

Лабораторная работа № 3.

Методология функционального моделирования IDEF0

Цель работы: изучить теоретические основы структурного подхода к проектированию информационных систем, познакомиться с основными понятиями функционального моделирования и графической нотацией IDEF0.

Теоретические сведения

IDEF0 используется для создания функциональной модели, отображающей структуру и функции системы, а также потоки информации и материальных объектов, связывающие эти функции.

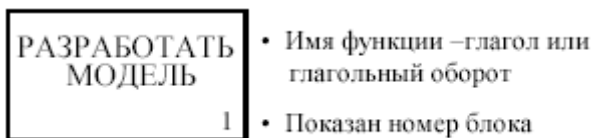
Синтаксис IDEF0

Набор структурных компонентов языка, их характеристики и правила, определяющие связи между компонентами, представляют собой синтаксис языка. Компоненты синтаксиса IDEF0 – блоки, интерфейсные дуги, диаграммы и правила. Блоки представляют функции, определяемые как деятельность, процесс, операция, действие или преобразование. Интерфейсные дуги представляют данные или материальные объекты, связанные с функциями. Правила определяют, как следует применять компоненты; диаграммы обеспечивают формат графического и словесного описания моделей. Формат образует основу для управления конфигурацией модели.

Блок

Блок описывает функцию. Внутри каждого блока помещается его имя и номер. Имя должно быть активным глаголом или глагольным оборотом, описывающим функцию. Номер блока размещается в правом нижнем углу. Номера блоков использу-

ются для их идентификации на диаграмме и в соответствующем тексте.

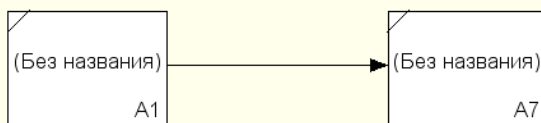


Синтаксические правила использования блоков:

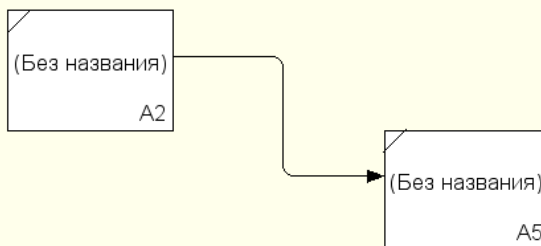
- размеры блоков должны быть достаточными для того, чтобы включить имя блока;
- блоки должны быть прямоугольными, с прямыми углами;
- блоки должны быть нарисованы сплошными линиями.

Интерфейсная дуга

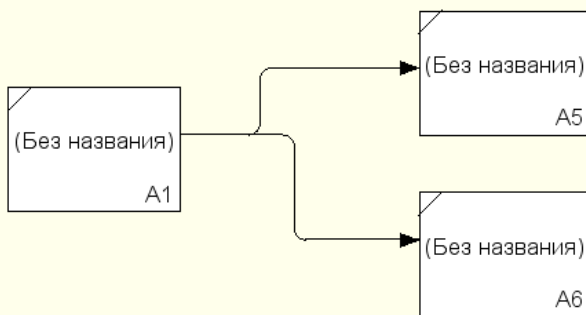
Интерфейсная дуга формируется из одного или более отрезков прямых и наконечника на одном конце. Сегменты интерфейсных дуг могут быть прямыми или ломаными; в последнем случае горизонтальные и вертикальные отрезки сопрягаются дугами, имеющими угол 90° .



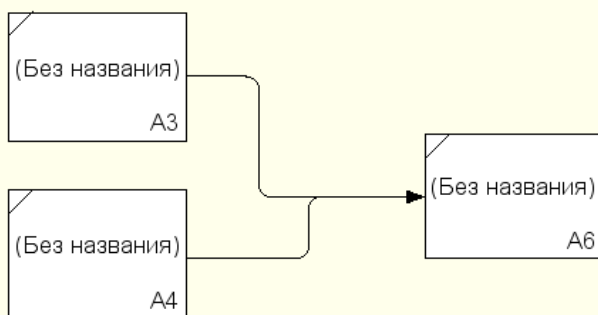
Прямолинейный отрезок
интерфейсной дуги



Ломаный отрезок
интерфейсной дуги



Ветвление
интерфейсной дуги



Слияние
интерфейсной дуги

Интерфейсные дуги не представляют поток или последовательность событий, как в традиционных блок-схемах потоков или процессов. Они лишь показывают, какие данные или материальные объекты должны поступить на вход функции для того, чтобы эта функция могла выполняться.

Синтаксические правила использования интерфейсных дуг:

- ломаные интерфейсные дуги изменяют направление только под углом 90 град;
- интерфейсные дуги должны быть нарисованы сплошными линиями различной толщины;

- интерфейсные дуги могут состоять только из вертикальных или горизонтальных отрезков; отрезки, направленные по диагонали, не допускаются;
- концы интерфейсных дуг должны касаться внешней границы функционального блока, но не должны пересекать ее;
- интерфейсные дуги должны присоединяться к блоку на его сторонах (присоединение в углах не допускается).

Семантика языка IDEF0

Семантика определяет содержание (значение) синтаксических компонентов языка и способствует правильности их интерпретации. Интерпретