** Состоит из автономных независимых модулей.
5. ***Свойство непрерывности.*** Система непрерывно меняет свои параметры во времени (она динамична)  $\Rightarrow$  создать АСУ раз и навсегда до конца не возможно.
6. ***Система динамична,*** т.е. находится в постоянном развитии.
7. ***Способность к само регуляции и адаптации.*** Система имеет возможность менять свои параметры и подстраиваться к возмущениям и внешней среде. (чем выше уровень системы, тем больше возможность адаптации). Например, если увезти на курорт на два месяца все начальство завода то завод все равно будет работать
8. ***Самообучаемость.*** Идет накопление опыта, система совершенствуется.

*Все кибернетические системы, в основном, большие и сложные. То есть АСУ - большая и сложная система всегда.*

## ***Классификация АСУ.***

Все АСУ относятся к большим и сложным системам, поэтому обладают их свойствами. Т.к. большие и сложные системы обладают свойством необозримости, то их можно рассматривать с нескольких точек зрения. Следовательно, классификационных признаков тоже много (дать классификацию по одному признаку невозможно).

Классифицировать АСУ можно:

1. ***Интеграция АСУ по уровню.***

- *ОГАС* - общая государственная автоматизированная система (система выборов, система резервирования билетов, электрическая система)
- *АСУО* - АСУ отрасли
- *АСУГ* - АСУ главка (главк позволяет управлять группой предприятий)
- *АСУП* - АСУ предприятия
- *АСУЦ* - АСУ цеха
- *АСУУ* - АСУ участка
- *АСУТП* - АСУ технологического процесса

2. ***Интеграция по уровню объекта.***

- АСУ федерации
- АСУ региона
- АСУ области
- АСУ города

3. ***По уровню использования ЭВМ.***

- большие
- малые
- персональные
- супер-большие
- супер-малые
- ЦВМ
- АВМ
- гибридные.

4. ***По назначению.***

- производственного типа
- экономические (предприятия, конторы, управляющие властные структуры)
- коммерческие
- медицинские
- военные
- информационно-поисковые системы
- многие другие.

У этих систем разные критерии.

5. ***По типу выдаваемой информации.***

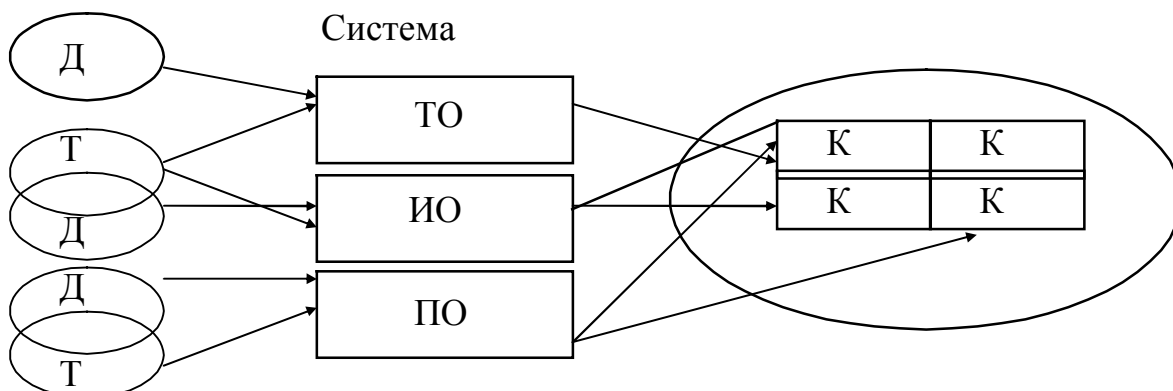
- чисто информационные системы (экспресс, сирена)
- информационно-советующие

- управляющие АСУ (они не только предлагают варианты но и могут реализовать выбранный человеком вариант).
6. **По типу производства.**
    - АСУ с дискретно-непрерывным производством
    - АСУ с дискретным производством
    - АСУ с непрерывным производством
  7. **По областям человеческой деятельности.**
    - медицинские системы
    - экологические системы
    - системы телефонной связи
    - многие другие.

### **Принципы построения АСУ.**

1. **Принцип новых задач.** В первую очередь должны автоматизироваться те задачи, не решение которых наносит наибольший ущерб с одной стороны и решение которых не было возможно до сих пор по какой либо причине.
2. **Принцип непрерывного развития.** Система должна быть построена так, что бы изменение в состоянии объекта или условиях управления не должны приводить к гибели системы.
3. **Первого руководителя.** Руководить внедрением подсистемы или всей системы АСУ должен руководитель по уровню не ниже главного специалиста (идеально директор).
4. **Принцип модульности.** Система или подсистема должна разрабатываться и внедряться как набор автономных модулей. Должно быть поэтапное подключения модулей.
5. **Принцип системного подхода.** Различают три подхода:
  - индуктивный,
  - дедуктивный,
  - системный.

**Индуктивный** и **дедуктивный** подходы называют еще **отзадными** подходами, т.е. к построению системы приходят от задачи. Собираются данные, а также требования к системе, на основе части данных формируются дополнительные требования, на основе требований формируют компоненты системы, затем объединяют компоненты в систему.

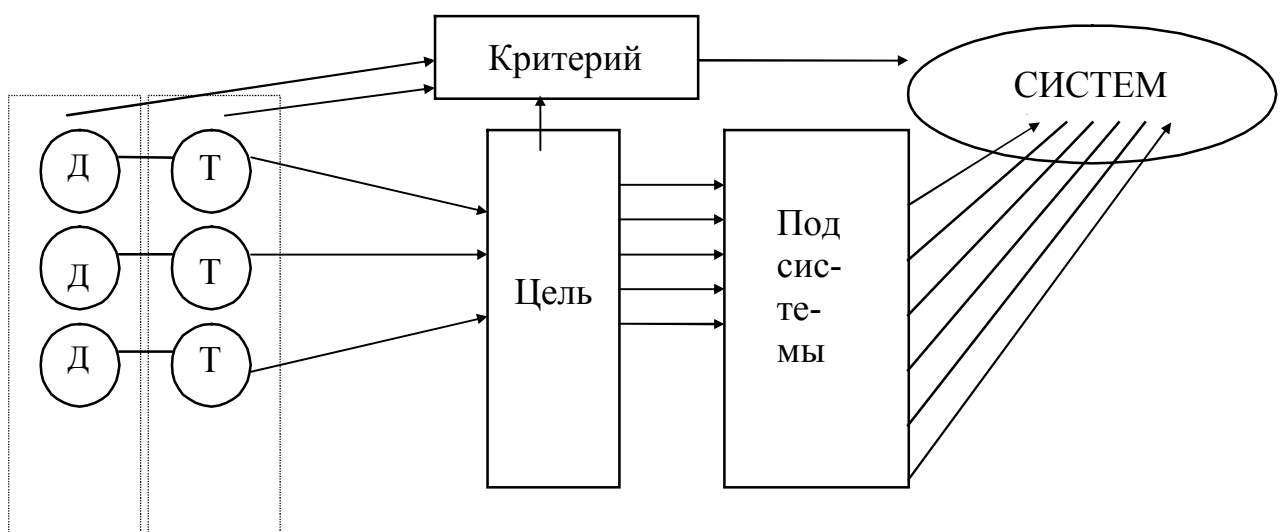


**Отзадачный** подход наиболее популярный, но у него много *недостатков*:

- компоненты системы формируются насильственно;
- отдельные задачи плохо стыкуются друг с другом, так как разрабатывались разными людьми и преследовали разные цели.

**Системный** подход имеет следующие отличия. Собираются данные и некоторые требования к объекту. На основании этих требований и данных формируется цель системы (что мы хотим от системы в целом). На основании цели формируются принципы построения системы в целом, одновременно формируются показатели и критерии достижения цели.

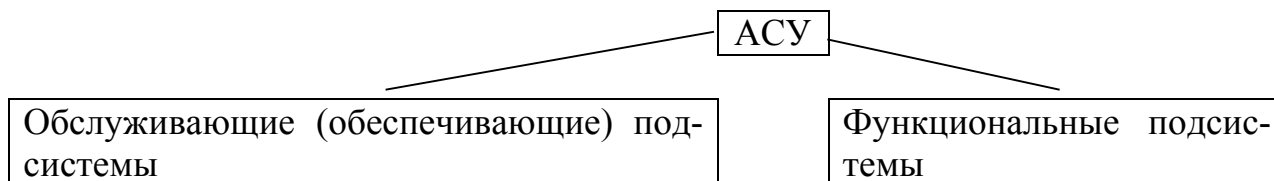
На основе принципов формируется система требований, на основе которых разрабатываются компоненты, причем каждая компонента формируется в нескольких вариантах, лучший из которых выбирается на основе критерия достижения цели исходя из задач всей системы. На основе этого формируется единственный вариант системы.



«-» Большие затраты времени на разработку, а именно системных вопросов. Ведется разработка всей системы в целом, а не отдельных ее задач, что требует достаточно больших ресурсов.

# Структура АСУ.

## Общая схема структуры АСУ.



Функциональные подсистемы решают задачи управления в рамках не которых функций управления объектом.



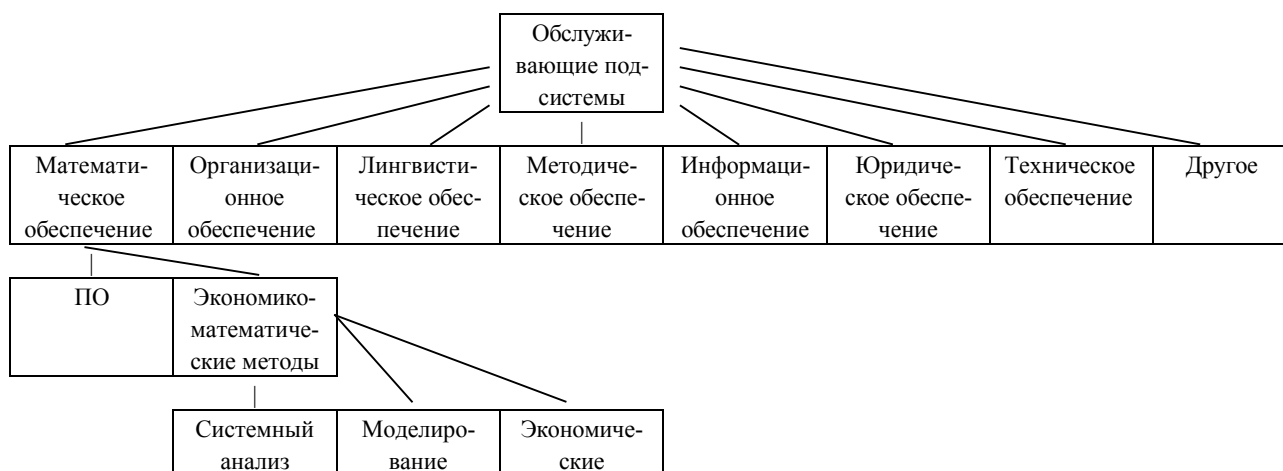
Дальнейшее деление зависит от типа АСУП.

Обслуживающие подсистемы предназначены для обеспечения эффективной работы функциональных подсистем.

К обслуживающим системам относятся подсистемы:

- технического обеспечения - комплекс технических средств, который включает средства коммуникации, связи и др.;
- информационного обеспечения - справочники, классификаторы, кодификаторы, система документооборота, информационные массивы;
- программного обеспечения - совокупность всех программных систем;
- организационного обеспечения - совокупность приказов, директив, инструкций, указаний, которые позволяют объекту работать слаженно;
- методического обеспечения - комплекс методических указаний для всех систем;
- лингвистического обеспечения - комплекс языковых средств;
- юридического обеспечения - комплекс нормативных и правовых документов, обеспечивающий юридическую поддержку АСУ;
- математического обеспечения - набор материала, который позволяет грамотно вести управление.
- многие другие подсистемы.

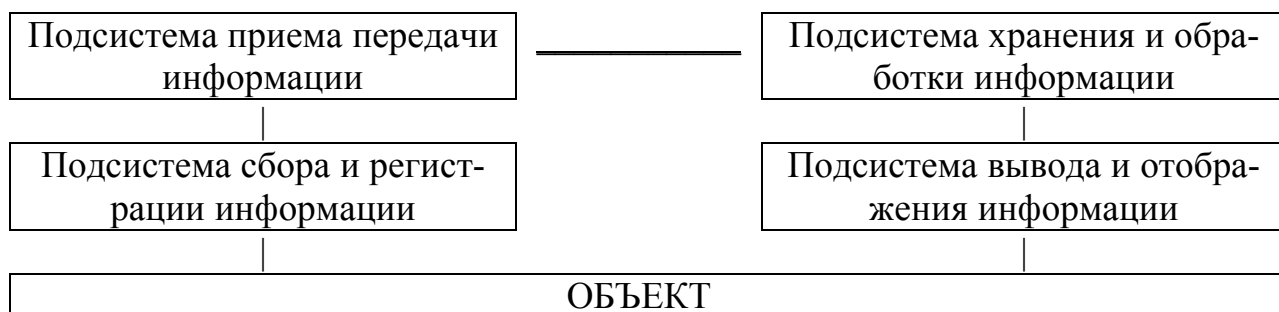
Структурная схема обслуживающих подсистем приведена ниже:



### *Техническое обеспечение АСУ.*

#### *Общая схема технического обеспечения АСУ.*

Техническое обеспечение АСУ состоит из подсистем, которые приведены ниже в структурной схеме.



#### *Подсистема сбора и регистрации информации.*

##### *Состав.*

Состоит из:

- Средств сбора информации (датчиков);
- Средств регистрации информации;
- Средств преобразования информации.

##### *Датчики.*

Датчики являются средством выделения информации.

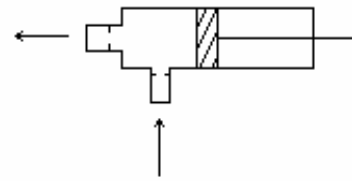


1) Датчики физико-химических величин.

а) Датчики объема

Мерная тара

Автоматические дозаторы

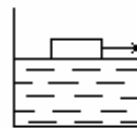


Измерители уровня

\* Оптического типа

\* Линейчатого типа

\* Поплавкового типа



\* Индуктивного типа

\* Емкостного типа

б) Датчики расхода  
Датчики расхода

делятся на датчики расхода

- Сыпучих веществ
- Газа
- Жидкости

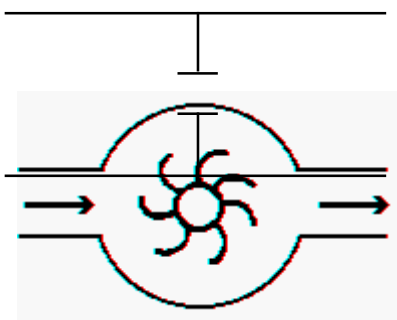
По устройству они делятся на:

датчики определяющие

крылатка

мембранного типа

расход по скорости вращения (меряет давление и скорость)



Датчики на основе измерения давления

Электронные расходомеры измеряют скорость движения среды за счет меченых ионов.

в) Датчики перемещения:

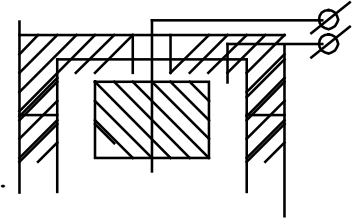
- Датчики скорости
- Датчики ускорения
- Датчики пути

Датчики перемещения: *контактные и оптические.*

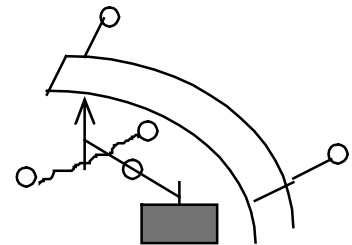
Датчики скорости обычно на основе тахометров т.е. преобразуют скорость вращения в поступательную скорость либо меряют ускорение а затем его интегрируют. Датчики Пути строятся на основе: спидометров, тахометров.

Измерители ускорения:

Контактного типа Колокол  
Регистрирует только сам факт наличия ускорения можно использовать как противоугонное устройство (наличие ускорения).



Датчики Скорости в основном строятся на основе измерения: числа оборотов, угла скорости. Тахометр. Последовательно с измерителем ускорения ставится интегратор.



2) Датчики экономических сообщений.

Благодаря этим датчикам можно оценить экономическую ситуацию на производстве, степень незавершенного производства, примерно оценить время готовности продукта труда и т.п.

- Автоматические сигнализаторы. Представляют собой контактную или оптическую цепь на конвейере срабатывающую при прохождении предмета труда.
- Ручные сигнализаторы. Чаще всего кнопка рядом со станком. Рабочий изготовив деталь нажимает на эту кнопку.
- Регистраторы производства. Представляют собой сеть ПК или большую ЭВМ с разнесенными терминалами или со специальными терминалами (клавиатура + ЖК дисплей на 3 строчки + принтер + элементарный процессорный блок. Цена такого терминала около 300\$ (не жалко выбросить) + он очень высокая степень надежности). Принцип таких регистраторов следующий в конце рабочего дня каждый рабочий подает данные о том что он сделал мастеру а мастер составляет отчет по цеху на день. Этот отчет поступает на центральный компьютер где данные проходят анализ и обработку.

### ***Преобразователи информации.***

Предназначены для преобразования сигнала снимаемого с датчика в унифицированный сигнал для компьютера.

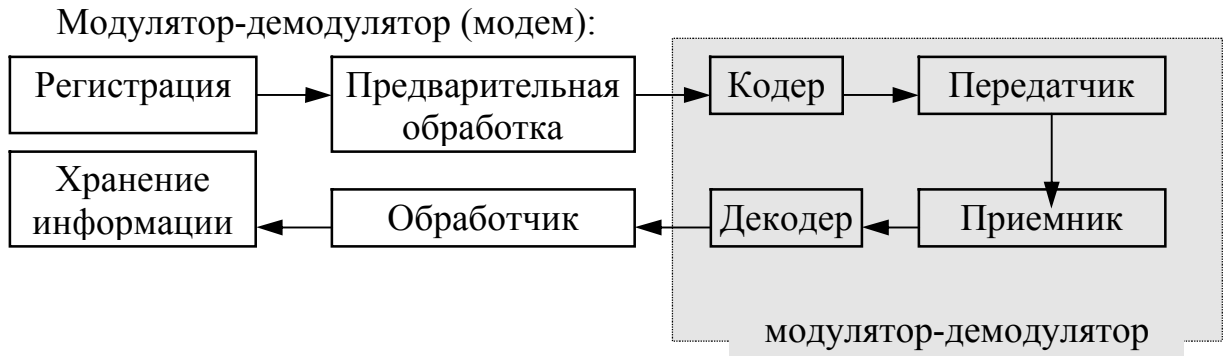
Существуют три причины, по которым датчики не совмещают с преобразователями:

- 1) По *габаритным* соображениям. Часто преобразователь имеет много большие габариты, чем датчик. Например, датчиком в магнитофоне является головка, а преобразователем различные усилители (пред усилитель, усилитель НЧ, усилитель ВЧ, усилитель мощности).
- 2) По *экономическим* соображениям. Если бы датчик и преобразователь были совмещены, то при выходе одного из них приходилось бы выбрасывать эту совмещенную систему и приобретать новую. Например, при выходе в магнитофоне из строя головки приходилось бы не головку менять, а покупать новый магнитофон.
- 3) По соображениям *безопасности*. Часто в цехах с повышенной пожарной опасностью все измерители имеют гидравлический принцип работы, а преобразователи обычно имеют электрическую основу. Поэтому, если поставить такой преобразователь в цех, то он вовсе не повысит пожарную безопасность, а скорее наоборот. В подобных случаях измерители и преобразователи необходимо разносить по соображениям безопасности.

Преобразователи бывают:

- 1) Гидравлические:
  - гидро - гидравлические
  - гидро - пневматические
  - гидро - электрические
  - гидро - механические.
- 2) Пневматические:
  - пневмо - пневматические
  - пневмо - гидравлические
  - пневмо - электрические
  - пневмо - механические.
- 3) Электрические:
  - электро - электрические
  - электро - гидравлические
  - электро - пневматические
  - электро - механические.
- 4) Механические:
  - механо - электрические
  - механо - гидравлические
  - механо - пневматические
  - механо - механические.

## *Подсистема приема и передачи информации.*



Характеристики модемов:

- скорость передачи данных;
- протокол соединения;
- наличие возможности голосовой связи;
- исполнение (конструктивные особенности).

Порог, когда помехи начинают влиять на основной сигнал определяют расстояние, на которое можно передавать информацию. Для увеличения этого расстояния существуют промежуточные устройства, которые позволяют усиливать сигнал, сохраняя уровень помех неизменным.

Модемы работают либо по выделенному каналу, либо по телефонным линиям, что не всегда приемлемо из-за их качества.

Иногда проще оказывается протянуть несколько проводов и поставить быстрый и дешевый short-модем, но этот способ годится только при небольших расстояниях.

Радиомодемы у нас в стране используются очень редко из-за очень малого количества разрешенных частот.

Если прокладка кабеля затруднена или невозможна, связываемые пункты находятся в пределах прямой видимости, то достаточно эффективным способом решения этой проблемы является установка модема, передающего информацию по лазерному лучу. Недостатки такого способа заключаются в достаточно сильном влиянии погодных условий (дождь, снег, туман) на качество передачи информации.

## *Подсистема хранения и обработки информации.*

### *Подсистема обработки информации.*

Подсистема обработки информации классифицируется, в основном, по типу процессора:

- большие ЭВМ
- малые ЭВМ
- микроЭВМ
-

**Производительность** измеряется в элементарных операциях в секунду, задается тактовой частотой генератора.

Класс ЭВМ	Производительность, млн. опер/с	Емкость ОЗУ, Мбайт	Цена, млн. долларов
СуперЭВМ	более 100	более 50	более 5
Средние ЭВМ	10 -100	10 - 50	0.5 - 5
Супер мини ЭВМ	5 - 20	2 - 10	0.1 - 0.5
Мини ЭВМ	1 - 5	1 - 4	0.02 - 1
Микро ЭВМ	0.5 - 1	0.1 - 1	менее 0.02

Постепенно влияние характеристик процессоров на возможности платформы все более увеличивается, так как они берут на себя все большую часть функций плат расширения. На данный момент ситуация на микропроцессорном рынке такова:

В августе 1995 года промышленной группой SPEC (System Performance Evaluation Cooperative) для более корректной оценки производительности современных компьютерных систем принят новый эталонный контрольно-оценочный тестовый набор SPEC95, единицей измерения которого стал компьютер SPARCStation 10 (40 МГц).

Процессор	Частота, МГц	SPECint95	SPECfp95
IBM-Motorola-Apple PowerPC			
PowerPC 604	133	4,55	3,31
PowerPC 604e	166	6	5
PowerPC 620e	166	7,7	9,5
Intel			
Pentium	150	4,27	3,04
Pentium	166	4,76	3,37
Pentium Pro	150	6,08	5,42
Pentium Pro	166	7,11	6,21
Pentium Pro	180	7,29	6,1
Pentium Pro	200	8,09	6,7
Hewlett-Packard PA-RISC			
PA-7300LC	>150	5,5	7,3
PA-8000	>=200	9	17
SUN SPARC			
UltraSPARC I	143	5,1	7,4
UltraSPARC II	250	8,5	15
Silicon Graphics MIPS			
R5000	200	5,5	5,5
R8000	200	9	14

R10000	275	12	24
DEC Alpha			
Alpha 21164	266	6,43	10,6
Alpha 21164	300	7,3	11,6
Alpha 21164	>400	11	17

SPECint95 - целочисленные операции; SPECfp95 - операции с плавающей точкой.

Но производительность системы определяется не только процессором. Второй по важности деталью современной ЭВМ является «материнская плата» а именно разрядность шин интерфейсов, тип арбитража и способ обработки ПДП и прерываний а также от производительности набора микросхем используемого на плате.

#### Характеристики материнских плат:

Название платы	Изготовитель BIOS	Чип сет (Набор микросхем)	Число слотов SIMM/DIMM	Наличие гнезда для доп. КЭШ	Число слотов ISA	Число слотов PSI	Поддержка AMD R5	Поддержка Cirix 6x85	Производительность	Цена \$
Asus P/I-P55TVP4	Award	Intel 430HX	4/1	+	3	4	+	+	хорошо	160
A-Trend ATC-1020	Award	Intel 430VX	4/1	+	3	4	+	+	хорошо	110
Intel Advanced ML	AMI	Intel 430HX	4/0	-	3	4	н/д.	н/д.	хорошо	115
SOYO SY-5TF2	Award	Intel 430HX	4/0	+	4	4	+	-	хорошо	130
Tomato 5DHX	Award	Intel 430HX	4/0	-	4	4	+	+	удовлетворительно	100
TyanS1470(Titan)	Award	Intel 430VX	4/2	+	5	5	+	+	хорошо	200

Звуковые карты преобразуют цифровой сигнал в звуковой.

#### Характеристики звуковых карт:

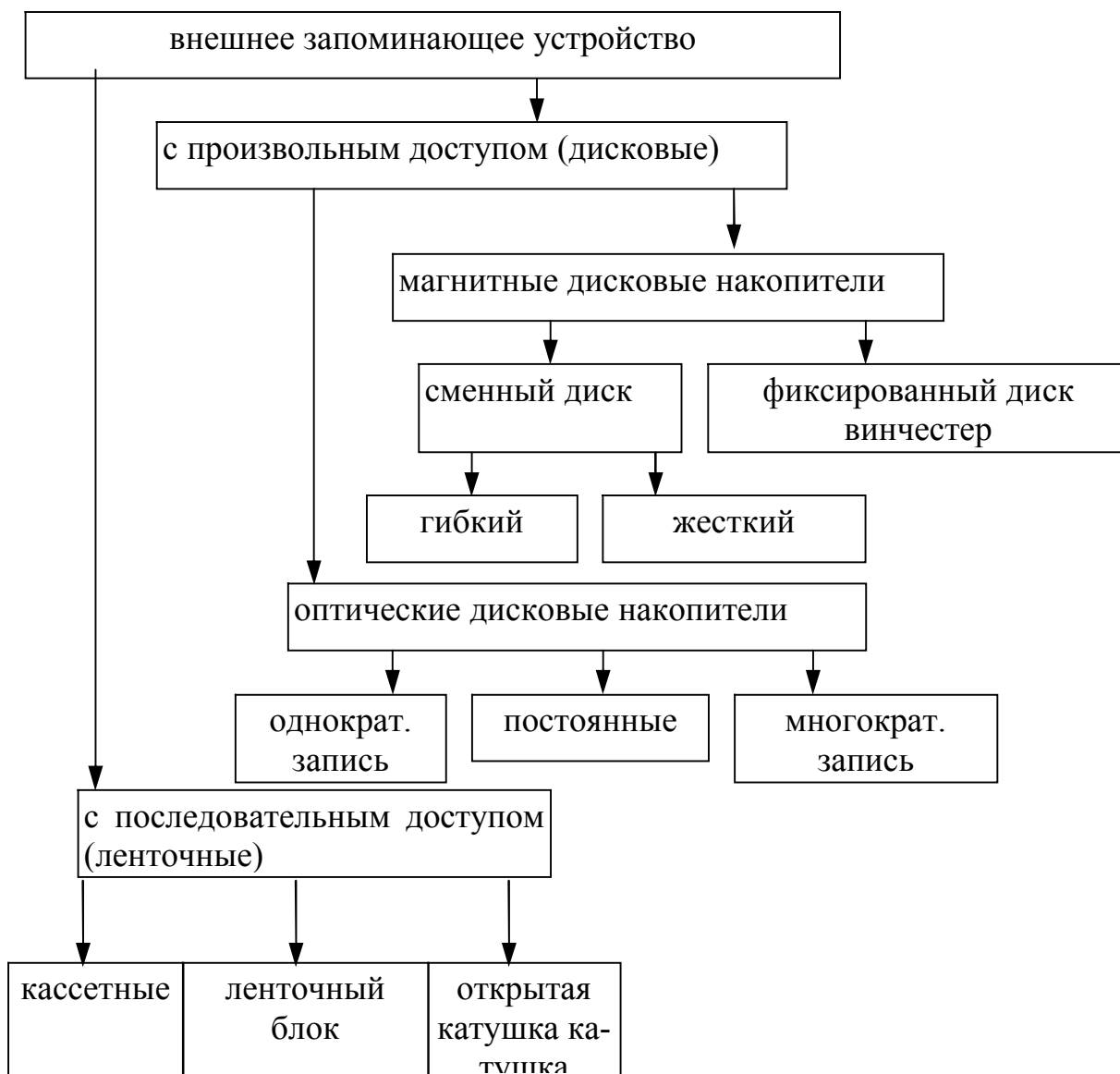
Карта	Цена	Вход	Выход	Тбл инст р.	Тбл нот	Тбл го-лосо	3D	ROM /RAM	Сэм-плинг	Мощность Вт	Аудио CD	MID I	WA V
Sound Blaster 16 value	85	ЛМ	ДЛ			в	Д	Н	16бит / 441	4	3,5	4	4а
Sound Blaster AWE 32	155	ЛМ	ДЛ	16	32	128	Д	1МБ/ 512К	16бит / 44.1	4	3,5	4	3,5
Sound Galaxy Waverider	59,95	ЛМ	ЛД	16	32	128	Д	1Мб	16бит / 44.1	4	3,5	4,5	4

Maxi Sound 64	189, 9	ЛМ	ДЛ П	16	32	425	Д	4Мб/	16бит / 44.1	4	4	4,5	4
---------------	-----------	----	---------	----	----	-----	---	------	--------------------	---	---	-----	---

Вход: Л - линейный М - микрофон Выход: Д - динамики Л - линейный

### Подсистема хранения информации.

Структурная схема ЗУ приведена ниже.



**Оптические накопители.** Информация хранится в виде микроскопических углублений на поверхности диска; яркость отражения лазерного луча изменяется, когда он попадает в углубления. Эти колебания яркости воспринимаются оптической считывающей головкой, которая преобразует их в двоичный последовательный сигнал.

В **накопителях на гибких дисках (НГМД)** в качестве носителя используется пластиковый диск с ферромагнитным покрытием, заключенный в специальный защитный конверт. При работе диск вращается; информация на дорожке записывается по секторам. Ранее в обиходе были 8 а затем 5-дюймовые ГМД. Сначала односторонние с емкостью 180 Кб, затем двусторонние с емкостью 360, 720, 800, 1200 Кб. Позже им на смену пришли более компактные 3,5-дюймовые ГМД, которые имели более прочный пластиковый конверт и большую емкость (до 2,88 Мб). Но постоянный рост объемов информации требует использования более эффективных носителей. Сейчас появились новые накопители подобного типа. Это Zip и Jazz drives, помимо этих получивших наибольшее распространение подобные накопители выпускаются многими фирмами с небольшими отличиями.

Дисковые накопители	Объем, Гбайт	Цена носителя \$	Цена \$	Производительность
Omega Jaz	2/1	99	500	отлично
Omega Zip	0,2/0,1	15	150	хорошо

**Накопители на жестких магнитных дисках (винчестеры).** В качестве носителя используются алюминиевые диски (1 или 2), вращающиеся с очень высокой скоростью (тысячи оборотов в минуту), что позволяет получать более быстрый доступ к информации (несколько миллисекунд). Скорость передачи данных около 2Мб/с. Во избежание попадания пыли механизм накопителя закрыт в вакуумный корпус. Винчестеры, обладая высокой скоростью вращения, являются сильными гироскопами, поэтому перемещение работающего винчестера в пространстве может вызвать выход его из строя. Объем информации, хранимой на винчестере может достигать нескольких гигабайт.

Марка	Емкость, Мбайт	Форм фактор, дюйм	Среднее время поиска, мс	Время поиска соседней дорожки, мс	Режим PIO	Внутренняя скорость обмена данными, Мбайт/с.	Размер кэш буфера, Кбайт	Скорость вращения шпинделя, об/мин
Maxtor 71050A	1050	3,5	<12	<2	Mode 3	3.97-6.19	256	4500



72004AP/A	2004	3,5	<12	2	Mode 4	4,74-8,10	128/64	4480
Fuji tsu								
M1606TA	1089	3,5	10	3	Mode 4	4,34-7,84	256	5400
M1638T	2568	3,5	10	3	Mode 4	7,7-13,8	128	5400
Seagate								
ST31081A		3,5	14	н/д.	Mode 4	н/д.	64	3600
ST31720A		3,5	12	3	Mode 4	н/д.	128	4500

Модель	Цена \$	Объем, Гбайт	Скорость вращения об/мин	Время доступа мс	КЭШ память Кбайт	Под- держка SMART 1	Форм фактор, дюйм	Прилагае- мое ПО	Направляю- щие/адапте р
Western Digital Cavair 32500	310	2,5	5200	11	128	Да	3,25	E2 Drive	Да/Да
Quantum Bigfoot CY 6.4	499	6,4	3600	14	128	Да	5,25	DiskManager Norton SmartDoct.	Да/Нет

**Приводы CD-ROM** работают со сменными лазерными дисками емкостью около 600Мб и позволяют достигать скорости передачи данных 900Кб/с при среднем времени доступа около 150 мс. Достоинствами этого типа носителя являются его нечувствительность к внешним воздействиям (магнитным, тепловым, можно уронить), легко заменяемость и невысокая стоимость. Однако информацию, однажды записанную, ни стереть ни изменить нельзя (за исключением дорогих перезаписывающих приводов CD-ROM, которые позволяют несколько раз записывать информацию на диски с золотой основой).

Название дисководов, CD-ROM	Интерфейс	Скорость передачи данных, кратность	Скорость передачи данных, Кбайт/сек	Среднее время доступа, мс	Способ загрузки диска	Наличие "муз." кнопки	Ориентировочная цена, \$
Aztech CDA 1068-01I	Atapi	10x	1500	135	Tray	+	140
Hitachi CDR-8130	Atapi	12x	1200-2400	90	Tray	-	140
Panasonic CR-548B	Atapi	12	1200-1944	110	Tray	-	140
Philips 1070/10	Atapi	10x	1500	140	Tray	+	125
Pioneer DR-444 (DR-A12X)	Atapi	12x	1800	110	Tray	-	145

**Стримеры** - устройства, работающие с кассетами с магнитной лентой. Позволяют хранить огромные объемы информации. Доступ к информации только последовательный. Из-за низкой скорости используются преимущественно для резервного копирования данных.

Стимеры	Объем, Гбайт	Цена носителя, \$	Цена, \$	Производительность
Iomega Ditto 2Gb	2/1	15	180	хорошо
Iomega Easy 3200	3,2/1,6	35	230	хорошо

## *Подсистема вывода и отображения информации.*

### *Мониторы.*

Характеристики мониторов:

1. монохром/цвет;
2. размер светового пятна;
3. количество строк и столбцов;
4. размер по диагонали;
5. назначение:
  - алфавитно-цифровой;
  - графический;
  - смешанный;
6. способ формирования символа;
7. по устройству воспроизведения:
  - электронно-лучевая трубка;
  - на жидких кристаллах.

«**Шаг**» - расстояние между точками изображения, чем меньше «шаг», тем выше качество изображения.

Фирма-изготовитель	Цена, \$	Формат экрана	Шаг, мм	Монохром/Цвет	Устройство воспроизведения
AmplexCorp.		80*25		зеленый	ЭЛТ
Communications		80*48		8 цвет	ЭЛТ
PET Computers				256 оттенков серого	ЖК
Idek	1795		0.28	цвет	ЭЛТ
Pilips	2695	1280*1024	0.26	цвет.	ЭЛТ

На мониторы существует множество стандартов защиты. Самый лучший это стандарт MPR II но только один монитор был протестирован и соответствует этому стандарту.

### *Защитные экраны.*

Для защиты часто применяют защитные экраны:

- сетчатые (убирают блики),
- стеклянные (снижают яркость свечения и снимают статику),
- полароидная полная защита (нет бликов, изображение видно только в диапазоне 15 градусов).

### **Принтеры.**

Основные характеристики принтеров:

1. Назначение:

- *принтеры для коротких сообщений;*
- *эскизный;*
- *принтеры для непрерывной работы.*

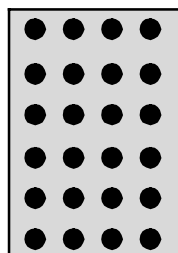
2. Скорость печати:

- *медленные;*
- *быстрые.*

3. Способ формирования изображения.

- *матричные принтеры:*

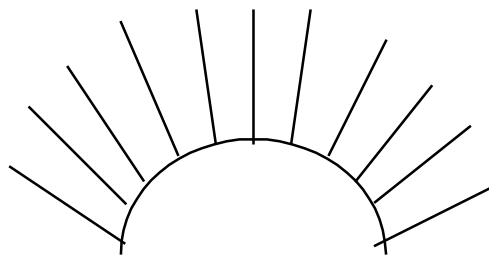
Принцип действия: иголки пробивают через красящую ленту нужное изображение.



Игольчатый принтер  
9-ти игольчатые  
(Epson, Citizen)  
24-х игольчатые  
(Braser, Epson)

- *лепестковые принтеры:*

Принцип действия: на каждом лепестке ромашки свой символ.



«-» отсутствует: графика, быстрая смена шрифтов;

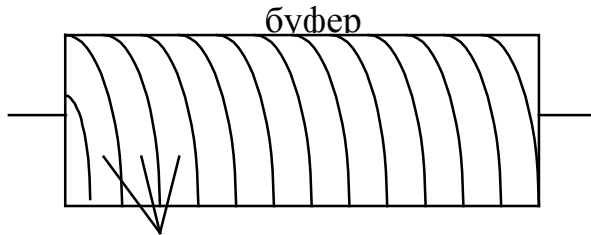
«-» машина очень шумит и ненадежна;

«+» довольно дешевое устройство.

- *АЦПУ барабанного типа:*

Принцип действия: на каждом барабане все символы алфавита/цифр.

Построчная печать (1200 строк в минуту).



«-» отсутствие графического режима, смены шрифтов.

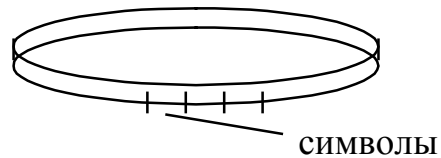
«+» высокая скорость печати и надежность устройства ( для круглосуточной работы ).

- *АЦПУ цепного типа:*

Принцип действия: в устройстве имеется лента с нанесенными символами.

«-» низкая скорость печати

«-» шрифт сменить невозможно



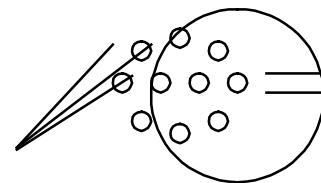
- *АЦПУ шарикового типа:*

Принцип действия: печатающим элементом является шарик с символами.

«-» высокий уровень шума.

«+» высокая стойкость.

«+» есть возможность смены шрифта.

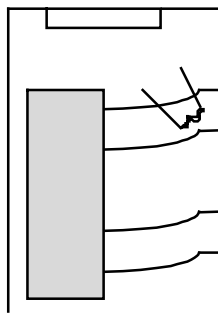


- *Термопринтеры:*

Принцип действия: похож на принцип матричных, только вместо иголок стоят нагревательные элементы.

«+» относительная бесшумность.

«-» невысокая скорость. Требуется наличие специальной бумаги.



- *Струйные принтеры:*

Принцип действия: матрица с форсунками, из которых на бумагу тем или иным способом наносятся капельки чернил.

Ведущие производители: Hewlet-Packard, Canon Computer Systems.  
 Цена: 400-2000 долларов.

- *Струйные (цветные):*

Принцип действия: в цветных струйных принтерах используются чернила четырех цветов (бирюзовый, малиновый, желтый и черный - Cyan, Magenta, Yellow, black - СМΥК принтеры) или только трех (СМУ принтеры - черный цвет получается путем смешения всех остальных цветов).

«+» относительная бесшумность, высокое качество изображения, возможность работы с графикой, несколько цветов (3-х цветные).

«-» невысокое быстродействие, полученное изображение приходится просушивать.

	Каретка	Число цветов / соплов	dpi (текст)	dpi (графика)	ОЗУ, Мб	Windows 3.1	OS/2	Windows NT
Canon BJC-70	узкая	64/64	720x360	360x360	0,035	BBS	BBS	-
DEC Color-Writer 550 ic	узкая	50/51	300x300	300x300	0,128/0,256	компл.	-	-
HP DeskJet 600c	узкая	48/48	600x600	600x300	0,512	BBS	BBS	-
HP DeskJet 850c	узкая	300/492	300x300	600x300	1	BBS	BBS	-
HP DeskJet 1600cm	широкая	300/416	600x600	600x600	6/70	компл.	в OS/2	комплект
Lexmark WinWriter 150c	узкая	64/192	600x300	600x300	0,029/0,029	BBS	-	-
Lexmark Color JetPrint 4079 plus	широкая	64/192	360x360	360x360	4/36	BBS	BBS	BBS
Olivetti JP 360	узкая	50/51	300x300	300x300	0,08/4	компл.	нет	нет
Tektronix Phaser 140	узкая	64/256	360x360	360x360	8/24	BBS	BBS	BBS

Принтеры	Разрешение точек /дюйм	Максимальная скорость, стр./мин	Тип	Цена, \$
HP LaserJet SL Xtra	600X600	4	лазерный	479
Epson Stylus Color 500	720X720	4	струйный	279
HP DeskJet 820 Cse	600X600	5	струйный	389

- *Цветные струйные с твердыми чернилами:*

Принцип действия: твердые чернила путем нагревания переводятся в жидкое состояние и наносятся на носитель тем же способом, что и в струйных принтерах. Перед началом работы такие принтеры должны нагреваться в течение 25 мин.

Ведущие производители: Tektronix, Brother International.

Цена: 10 тыс. долл.

«+» хороши для макетирования и создания пробных оттисков графических изображений. Широкий диапазон типов поверхностей, на ко-

торых возможна печать данным принтером (пористые и неровные поверхности).

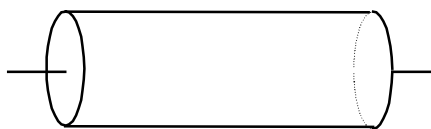
«-» отпечаток имеет характерный блеск и неровен на ощупь, по стойкости уступает изображению, полученному на лазерном принтере.

- *Лазерные принтеры:*

Принцип действия: схож с принципом действия ксероксов. Селеновый слой электризуется и запоминает символы, барабан обсыпается угольным порошком, в местах, где барабан наэлектризован, тонер прилипает; между валиками прокатывается бумага, порошок попадая на нее спекается.

Ведущий производитель: Hewlett Packard.

Цена: \$ 1396 – 5800.



«+» очень высокое качество изображения, непрерывный символ. Принтер формирует сразу изображение всего листа ( большой буфер до 4Мбайт ).

«-» высокая стоимость.

Для многоцветной печати существует модификация лазерного принтера фактически это три лазерных принтера собранных в один.

- *Цветные лазерные принтеры:*

Принцип действия: статический заряд помещается на фото приемный барабан или ленту. Луч лазера выделяет области, на которые не нужно наносить краситель. Полученное изображение электростатическим способом переносится на бумагу. Барабан или лента делает до четырех оборотов - по одному для нанесения тонера каждого цвета. В заключение для закрепления изображения на него воздействуют теплом или давлением либо тем и другим.

Ведущие производители: QMS, Hewlett-Packard.

Цена: 7,2 - 11 тыс. долл.

«+» отличное качество печати черно-белого текста, хорошее качество печати в цвете.

«-» расходные материалы требуют частой замены, может быть цветное несоответствие.

#### Цветные лазерные принтеры:

	Скорость монохр.	Скорость цвет.	ОЗУ, Мб	Печатающий механизм	Процессор	Тип процессора	Ресурс тонера
Apple Color Laser-Writer 12/600PS	12	3	12/40	Canon	30 AMD	RISC	4000
QMS magicolor LX	12	6	12/64	Hitachi	33 Intel	RISC	3000

Tektronix Phaser 340	4	4	8/24	Tektronix	33 AMD	RISC	3000
Tektronix Phaser 540 Plus	14	3-5	80/52	Matsushita	32 AMD	RISC	6500
Xerox Xprint 4915	12	8	16/46	Hitachi	25 AMD	RISC	4000
Xerox Xprint 4925	12	3	14/46	Hitachi	25 AMD	RISC	4000

- *Термические пигментные принтеры:*

Принцип действия: процесс печати состоит в нагревании цветного пигмента, находящегося на красящей ленте, для перенесения его на бумагу. Для печати одной страницы требуется четыре прохода, при каждом из них переносится краситель одного из цветов - бирюзового, малинового, желтого или черного. Черный краситель можно сымитировать, смешивая красители остальных трех цветов.

Ведущие производители: Tektronix, Fargo Electronics, Seiko InstrumentsUSA.

Цена: 1-2 тыс. долл. за принтеры младших моделей и 3-9 тыс. долл. за принтеры более высокого класса.

«+» хорошее качество печати на прозрачных пленках.

«-» низкая скорость печати, низкое качество печати на обычной бумаге, одинаковая стоимость печати при любой площади покрытия бумаги красителем.

- *Термические сублимационные принтеры:*

Принцип действия: используется технология прокатки. Краситель, нагретый до 400 градусов по Фаренгейту, испаряется с печатающей головки и через небольшое отверстие попадает на бумагу. Такая технология требует специальной бумаги, похожей на неэкспонированную фотопленку.

Ведущие производители: Tektronix, Eastman Kodak.

Цена: 7 - 20 тыс. долл.

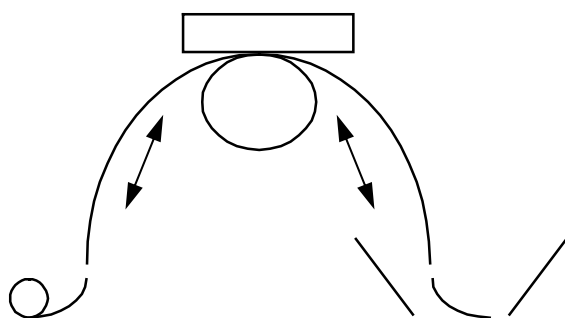
«+» печать с фотографическим качеством, подходит для электронной фотографии и создания изображений для целей медицины.

«-» непригодны для печати на обычной бумаге, расходные материалы стоят дорого, качество печати текста оставляет желать лучшего.

### ***Плоттеры (графопостроители).***

Графопостроители различают:

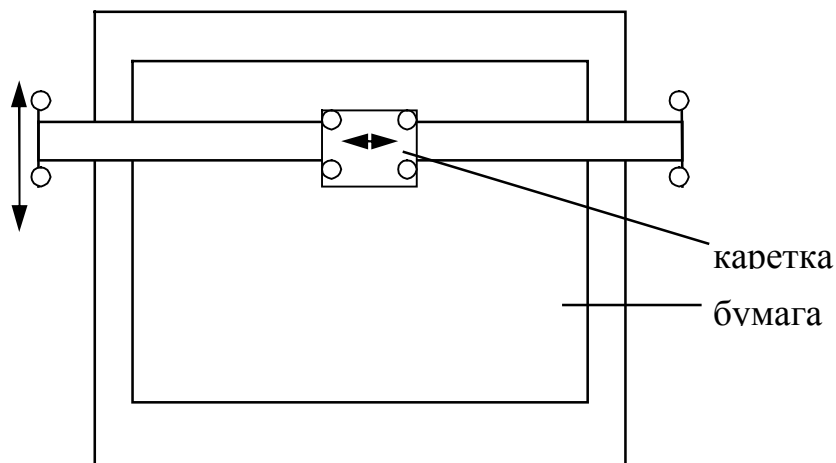
1. Рулонные.



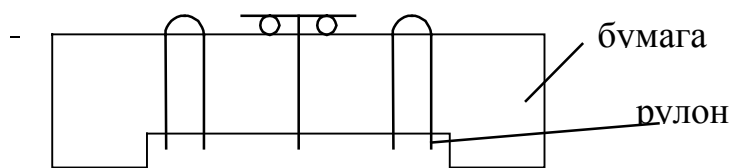
## 2. Планшетные.

Цена: \$ 4055 - 11902.

Ведущий производитель: Hewlett-Packard.

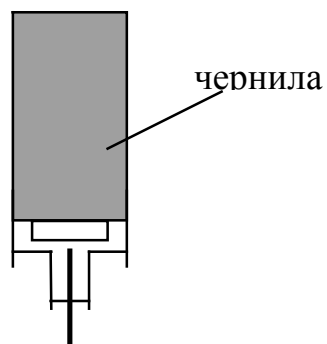


## 3. Листовые.



Существуют следующие системы (получения изображений) для плоттеров:

- *Рапидограф*



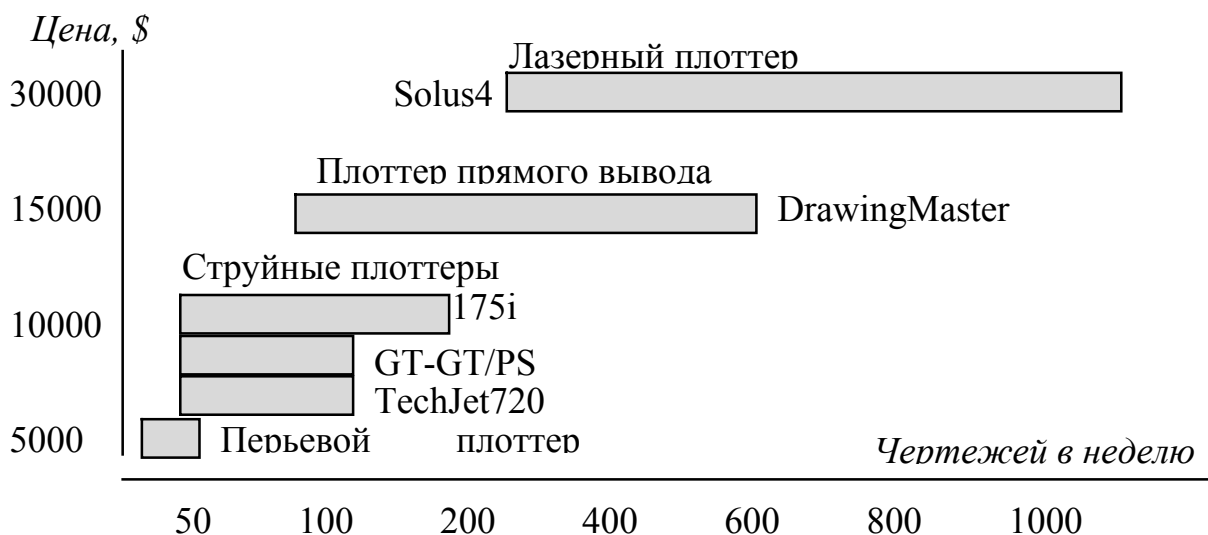
«+» самые качественные линии. Позволяет рисовать чернилами любого качества.

«-» требует обслуживания, надо промывать после работы и заправлять перед работой.



- *Фломастер*  
«+» позволяет прорисовку чертежа с очень высокой скоростью. Не является обслуживаемым расходным материалом.  
«-» ограниченная цветовая гамма (36 цветов), очень плохое смешивание цветов. Не дает линий одинаковой толщины.
- *Графитные карандаши*  
Используются для эскизных чертежей. Пишущий элемент на основе шариковой ручки.  
«-» Плохое качество.

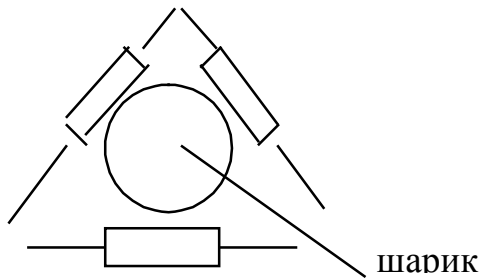
Плоттеры для построения цветного изображения бывают Color (цветные) и Color Capable (способные к цвету). Это далеко не одно и то же. Плоттеры “способные к цвету” используют алгоритмы одноцветной печати - четырехкратное повторение разными цветами, поэтому цвета попадают на носитель через значительные промежутки времени и не смешиваются, что не позволяет получить нужных цветовых оттенков. Цветной же плоттер наносит все четыре базовые цвета и работает существенно быстрее и качественнее.



Существуют еще экзотические плоттеры у которых вместо пишущего инструмента режущий. Иногда ставится фреза вместо режущего инструмента. Такая система позволяет вырезать 3-х мерные фигуры.

### *Подсистема ввода информация.*

*Мышь.*

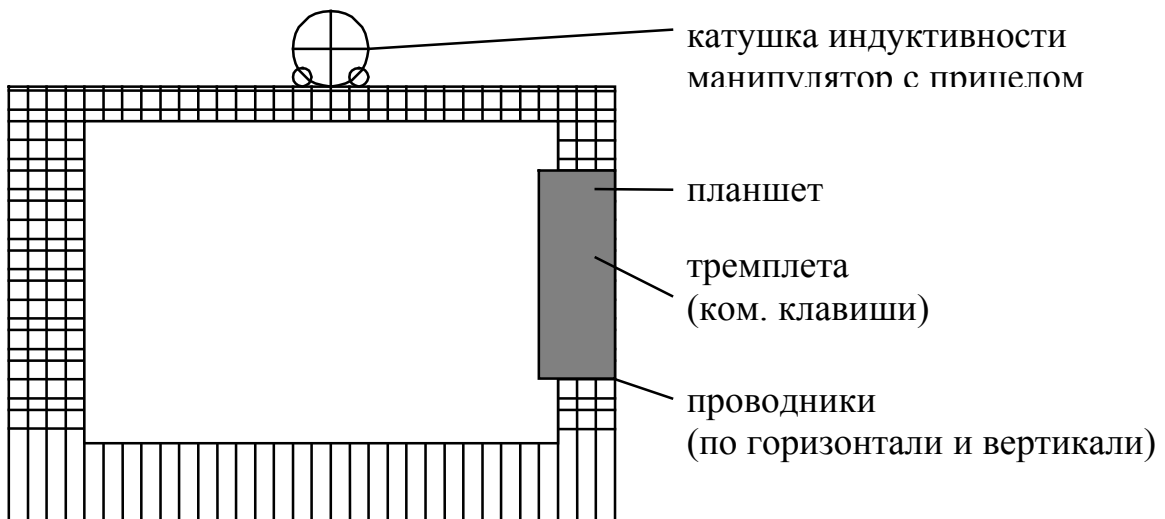


Присоединены потенциометры, которыми снимается напряжение пропорциональное углу поворота. Несколько клавиш (от 2 до 48).

### *Trackball.*

Модификация мыши, отличие - шарик сверху.  
«-» не отслеживает физических координат.

### *Дигитайзер.*



По величине наведенной индукции отслеживаются физические координаты. Обычно манипулятор имеет от 2 до 16 клавиш. Крепеж чертежа различный у разных фирм: вакуумный, электростатический, на магнитики.

Дигитайзеры бывают:

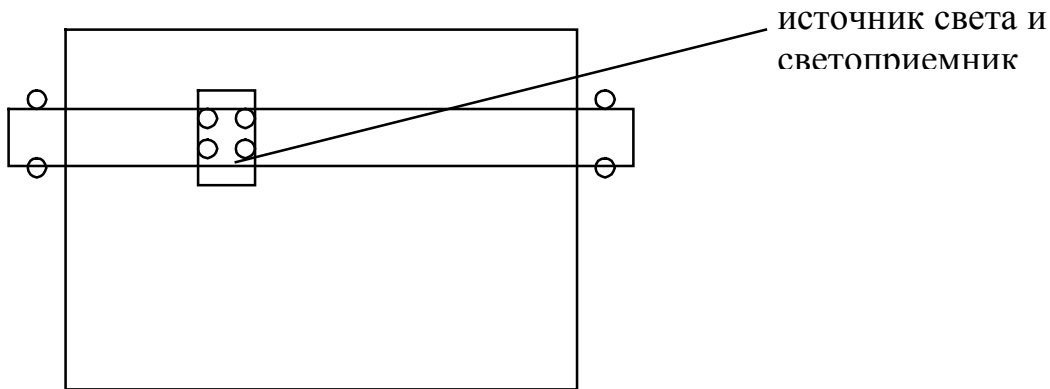
- индуктивные,
- лазерные,
- оптические.

«-» Ограничен размерами панели.

Ведущие производители: Graphics Tablet ScethPro, Hewlett Packard.

Цена: \$350.

### *Сканер.*



Источник света испускает свет на чертеж, который имеет разную отражательную способность в местах где есть линия и где ее нет. В зависимости от этого светоприемник принимает информацию и представляет ее в виде 1 или 0.

Различают:

- по количеству цветов (цветные и монохромные);
- по механике (страничные, планшетные, листовые).

Цена:

- монохром.(HP ScanJet 3P) \$ 460.
- цветной (HP ScanJet 4C) \$ 1230.

Модель	Цена, \$	Оптическое разрешение (горизонтально x вертикально), точек на дюйм	Максимальная длина листа, см	Габариты, см	Соединение с ПК
Страничные сканеры					
Storm Technology EasyPhoto SmartPage	299	300x300	35	28,7x11x12,7	SCSI - адаптер
Планшетные сканеры					
Hewlett Packard Scanjet 4p	499	300x600	35	38,7x58,4x12,7	SCSI - адаптер

### ***Математическое обеспечение АСУ.***

Теория оптимального управления, теория математического моделирования и многие другие математические теории входят в данную систему обеспечения.

## Моделирование систем.

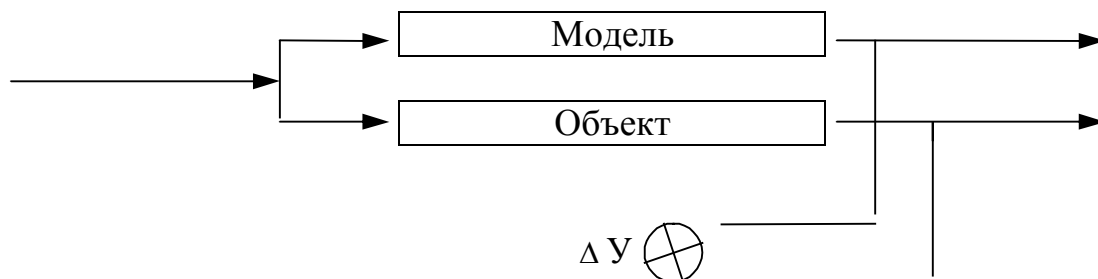
### Основные понятия.

Модель - объект-аналог, позволяющий заменить на определенных этапах исследования объект-прототип в рамках исследуемых характеристик.

Назначение: модели используют там, где затруднителен, невозможен или нерентабелен эксперимент над настоящим объектом.

### Этапы моделирования.

1. Составляется *описание* объекта.
2. Расчленение, *вычленение*. Выявляются существенные и несущественные черты.
3. *Формализация переменных*. Что вход, что выход.
4. *Формализованное построение модели*.
  - а) Построение модели в неявном виде. Нет прямой зависимости выходных переменных от входных. Решается только прямая задача.
  - б) Формирование модели в явном виде т.е. что будет с объектом если на вход по  
дать ... ?
5. *Параметрическая идентификация*, т.е. подбор параметров.



6. *Сопоставление с объектом*. Если результаты отрицательны то идет уточнение (весь алгоритм с первого пункта).

### Классификация моделей.

*Статические* модели подчиняются вероятностным законам.

*Вероятностные* законы - это такие законы, которые на одно входное воздействие могут дать несколько выходных.

*Динамический объект* - это такой объект, параметры которого изменяются во времени.

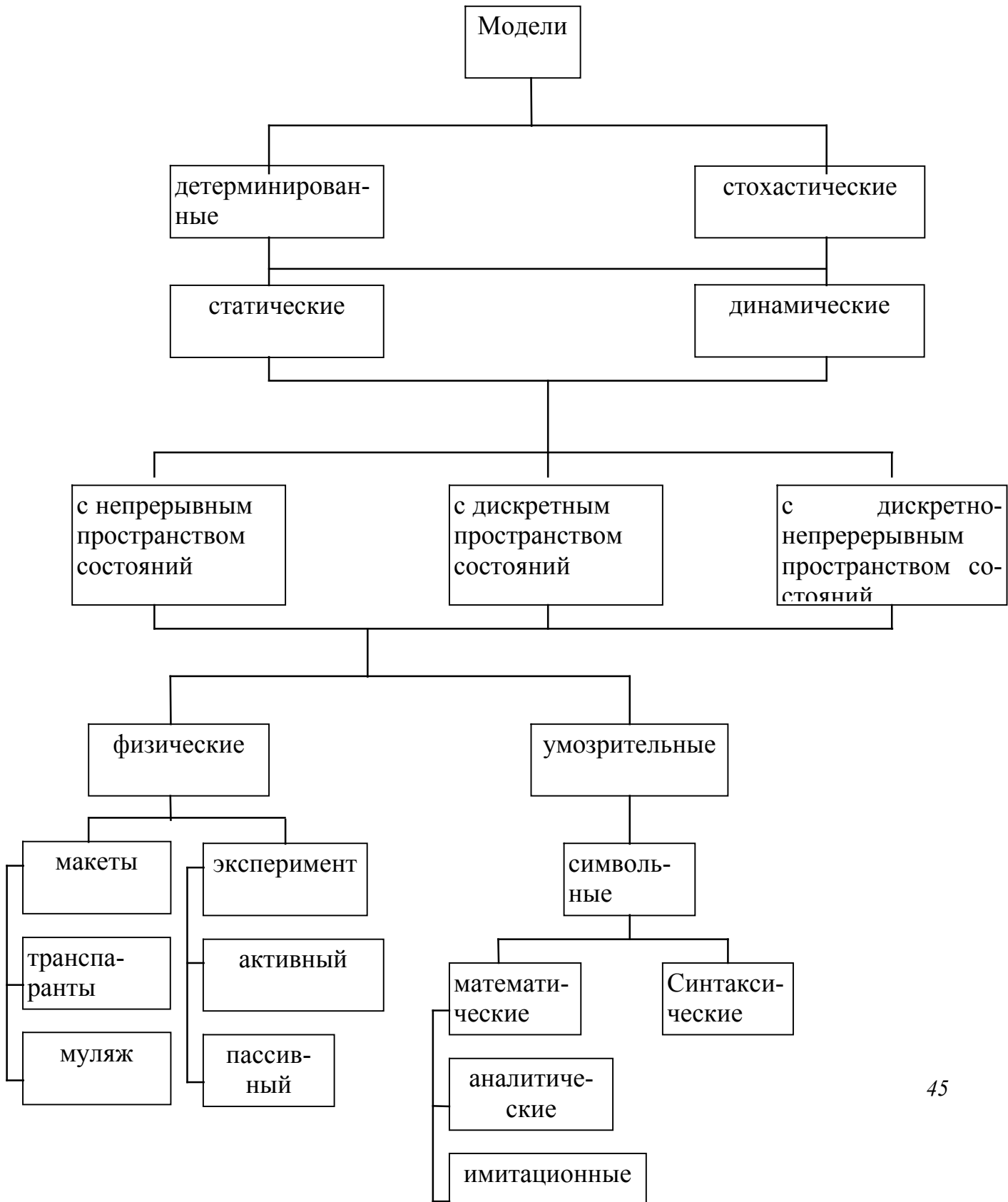
К объектам с *дискретным пространством* состояния относятся объекты, состояния которых изменяются скачками.

*Аналитические* - это модели, построенные на основе математических соотношений между входными и выходными переменными.

**Синтаксические** модели позволяют моделировать статические объекты. В этом случае каждому состоянию объекта соответствует определенный символ.

**Активный эксперимент** - это такой эксперимент, при котором воздействия на объект направляются целенаправленно. Примером пассивного эксперимента является наблюдение.

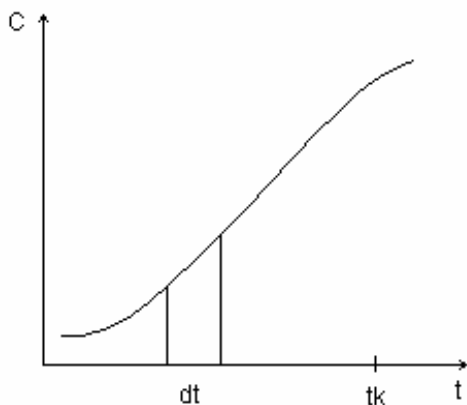
**Имитационные** - это модели основанные на модульном подходе. Когда каждый объект в системе описывается своей моделью а вся совокупность объектов описывают систему в целом.



## Основные принципы моделирования.

### I. Принцип $\Delta t$ .

В соответствии с этим принципом весь интервал времени моделирования разбивается на одинаковые участки, равные минимальной величине, за которую могут произойти существенные изменения. Чем меньше  $\Delta t$  тем



точнее наши вычисления. У нас должна быть модель объекта явно зависящая от времени.

«-» Точность зависит от величины шага по времени. Чем меньше шаг тем больше точность но чем меньше шаг тем больше время расчета.

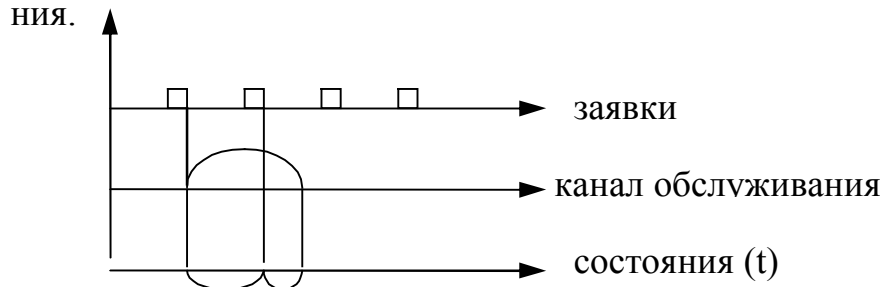
«->» Во многих случаях ошибка моделирования накапливается (носит интегральный характер).

Когда объект носит дискретный характер то метод становится неудобен. Зачем рассчитывать состояния в то время когда ничего не происходит.

### II. Принцип особых состояний.

Выделим моменты времени, в которые происходит существенное изменение состояния. Было обнаружено, что во многих случаях такие состояния заранее известны. Так появился метод особых состояний.

**Особым** состоянием будем называть одно из состояний, в котором может оказаться объект. В этом случае наблюдение за объектом происходит не в каждый промежуток времени, а только в моменты смены состояния.

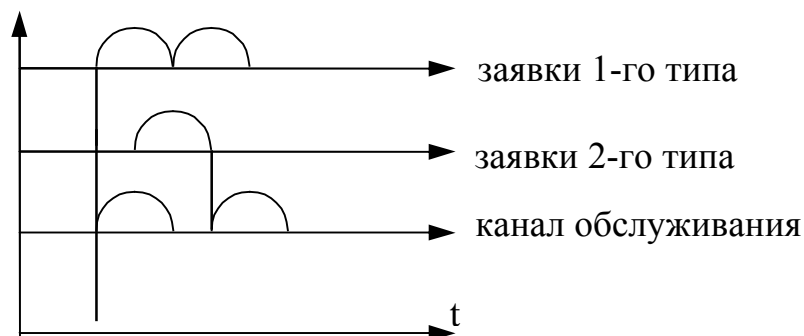


*Алгоритм:*

1. Моделируют время прихода заявки.
2. Выбирают заявку с минимальным временем прихода.
3. Определяют наличие свободного канала.
4. Определяют тип заявки с минимальным временем.
5. Определяют процедуру обслуживания в зависимости от типа заявки.
6. Определяют время конца обслуживания заявки.
7. Моделируют время прихода следующей заявки.
8. Все повторяется.

*Задача:*

Необходимо промоделировать процесс формирования очереди в парикмахерской если известно, что раз в 20-25 мин. приходят клиенты первого типа, которым необходимо только побриться; раз в 30-35 мин. Приходят клиенты второго типа, которым требуется подстричься. Имеется один канал обслуживания. Время обслуживания клиентов первого типа - 15 мин., клиентов второго типа - 20 мин.



Время поступления заявки определяем следующим образом:

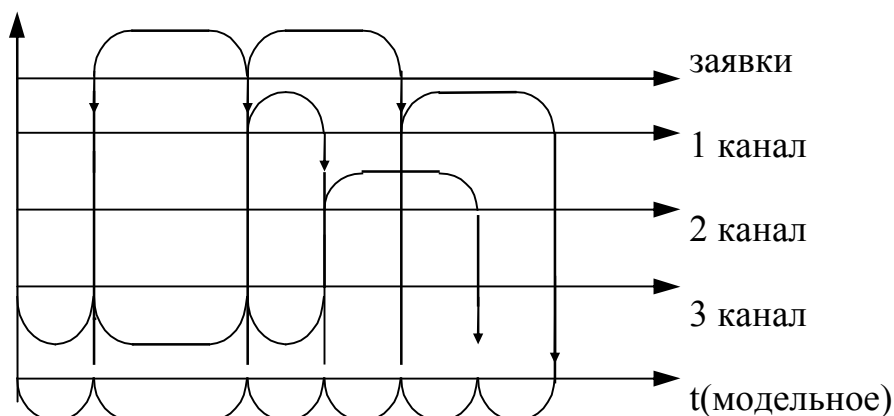
$$(RND) * 5 + 20 + t_{i-1}$$

### III. Принцип последовательной проводки заявок.

*Особенность:* данный принцип, как и принцип особых состояний, позволяет перемещаться только по существенным моментам времени, но в отличие от него отслеживает прохождение заявки от первой операции обслуживания до последней. Модельное время все время возвращается в момент прихода следующей заявки.

*Задача:*

На техническое обслуживание поступают с интенсивностью раз в 2-2,5 часа автомобили, при этом они проходят на 1-ой операции технический осмотр: в среднем в течении 1 часа  $\pm$  20 мин., на 2-ой операции - регулировочные работы : в среднем 40 мин.  $\pm$  10 мин., на 3-ей операции - смазочные работы : 1 час  $\pm$  30 мин.



*Алгоритм:*

1. Выбираем время прихода заявки.
2. Определяем свободен 1-ый канал или нет.
3. Если свободен, то модельное время переводим на момент прихода заявки.
4. Вычисляем время окончания обслуживания на 1-ой операции.
5. Модельное время переводим на этот момент.
6. Определяем свободен ли канал обслуживания на 2-ой операции.
7. Ставим заявку на обслуживание на 2-ой операции.
8. Вычисляем время окончания обслуживания на 2-ой операции.
9. Переводим модельное время в момент окончания обслуживания на 2-ой операции.
10. Определяем свободен ли 3-ий канал.
11. Ставим заявку на обслуживание на 3-ей операции.
12. Вычисляем время окончания обслуживания на 3-ей операции.
13. Переводим модельное время на конец обслуживания на 3-ей операции.
14. Выбираем следующую заявку из входного потока.
15. Модельное время возвращаем в момент прихода этой заявки.

Если на каком-либо этапе канал занят, то вычисляется время ожидания, после чего заявка ставится на обслуживание, а модельное время переводится в этот момент.

Данный метод используется :

1. В тех случаях, когда необходимо проследить движение заявки от начала до конца.
2. В тех случаях, когда количество типов заявок небольшое.
3. В тех случаях, когда маршрут обслуживания заявок сложный.

## **Методы оптимизации.**

### **Основные понятия.**



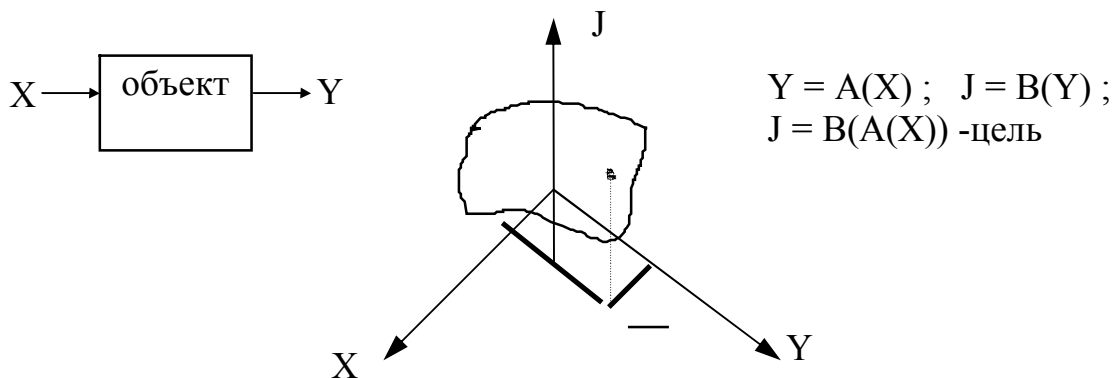
Методы оптимизации призваны найти наилучшее (оптимальное) решение в случае известных целей и сформулированных математически и в случае известных ограничений.

Математическое описание цели называется **целевой функцией**. Объект характеризуется несколькими параметрами. Часть из них являются **наблюдаемыми**. Часть параметров можно определить, зная величины наблюдаемых, они называются **вычисляемыми**.

На основе вычисляемых и наблюдаемых параметров можно сформулировать функцию, оценивающую степень достижения цели.

Если известна модель объекта (т.е. связь входных и выходных переменных), то можно спрогнозировать к каким последствиям приведет то или иное управляющее воздействие. А значит можно оценить к каким изменениям целевой функции приводит изменение управляющих переменных.

Далее остается найти такие значения управляющих переменных, которые доставляют экстремум целевой функции.



### **Методы оптимизации.**

Все методы оптимизации делятся в зависимости от вида целевой функции на:

- линейные
- нелинейные

И в тех и других методах отдельно выделяют целочисленное программирование.

Особенностью линейных методов является то, что у линейной функции экстремум всегда лежит на границе области допустимых значений.

При нелинейном программировании используются другие методы поиска экстремума (например метод итераций).

Большинство из методов численных оптимизаций находят не абсолютный экстремум, а первый попавшийся локальный.

Как отдельный класс стоят методы случайного поиска. Большинство из них ориентировано на искусственном сужении области допустимых значений.

## *Программное обеспечение.*

### *Классификация ПО.*

Можно выделить две основные части программного обеспечения:

1. Обслуживающие программы.
2. Программы АСУ.

### *Обслуживающие программы.*

1. Проблемно-ориентированные:

- с жесткой моделью;
- с настраиваемой моделью.

2. Универсальные (ОС)

Если известен определенный тип задач и конкретные периферийные устройства, на которых их решать, то вместо ОС используют проблемно-ориентированную систему.

**Проблемно - ориентированной системой** называется комплекс программ, обеспечивающий эффективную работу пользователя и ЭВМ в рамках узкого класса задач при конкретном перечне периферийных устройств.

**ОС** называется комплекс программ, используемый для работы пользователя и ЭВМ при решении задач различной постановки и целей.

**ОС включает:**

- управляющую часть;
- сервисную часть.

Управляющая часть состоит:	К сервисным программам относятся:
<ul style="list-style-type: none"> <li>• Программа первоначальной загрузки</li> <li>• Программа управления задачами</li> <li>• Программа управления заданиями</li> <li>• Программа управления данными</li> <li>• Супервизор</li> <li>• Библиотекарь</li> <li>• Диспетчер и др.</li> </ul>	<ul style="list-style-type: none"> <li>• Набор утилит (создание файлов, добавление, удаление, корректировка и т.д.)</li> <li>• Трансляторы</li> <li>• Интерпретаторы</li> <li>• Компиляторы</li> <li>• А также программы, выполняющие специфические функции</li> </ul>

## *Характеристики ОС.*

### *1. MS DOS.*

Эта ОС является фактически промышленным стандартом для 16-разрядных ПЭВМ на базе МП типа 8086/80186/80286. MS DOS является однопользовательской ОС. Однопользовательские ОС имеют более развитые средства доступа ко всем аппаратным компонентам машины, развитую файловую систему, основанную на иерархической структуре каталога, с удобным командным языком.

*Основные достоинства MS DOS :*

- Организация многоуровневых каталогов.
- Возможность работы со всеми последовательными устройствами как с файлами.
- Подключение дополнительных драйверов внешних устройств.
- Возможность запуска фоновых задач одновременно с диалоговой работой пользователя.
- Развитый командный язык.

Для работы DOS требуется до 60 Кбайт оперативной памяти.

Модульность DOS позволяет существенно упростить разработку отдельных частей системы, модифицировать при необходимости отдельные модули.

*Основные модули DOS :*

- Блок начальной загрузки (БНЗ)
- Базовая система ввода-вывода (БСВВ)
- Модуль обработки прерываний (МОП)
- Командный процессор
- Утилиты

БНЗ - небольшая программа, обеспечивающая считывание с диска в оперативную память двух других частей DOS : БСВВ и МОП. Модуль размещается на системном диске всегда на первом секторе нулевой стороны диска и занимает один сектор емкостью 512 байт. Модуль просматривает каталог системного диска с целью обнаружения двух определенных частей на первых двух позициях системного диска или выдает сообщение при их отсутствии.

Базовая Система Ввода-Вывода - программа, доступ к которой производится только из МОП. Она состоит из набора подпрограмм(файлов), обслуживающих ввод и вывод на внешние устройства. БСВВ выполняет помимо этого функции завершения загрузки DOS в оперативную память, после чего передает управление загруженному с диска командному процессору, завершая свою работу.

Модуль Обработки Прерываний образует в отличии от БСВВ верхний уровень системы и взаимодействует с большинством прикладных программ. Подпрограммы модуля обеспечивают работу файловой системы, периферийных устройств, обслуживание ситуаций, связанных с заверше-

нием программ, их искусственным прерыванием и обработкой ошибок. Обращение к МОП осуществляется командами вызова прерываний с номерами 32-39. Прерывание с номером 33 является комплексным, объединяющим множество других функций или операций DOS по обслуживанию стандартных устройств и файловой системы.

Файл с Командным Процессором может занимать на системном диске любое место и трактуется как обычная программа. При загрузке в ОЗУ командный процессор распадается на две части в зависимости от расположения - на резидентную и нерезидентную, располагаемую в старших адресах ОЗУ.

Резидентная часть содержит подпрограммы стандартной обработки прерываний с номерами 34-36, а также подпрограммы подзагрузки нерезидентной части и инициализации, обрабатывающие специальный файл автозапуска.

Нерезидентная часть содержит подпрограмму обработки команд DOS, поступающих с терминала или из командных файлов, загрузчик внешних программ, обеспечивающий чтение очередной программы с диска в ОЗУ, настройку адреса и переход на ее выполнение.

Основные функции командного процессора: прием, анализ и исполнение команд пользователя, обращенных к DOS; обработка командных файлов.

Команды, называемые командами DOS, служат основным средством общения пользователя с DOS. После выполнения прикладной программы они позволяют вновь обрабатывать команды, поступающие от пользователя или из командного файла.

Команды DOS дают возможность готовить диски для работы, копировать файлы, переименовывать их, удалять из каталогов, сменять текущий каталог и накопитель, изменять режим работы дисплея, выводить содержимое текстовых файлов на экран дисплея, печать или в коммуникационный канал, и т.д.

### *Основные команды MS DOS:*

#### Управление командными файлами

GOTO : Переход на метку в командном файле

IF : Проверка условия

FOR : Повторяющееся исполнение со сменными аргумен-

тами

COMMAND: Вызов новой копии командного процессора

#### Формирование операционной среды

PATH : Задание альтернативных маршрутов для поиска программ

SET : Задание параметров операционной среды

PROMT : Задание "приглашения" для ввода команд  
CTTY : Установка стандартных устройств ввода и вывода  
ASSIGN : Установка логических имен накопителей

#### Управление файловой системой

DIR : Выдача каталога файлов  
CHDIR : Смена текущего каталога  
MKDIR : Создание нового каталога  
RMDIR : Удаление каталога  
REN : Переименование файлов  
COPY : Копирование файлов

#### Конфигурирование

BREAK : Установка/отключение прерываний по Ctrl+Break  
BUFFERS : Установка числа буферов для обменов с дисками  
FILES : Установка числа одновременно открытых файлов  
DEVICE : Подключение к DOS новых драйверов внешних устройств  
SHELL : Задание имени нового командного процессор

## 2. UNIX.

Система UNIX первоначально создавалась для мини-ЭВМ PDP-7. Позже она была модифицирована для работы на других мини-ЭВМ (семейства PDP-11 и для семейства VAX), а затем и на микро-ЭВМ. Это повысило популярность системы UNIX.

ОС UNIX отличается от других операционных систем по следующим основным характеристикам:

- Структурированная многоуровневая файловая система.
- Средства, позволяющие многим пользователям одновременно работать на одной и той же ЭВМ, что делает систему UNIX многопользовательской.
- Возможность для любого пользователя, работающего на ЭВМ, выполнять одновременно более одной программы.
- Механизмы, позволяющие одной программе передавать ее результаты непосредственно в другую программу (поэтому необязательно использование дополнительного пространства памяти).
- Возможность для любого пользователя переводить выдачу результатов с одного периферийного устройства на другое.
- Встроенный командный интерпретатор и язык shell.
- Встроенный командный интерпретатор и язык shell.
- Структурированный язык Си для системного программирования.
- Расширенные средства для написания и разработки программ на Си и других языках программирования.

- Расширенные средства для ввода, внесения изменений и обработки текста и подготовки его к печати.

#### *Функционирование системы UNIX.*

Когда запускается прикладная программа под управлением ОС UNIX, то совместно работают три программы:

- Ядро
- Интерпретатор shell
- Сама прикладная программа

Программа, которая непосредственно взаимодействует с вычислительной машиной, называется *Ядром*. Ядро выделяет такие аппаратные ресурсы ЭВМ, как накопители на магнитных дисках, лентопротяжные механизмы и оперативную память для каждого программного запроса.

Интерпретатор SHELL распознает и выполняет команды, вводимые с клавиатуры, готовит другие программы к запуску. Его принято представлять как "средний слой" между ядром и прикладными программами, т.к. интерпретатор SHELL играет исключительно важную роль при взаимодействии ЭВМ с пользователем.

#### *Составные части ОС UNIX.*

Файловая система состоит из файлов трех типов:

- директории
- обычные файлы
  - текстовые
  - исполняемые
- специальные файлы (каждый специальный файл соответствует внешнему устройству)

Файловая система UNIX имеет 2 особенности:

1. Занимает на диске весь свободный объем и не разбивается на части.
2. Файловая система представляет собой логическую структуру, не зависящую от логического носителя.
3. Языки-оболочки. Предназначены для написания программ из команд.
4. Набор команд.

В полный список входит более 400 команд. Есть возможность писать собственные команды.

#### *Основные команды работы с файловой системой :*

pwd - выдать на терминал абсолютное имя текущего директория  
 cd [имя] - перейти в директорий с указанным именем  
 ls [флаги] [имя] - просмотреть содержимое директория  
 rm [флаги] [имя] - удалить директорий  
 find [где искать] [флаги] - поиск файла

## ***РВ.***

Операционная система РВ используется в качестве базовой в различных системах реального времени, а также в качестве универсальной инструментальной многопользовательской системы для разработки и отладки программ.

Система характеризуется эффективным использованием ресурсов вычислительного комплекса, быстрой реакцией на события реального времени и обеспечивает следующие возможности :

- Мультипрограммный режим выполнения задач в реальном времени;
- Приоритетную диспетчеризацию задач и распределение ресурсов на основе приоритетов;
- Управление файлами на различных магнитных накопителях;
- Использование и поддержку широкого набора внешних устройств из номенклатуры СМ ЭВМ;
- Многопользовательский режим работы со многих терминалов;
- Включение в систему пользовательских драйверов для нестандартных внешних устройств;
- Динамическое распределение и уплотнение памяти;
- Динамическую загрузку задач с диска и выгрузку их на диск;
- Работу с расширенной памятью емкостью до 1920 Кслов.
- 

Операционная система реального времени состоит из следующих основных частей:

- Ядро
- Файловая система
- Драйверы периферийных устройств
- Программа связи с оператором
- Системные обслуживающие программы
- Системы программирования

*Ядро ОС РВ:*

Ядро является резидентной в основной памяти составляющей ОС РВ, реализующей функции управления задачами, основной памятью, операциями ввода-вывода.

*Файловая система:*

Файловая структура представляет собой способ организации данных на томах. Томами с файловой структурой ОС РВ являются магнитные носители (диски или ленты), инициализированные по команде INI программы связи с оператором. В результате инициализации на томе создаются стандартные системные файлы : индексный, главный каталог файлов (MFD), файл распределения памяти на диске (BITMAP.SYS), список дефектных блоков (BADBLK.SYS), файл для выгрузки задач (CORIMG.SYS). Индексный файл содержит управляющую информацию о томе и блоки заголовков для всех файлов, находящихся на томе. Каждому файлу соответ-

вует свой блок заголовка, содержащий информацию о файле, необходимую управляющей программе для доступа к нему. Главный каталог содержит список всех каталогов файлов пользователей (UFD).

*Драйверы периферийных устройств:*

Управляют работой внешних устройств, обрабатывают прерывания и сбои внешних устройств.

*Программа связи с оператором:*

Связь оператора с системой осуществляется через команды, вводимые с терминала и обрабатываемые программой связи с оператором (MCR) или интерпретатором командных строк (DCL).

Команды программы связи с оператором делятся на следующие группы:

- команды инициализации;
- команды управления внешними устройствами;
- команды управления выполнением задач;
- информационные команды;
- команды эксплуатации системы;
- команды для обеспечения многопользовательской защиты;
- команды интерпретатора командных строк.

Преимуществом DCL по сравнению с MCR являются более широкий набор команд и средств для управления выполнением задач в системе, простота в применении и гибкость командного языка.

*Системные обслуживающие программы:*

В соответствии с функциональным назначением они делятся на следующие группы:

- работы с томами и файлами,
- редактирования и форматирования текстов,
- обслуживания библиотек,
- корректировки объектных модулей и образов задач.

Программы работы с томами и файлами включают в себя программы :

- Работы с файлами (PIP), обеспечивающую обслуживание файлов и каталогов (копирование, переименование, удаление, распечатку и др.);
- Распечатки содержимого файлов и томов в различных форматах (DMP);
- Сравнения символьных файлов (CMP);
- Копирования и уплотнения тома (DSC), обеспечивающую копирование дискового тома на магнитную ленту или другой диск с уплотнением файлов в целях устранения фрагментации свободного пространства;



- Форматирования дисков (FMT);
- Проверки томов (BAD), позволяющую определить на дисковом пространстве дефектные блоки;
- Сохранения и восстановления томов (PRESRV);
- И др.

Программы редактирования и форматирования текстов содержат :

- Диалоговый строчный редактор символьных файлов (EDI);
- Пакетный редактор (SLP);
- Диалоговый экранный редактор текстовой документации (RETEX);
- Программу форматирования текстов (DOC);
- Программируемый текстовый редактор (TECO);
- Универсальный текстовый редактор (EDT).

Программы обслуживания библиотек (LBR) обеспечивает возможность создания и модификации библиотек макроопределений и объектных модулей, а также универсальных библиотек, содержащих однотипные модули произвольного вида.

Программы корректировки включают в себя программу корректировки объектных модулей (PAT) и образов задач (ZAP), позволяющую вносить изменения в файл образа задачи на диске.

#### *Системы программирования:*

Системы программирования ОС РВ включают в себя следующие языки и трансляторы :Макро, Фортран-IV, Кобол, Паскаль, Бейсик, Бейсик-Р2.

#### Основные команды ОС РВ:

DCL	MCR	Выполняемая функция
HELP	HELP	Запрос выдачи справочной информации о работе с системой
LOGIN LOGOUT MOUNT	HELLO BYE MOUNT	Вход в систему Выход из системы Запрос доступа к требуемому внешнему устройству
CREATE	EDT	Варианты вызова программ редактора текстов
FOR	FOR	Вызов компилятора языка Фортран Вызов макроассемблера с языка Макро
MACRO	MACRO	Вызов компоновщика (построителя задач)
LINK	TKB	

DCL	MCR	Выполняемая функция
RUN  ABORT  PIP  COPY	RUN  ABORT	Инициализация выполнения задачи Прекращение выполнения задачи Вызов подпрограммы обмена между периферийными устройствами, которая позволяет выполнять : копирование файлов;
TYPE  PRINT DELETE RENAME		вывод содержимого файлов на экран; вывод содержимого файла на АЦПУ; удаление файлов; переименование файлов;
DIR  PURGE  SHOW  DEVICE	SET  PB	вывод оглавления каталога или тома; удаление "старых" версий файла Просмотр установленных параметров ОС

### ***РАФОС.***

Операционная система с распределенными функциями РАФОС предназначена для использования в различных системах управления технологическими процессами и научным экспериментом, где сочетается решение задач реального времени с многопользовательской работой в режиме разделения времени по подготовке программ. Система обеспечивает работу в одно-, двух- и мультипрограммном режиме.

РАФОС представляет собой базовую ОС, ориентированную на использование в специальных многопроцессорных комплексах, в состав которых кроме основного процессора могут входить несколько спецпроцессоров или микропроцессоров. Такие специализированные комплексы дают возможность существенно повысить производительность при решении специальных классов задач. Это достигается за счет разделения функций между различными процессорами, ориентации процессоров на выполнение специальных функций, параллельного функционирования всех процессоров комплекса.

Существуют всевозможные версии ОС РАФОС, такие, как : ФОДОС, ФОБОС, ОСДВК и т.д. Базовой для всех этих операционных систем, в том числе и для РАФОС, является операционная система РТ-11.

Основным достоинством РАФОС является простота и удобство в эксплуатации. Работа с системой не требует высокой квалификации программиста.

РАФОС характеризуется повышенной реакцией на обработку прерываний по сравнению с другими операционными системами.

*ОС РАФОС имеет следующую структуру:*

- управляющая система;
- системные программы;
- системы программирования;
- дополнительные компоненты.

*Управляющая система.*

Ядром управляющей системы ОС РАФОС является монитор, который создается во время генерации системы. Монитор организует ввод/вывод в системе, ведет системы файлов на внешних запоминающих устройствах, осуществляет диалог с оператором, диагностику ошибок и т.д.

В ОС РАФОС имеется пять типов мониторов :

- исполняющий монитор реального времени (RM);
- однозадачный монитор реального времени (SJ);
- фоновый/оперативный монитор реального времени (FB);
- монитор управления расширенной памятью (XM);
- многопользовательский монитор разделения времени (TS).

Управляющая система включает набор драйверов для обслуживания внешних устройств. Драйвер представляет собой программную компоненту монитора и хранится в отдельном файле на системном запоминающем устройстве. Это позволяет легко включать в состав операционной системы драйверы для новых внешних устройств. Все общения операционной системы с внешними устройствами идет только через их драйверы.

*Системные программы.*

Системные программы, входящие в состав операционной системы, позволяют выполнять разнообразные функции по обслуживанию системы. Ниже приведены некоторые из этих функций :

- Копирование файлов с одного носителя на другой, объединение, переименование и удаление файлов (PIP).
- Распечатка каталогов внешних запоминающих устройств с предварительной сортировкой по датам создания, типам файлов, местоположению на носителе и т.п. (DIR).
- Компоновка объектных модулей и библиотек в выполняемые программы (LINK).
- Создание, коррекция, извлечение и обслуживание библиотек объектных модулей; создание макробиблиотек (LIBR).

- Сравнение двух текстовых файлов с выдачей различия между ними на терминал или устройство печати (SRCCOM).

#### *Системы программирования.*

В состав ОС РАФОС входит большое количество систем программирования, но самыми распространенными по применению являются следующие :

- МАКРО (Макроассемблер)
- Фортран-IV (компилятор)
- Бейсик (интерпретатор)
- Паскаль

Работа с системами программирования может осуществляться также на уровне команд монитора.

Пользователь управляет работой операционной системы с помощью команд монитору. Команды монитору - это язык, позволяющий с системного терминала получить справочную информацию о состоянии системы, управлять системой.

#### *Основные команды монитора:*

INITIALIZE : Выполняет инициализацию каталога внешнего устройства.

SQUEESE : Выполняет перераспределение дискового пространства с целью создания наибольшей свободной непрерывной области.

DIRECTORY : Распечатывает запрашиваемую информацию об устройстве, файле или группе файлов, взятую из каталога.

RENAME : Переименовывает один или более файлов.

DELETE : Удаляет указанные файлы.

COPY : Выполняет операции по передаче файлов, копированию носителей и служебные функции.

RUN : Загружает и запускает на выполнение программу с указанного устройства.

START : Запускает загруженную в память программу с указанного адреса (в восьмеричном коде).

FORTRAN : Вызывает транслятор с языка Фортран для трансляции одного или нескольких файлов.

MACRO : Вызывает Макроассемблер для трансляции одного или нескольких файлов.

LIBRARY : Позволяет создавать и модифицировать файлы библиотек и распечатывать их каталоги.

и другие.

### ***Пользовательские программы.***

В них входят программы АСУ.

*Программы АСУ :*

1. Программы функциональных задач. (Расчет зарплаты, управление сбытом, бухучет)
2. Программы общего назначения:
  - 1) Редакторы текстов:
    - Издательские системы (Ventura Publisher);
    - Текстовые процессоры (Word Perfect, MS Word);
    - Документаторы (ЛЕКСИКОН, ChiWriter) ;
    - Текстовые редакторы для программ (Pe-2, MultiEdit).
  - 2) Электронные таблицы (Excel, SuperCalc, Lotus 1,2,3)
  - 3) Информационно-поисковые системы и банки данных (FoxPro, Clipper, dBase, Oracle)
  - 4) Пакеты специфических функций :
    - Художественно-графический (Dr.Hallo, PaintBrush, Adobe Illustrator);
    - Системы автоматизации графических работ (AutoCad);
    - Пакеты математической обработки (MathCad, Derive, Mathematica);
    - Пакеты оформления химических работ.

## ***Информационное обеспечение.***

### ***Общие понятия и состав.***

Информационное обеспечение состоит из справочников, кодификаторов, идентификационных справочников, словарей синонимов, систем документооборота, систем кодирования, систем унифицирования форм документов, экранных форм, структур баз данных и информационных массивов.

***Информацией*** называется сообщение, снижающее неопределенность в той области, к которой оно относится.

Информационное обеспечение выделяет два класса задач :

1. Внешнее информационное обеспечение. То что за пределами автоматизированной системы.
2. Внутреннее информационное обеспечение.

### ***Внешнее информационное обеспечение.***

#### ***Состав.***

1. Описание объектов и их свойств.
2. Классификация объектов. Заключается в том, что все объекты в зависимости от их свойств разбиваются на классификационные группы.
3. Задача идентификации. Под идентификацией понимается определение отношения анализирующего объекта к какой-либо классификационной группе.

4. Кодирование информации. Приведение информации в компактной форме представления. Управление решением задач идентификации и поиска информации.
5. Задача организации внешнего документооборота. Обмен информации на данном этапе в основном идет через *документы*. Для каждого документа есть технологический маршрут (инструкции по заполнению и обработке документа). Составление форм документов и последовательности его обработки это задача специалиста по информационным технологиям.

### **Системы кодирования.**

1. **Порядковая** система. Все объекты, события, элементы в качестве кода получают порядковый номер.
  - «+» Компактность кода
  - «+» Простота, понятность.
  - «-» Сложность кодирования вновь появляющихся объектов. Необходимость полной перекодировки системы при добавлении или удалении номера а значит не один из элементов не может иметь постоянного номера.
    - Аникин №1
    - < Антонов №1а
    - Петров №2
  - «-» Система не характеризует объект и не позволяет его идентифицировать
2. **Серийная** система. На каждую классификационную группу выделяется серия номеров. Внутри серии нормальное порядковое кодирование.
  - «+» По номеру серии легче идентифицировать объект (413 аудитория на 4-м этаже). Сужается круг поиска а значит и время на поиск.
  - «+» Не требуется полной перекодировки объекта в случае появления новых объектов.
  - «-» Внутри системы все равно порядковое кодирование.
  - «-» Хорошо идентифицирует но только по одному параметру.
3. **Позиционная** система. На каждую классификационную группу отводится своя позиция.
 

*Пример:*

специальность (1-3 позиции)	курс (4-5 поз.)	группа (6 поз.)	№ п./п. (7-8 поз.)
110	11	3	25

  - «+» Улучшается процедура идентификации. По коду можно много сказать об объекте.
  - «-» Чем сложнее код тем больше знаков он занимает.
4. Система **повторения** (164x54x90x2). Код повторяет некоторые характеристики объекта.

- «+» Простота кода.
  - «+» Легкость идентификации.
  - «-» Несколько объектов могут иметь одинаковый код.
5. **Комбинированная** система. Включает в себя элементы всех систем.  
*Пример:* АСУ-94-2 16
- «+» Эта система решает проблемы идентификации, упрощает поиск, позволяет однозначно характеризовать объект.
  - «-» Чтобы что то закодировать должна быть построена система кодирования, мы должны выполнить классификацию всех объектов, для классификации объектов нам необходимо описание всех возможных объектов.

### ***Системы информации.***

Если модули системы не взаимосвязаны, то может понадобиться многократный ввод разными пользователями одной и той же информации, что часто приводит к появлению ошибок, несоответствий, раздражению пользователей. Для эффективности работы системы необходимо предусмотреть одноразовый ввод информации.

Для этого необходимо:

1. Исходя из потребности всех задач, разработать документы и системы документооборота, которые позволяли бы агрегировать информацию. Но в этом случае несколько усложняются формы документов и правила их заполнения.
2. Модуль входного интерфейса нужно отделить от своей задачи и сделать его общим для нескольких задач.
3. Желательно также соответствие типа носителя характеру используемой информации.

### ***Внутреннее информационное обеспечение.***

#### ***Состав.***

1. Преобразование данных из внешней формы во внутреннюю.
2. Задача распределения информации между машинными носителями.
3. Распределение информации внутри носителя.
4. Хранение поиск и обработка информации. Записи (банки данных) могут иметь сетевую, древовидную и реляционную структуру. Хранение упирается в вопросы защиты. Хранит нужно так чтобы обеспечивать соответствие характера информации носителю.
5. Преобразование информации из внутреннего представления во внешнее.

#### ***Требования к информации.***



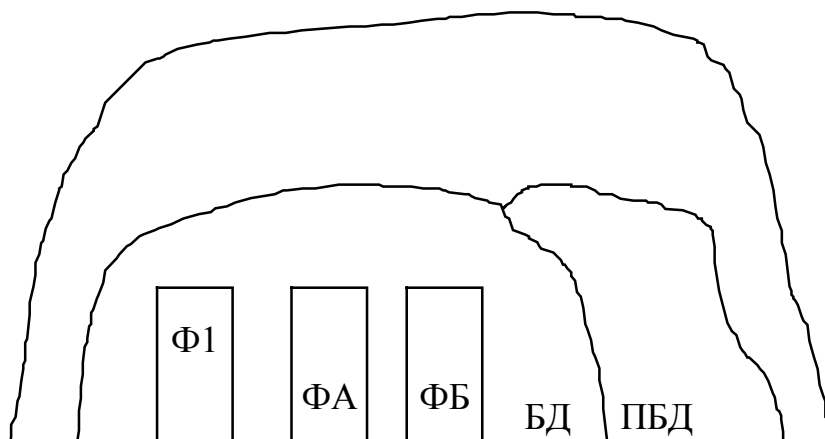


### **Физические структура размещения информации.**

1. Метод **последовательного размещения**. Записи размещаются одна за другой.
  - «-» Сложно вставлять новые записи - приводит к регенерации базы данных.
  - «+» Самое эффективное использование пространства памяти, самый быстрый доступ к информации.
  - «+» Понятность пользователю принципа размещения.
2. Размещение **по вычисляемому ключу**. После каждой записи оставляется место для нескольких (трех) страховочных записей. Например писать через одну две записи но возможны и более сложные ключи.
  - «+» Есть резервы для внесения новых записей без регенерации базы, ускоряется поиск при решении запросных задач, не нужно создавать таблицу адресов.
  - «-» Много неиспользуемого пространства.
  - «-» Есть затраты на поиск (вычисление) по формуле.
  - «-» Если надо записать еще элемент а все резервы заполнены то составляется таблица исключений.
3. Метод **рандомизации**. Все записи размещаются на всем пространстве памяти с помощью датчика случайных чисел, имеющего равномерный закон распределения. Последовательность записей и их адреса хранятся в специальной таблице. Роль разместителя берет на себя ОС но при организации БД это надо учитывать.
  - «-» Размещение записей по этому методу дает в среднем одинаково плохое время поиска информации. Резкое увеличение времени доступа к информации по сравнению с другими методами.
  - «+» Полное использование памяти без регенерации, малые затраты на редактирование базы.
  - «+» Регенерация базы очень редкое явление.

### **Логические структуры.**

*Реквизитом* - называется единица информации, неделимая без потери физического смысла. Реквизиты бывают призначные и количественные. Призначные идентифицируют объект а количественные характеризуют. Структуры обычно объединяются в логическую структуру БД которая представляет не только совокупность структур но и организацию (связи между ними).



**Под базы** данных делятся по функциональным признакам и располагаются на томах.

**Том** - физическая единица информации, равная одному физическому носителю.

На томах располагаются файлы.

**Файлом** называется массив на внешнем носителе. Файлы бывают многотомные (располагаются на нескольких носителях). Чаще встречаются многофайловые тома (на одном томе располагается несколько файлов). Файл обычно состоит из записей.

**Запись** - единица информации, на уровне которой идет обмен между устройствами. Записи часто приходится объединять в блоки для того, чтобы обеспечить непрерывное последовательное размещение записей на физическом носителе.

Каждая запись состоит из полей.

**Поле** - минимальная поименованная единица информации. Поля могут объединяться в группы.

**Агрегат данных** - совокупность полей, объединенных общим именем.

**Сортировка** - упорядочивание записей в файле.

Различают системные файлы (используются в большом количестве задач). Системные файлы обычно хранятся во всех необходимых сортировках.

Кроме этого выделяют файлы:

- первичные,
- выходные,
- промежуточные.

**Первичные файлы** создаются на основе первичных документов.

**Промежуточные файлы** возникают в результате решения дополнительной задачи и требуются для решения другой задачи.

**Выходные файлы** - являются результатом решения комплекса задач и используются для управления объектом.

Записи в файлах имеют различную структуру:

- последовательная
- древовидная
- списковая
- цепная.

### Структуры баз данных.

Базы данных бывают:

- древовидные;
- сетевые;
- реляционные.

В **древовидных БД** каждый элемент вышестоящего уровня имеет ссылки на две или больше типов записей.

Логическая структура:



Запись цехов (так логическая структура реализуется в физическую):



Ссылка типа next хранит адрес следующей за ней единицы. Каждая запись имеет ссылку типа маска на предыдущую. Поиск идет по цепи.

«+» Очень быстрый поиск информации.

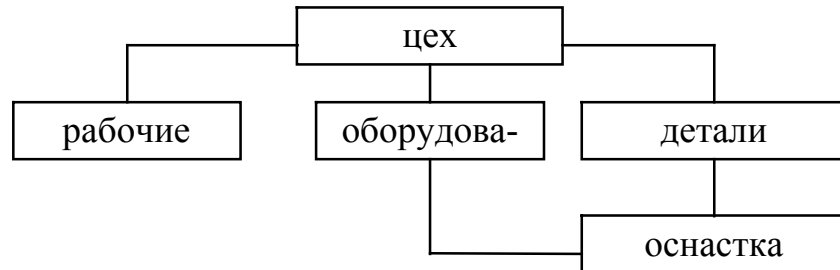
«+» Легкость восприятия логической структуры.

«-» Создание логической структуры требует высокой квалификации а физической чрезвычайно высокой. Написание алгоритмов и программ требует знания языка низкого уровня.

«-» Затраты на построение такой базы очень велики.

«-» Реализация любого типа запроса требует написания отдельной программы перемещения по спискам.

**Сетевая структура.** Допускаются связи между элементами одного уровня, либо два элемента одного уровня могут иметь элемент нижестоящего уровня.



«+» Есть возможность организовать больше точек входа и ускорить поиск т.е. уменьшить время реакции на запрос.

«-» Для каждого запроса имеется несколько алгоритмов поиска.

«-» В древовидной системе у нас был корень и ветки т.е. если мы дошли до конца и не нашли ответа на запрос значит такого элемента нет. У сети конца нет, по ней можно ходить вечно. Поиск окончания поиска должен быть иным. Он в каждом случае определяется по своему.

«-» Если пошли по цепи с целью найти Петрова и не нашли его то это не значит что его нет. Может надо зайти через другой вход. Цепи могут дублировать друг - друга и надо проводить сверку одноименных цепей.

«-» Сложность построения базы и требования к квалификации специалиста чрезвычайно высокие.

«+» Компактность БД. Максимум информации при минимуме избыточности информации.

«+» В этой БД минимум затрат на поиск информации. Процедура модификации очень проста.

**Реляционные БД.** Это такие БД, операции которых совпадают с операциями реляционной алгебры. Все файлы реляционной БД имеют четко выраженную табличную структуру. Столбцы называются **доменами**, строки - **кортежами**. Совокупность доменов и кортежей называют **отношением**.

*Основные свойства отношений:*

- 1) Отношение не имеет пустых кортежей (пустых строк нулевых доменов).
- 2) Отношение не имеет двух одинаковых кортежей (строк).

*Операции над реляционными БД:*

1. **Проецирование.** В результате этой операции из одного отношения можно получить другое отношение - проекцию в которое будут включены только часть домен.



Проекция по фамилии будут принадлежать только три кортежа.

Фамилия	И.О.	Принадлежность
Иванов	А.С.	КПСС
Петров	А.Н.	ВЛКСМ
Сидоров	П.Д.	ДР
Иванов	А.Д.	Память
Иванов	Л.К.	Память

2. **Пересечение.** Выполняется над двумя или более отношениями. В результате этой операции получа-

ется результирующее отношение, в которое включаются только те кортежи ключевые элементы которых совпадают во всех пересекаемых отношениях.

3. **Выборка** (поиск по ключу).

Доказано что с помощью этих трех операций можно выделить любой элемент и при этом получаются простые выражения.

«-» Квалификация требуется все же больше чем у не подготовленного пользователя.

Фамилия	И.О.	год рождения
Иванов	А.С.	1912
Петров	П.Б.	1972
Сидоров	Д.Л.	1939
Иванов	Л.К.	1956

«-» Оказалось что допустимы всевозможные типы запросов.

«+» Понятность структуры БД.

«+» Очень простой язык манипулирования данными.

«+» Возможность работы неподготовленного пользователя.

«+» Возможность реализации любого типа запроса.

«-» Неподготовленный пользователь все равно не работает с базой а значит ему и не надо понимать как это работает.

«-» Т.к. мы не знаем какие запросы будут то мы не можем оптимально разместить базу на диске.

«-» Чтобы легко найти запись мы должны реализовать поиск по ключу. Чтобы найти по ключу мы должны строить индексно-последовательные файлы а т.к. ключ часто не простой то мы должны создавать много индексно последовательных файлов следовательно все медленно.

Фамилия	Принадлежность	Год рождения
Иванов	КПСС	1912
Петров	ВЛКСМ	1972
Сидоров	ДР	1939
Иванов	Память	1956

Реляционные базы данных:

1. dBase - РЕБУС.
2. FoxBase - КАРАТ. Быстродействие в 6 раз больше по сравнению с dBase. В FoxBase нет "Ассистента" - программы для неопытного пользователя. UI - средство проектирования интерфейса пользователя. FoxBase = dBase + UI
3. Clipper. Более высокое быстродействие, позволял защищать программу, но эта СУБД для высококвалифицированных программистов. Достоинства реляционных БД: позволяют реализовать запрос любого типа, переделка такой базы менее болезненна.

### ***Банки данных.***

Банк данных - это большая сложная система, которая включает в себя:

- СУ БД,
- математическое обеспечение,
- программное обеспечение (совокупность программ),
- лингвистическое обеспечение (язык манипуляций заданиями, язык запросов, язык программирования, и другие),
- организационное обеспечение (совокупность инструкций по работе с БД),
- администрирование банка данных,
- юридическое обеспечение,
- методическое обеспечение,
- базы данных.

### ***Организационное обеспечение АСУ.***

#### ***Введение.***

Это совокупность приказов, распоряжений, штатных расписаний и инструкций. Организационное обеспечение определяет взаимодействие объекта управления и системы управления. Введение какой-либо модификации без организационного обеспечения невозможно в принципе. Часто такие решения принимаются на устном уровне, но это не допустимо. Необходимо добиться формального исполнения иначе никто не будет подчиняться. Противником обычно является объект управления. Нужно четко определить кому что должен.

#### ***Функции заказчика и подрядчика.***

Есть заказчик и пользователь. Часто эти два лица не совпадают.

1. Заказчик обеспечивает ***финансирование.***

2. Заказчик формирует **техническое задание** на проектирование (для разработки технического задания он может привлекать как других специалистов так и вас самих).
3. Заказчик должен обеспечить **своевременную поставку необходимой техники**.
4. Заказчик должен **своевременно согласовать всю техническую и организационную документацию** на предприятии пользователя.
5. Заказчик осуществляет **своевременную приемку и испытание автоматизированной системы** (нужно определить сроки).
6. Заказчик **обеспечивает изменения в штатных расписаниях, должностных инструкциях и организацию новых рабочих мест**.
7. Заказчик после проведения опытной эксплуатации **подписывает акт приемки работы**.
8. Далее заказчик обеспечивает **сохранение авторских прав**. Если организация финансирует, то она имеет право на авторство. Программа наша, если мы ее только продаем. Авторское право может делиться. Вопрос о передаче исходных текстов специализированных АРМ чаще всего ставится (тексты продаются) только один раз. Иногда делают так: сопровождают продукт в течении гарантийного срока, а лишь затем передают исходные тексты. При этом можно застраховаться (защитить продукт программно или аппаратным ключом или выбросить 2-3 листа текста).
9. Заказчик обеспечивает **соблюдение авторского надзора**. Обычно авторский надзор совпадает со сроком гарантии (в это время корректировки допустимы только при согласовании их с разработчиком). Авторский надзор выполняется обычно за отдельную плату.

### **Функции исполнителя (разработчика).**

1. Участвовать **в предпроектном исследовании**.
2. Он может **привлекаться для написания технического задания**.
3. **Разрабатывает техническую и рабочую документацию**.
4. **Разрабатывает программные и технические средства** и другие компоненты системы.
5. **Обучает персонал** пользователя.
6. Участвует в **приемо-сдаточных испытаниях**.
7. Проводит **опытную эксплуатацию**.
8. **Сдаст систему** в рабочую эксплуатацию.
9. Обеспечивает **авторский надзор и гарантийное и пост гарантийное обслуживание**.

### **Юридическое обеспечение АСУ.**

## *Введение.*

Юридическое обеспечение АСУ регламентируется положениями и нормами договорных отношений, практически в целом отсутствует, поэтому защита авторских прав является делом чрезвычайно трудоемким.

## *Оформление документов договорной деятельности при разработке.*

### *Основные понятия.*

Основным документом является *Договор*.

Обязательная часть договора - приложение, в котором должно быть:

1. Протокол соглашения о договорной цене.
2. График сдачи этапов работы (календарный план работы).
3. График и формы платежей.
4. Техническое задание, утвержденное заказчиком.

### *Состав договора.*

**1 Раздел:** Описание договаривающихся сторон и на основании чего они действуют.

**2 Раздел:** Предмет договора (разработка тех. задания, тех. проекта и т.д.).

**3 Раздел:** Обязанности сторон.

Заказчик должен взять на себя часть работ - он участвует в разработке технического задания (если оно еще не разработано). В этом разделе должно быть указано, кто обязан создавать информационную базу данных. В пункте о соблюдении авторских прав необходимо оговорить форму договора (либо продать полностью авторские права на программный продукт; либо сохранить их, а продать единую копию без исходных текстов программы, без права на продажу без согласия автора, без права на тиражирование, копирование и т.д.) - этот пункт является наиболее спорным в договоре, поэтому его следует тщательно продумать. Оговариваются обязанности разработчика на сопровождение программного продукта на время гарантийного срока.

**4 Раздел:** Особые условия.

Записываются индивидуальные особенности, которые отличают договор от типового. Описывается форма оплаты (сумма договора индексируется поквартально в зависимости от инфляции). Оговаривается механизм определения роста инфляции: обычно выбирают региональную или центральную биржу и отслеживают курс доллара на момент заключения договора и на момент сдачи продукта.



**5 Раздел:** Срок действия договора.

Необходимо отметить до какого срока независимо от даты расторжения договора :

1. Сохраняются авторские права.
2. Выполняется сопровождение программного продукта.

**6 Раздел:** Порядок разрешения конфликтных ситуаций.

Закрепляется процедура разрешения конфликтов и обязанности сторон в процессе решения спорных вопросов. Расписывается порядок расчета стороны, виновной в возникновении конфликтной ситуации, с пострадавшей стороной.

**7 Раздел:** Юридические адреса и платежные реквизиты сторон.

***Содержание технического задания.***

1. ***Основания для разработки.*** Ссылка на договор в соответствии с которым выполняется заказ либо приказ (распоряжение) предприятия.
2. ***Цель и назначение*** разработки.
3. ***Описание существующей системы*** решения задачи и ее место в общем комплексе решаемых задач.
4. ***Анализ достоинств***, недостатков существующей системы.
5. ***Формирование требований*** на основе достоинств и недостатков к разрабатываемой системе.
6. ***Технико-экономические требования.***

# ***Элементы проектирования.***

## ***Стадии проектирования.***

### ***1. Предпроектная стадия*** - обследование объекта.

Выходные документы:

- Отчет о предпроектном обследовании объекта.
- Техничко-экономическое обоснование разработки.
- Техническое задание.

На этой стадии выполняется предварительное построение системы (эскизный проект). На заводе это НИР и ОКР.

### ***2. Стадия проектирования.***

Пишется технический или технорабочий проект, далее пишется рабочий проект. После этого выполняется полный объем программирования, подготовка к проведению испытаний.

Выходными документами испытаний являются:

- Акт приема сдаточных испытаний.
- Протокол замеченных недоработок (в нем указываются сроки устранения замечаний).

### ***3. Стадия опытной эксплуатации.***

Устанавливаются сроки, которые позволяют выявить скрытые недостатки (полгода - год).

### ***4. Рабочая эксплуатация.***

Гарантийное и послегарантийное обслуживание системы.

## ***Содержание технического проекта.***

Техническое задание, предпроектное обслуживание, технический проект и технико-экономическое обоснование выполняются системотехником.

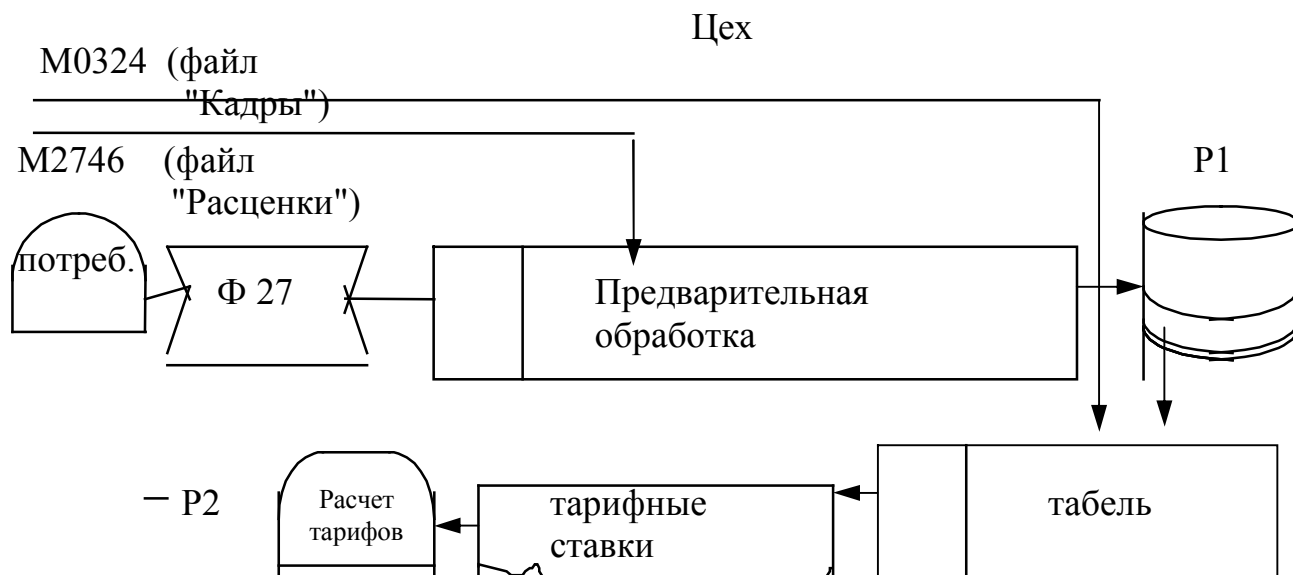
1. Основание для разработки.
2. Цель и назначение.
3. Место решаемой задачи в общем комплексе задач.
4. Требования к разрабатываемой системе.
5. Обзор существующих методов решения.
6. Выбор метода решения.
7. Формирование компонент организационного, юридического, лингвистического обеспечения.
8. Контрольный пример для проверки правильности программы.
9. Приложения:
  - принципиальная блок-схема;
  - формы документов;
  - схемы документооборотов.

### *Содержание рабочего проекта.*

1. Основание для разработки.
2. Цель и назначение разрабатываемой системы.
3. Технологическая система расчетов. (Закрепляется последовательность выполнения технологических операций на вычислительной технике).
4. Описание программ.
5. Анализ результата расчета на контрольном примере.
6. Инструкции пользователю, системному программисту, прикладному программисту.
7. Описание технологического процесса с технологическими картами. Особенности эксплуатации задач.


### *Правила рисования информационных систем.*


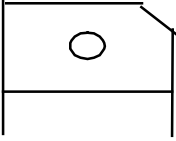

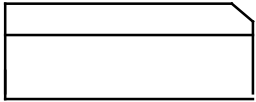
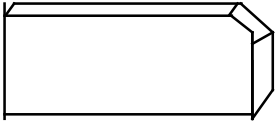


Сверху записываются информационные файлы. Слева пишутся источники информации. В средней части рисуется схема расчетов. С правой стороны рисуются выходные документы.





### *Условные обозначения применяемые в проектировании.*

#### *Носители.*

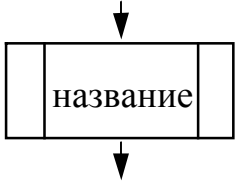
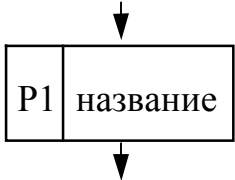
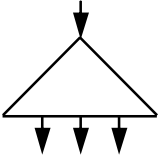
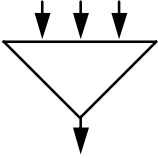
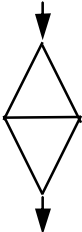
-  файл на магнитной ленте

-  файл на жестком диске
-  файл на перфоленте
-  файл на гибком диске
-  перфокарты
-  колода перфокарт
-  файл на перфокартах
-  передача сообщений



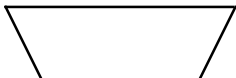
***Устройства ввода-вывода.***

-  вывод на дисплей
-  ввод с клавиатуры
-  вывод на бумажный носитель

**Операции.**

-  стандартная подпрограмма  
( процедура или функция )
-  расчет ( преобразование информации )
-  разбиение информации
-  слияние информации
-  процедура сортировки информации

**Другие.**

-  подразделение потребитель-поставщик информации
-  ручной документ
-  ручная обработка

-  переход на метку

-  межстраничный соединитель

- <имя>\_\_\_\_\_ системные файлы (файлы которые используются более чем в двух задачах называют системными)

## *Экзаменационные вопросы.*

1. Основные понятия ИСАУ. Предпосылки создания АСУ.
2. Структуры систем управления.
3. Понятие автоматизированных систем, их классификация. Свойства большой системы.
4. Основы современного производства. Стадии изготовления. Календарно-плановые нормативы.
5. Существующие системы адаптивного управления.
6. Принципы построения АСУ.
7. Подсистема сбора и регистрации информации.
8. Подсистема приема и передачи информации. Подсистема преобразования информации.
9. Подсистема хранения и обработки информации.
10. Характеристики процессоров.
11. Характеристики устройств отображения. Мониторы.
12. Подсистема отображения. Принтеры.
13. Подсистема отображения. Плоттеры.
14. Подсистема ввода информации.
15. Подсистема хранения информации. Характеристики.
16. Понятие модели. Классификация.
17. Основные принципы моделирования. Принцип  $\Delta t$ . Пример.
18. Основные принципы моделирования. Принцип последовательной проводки заявок. Пример.
19. Основные принципы моделирования. Принцип особых состояний. Пример.
20. Этапы моделирования. Методы оптимизации.
21. Программное обеспечение. Структура.
22. Характеристики ОС MS-DOS.
23. Характеристики ОС UNIX.
24. Сравнительные характеристики других операционных систем.
25. Программное обеспечение. Пользовательские программы.
26. Информационное обеспечение. Задачи внешнего информационного обеспечения.
27. Информационное обеспечение. Задачи внутреннего информационного обеспечения.
28. Методы физического размещения информации.
29. Логические структуры данных.
30. Структуры баз данных.
31. Понятие банков и баз данных. Требования к базам данных.
32. Юридическое обеспечение.
33. Стадии проектирования. Состав документов.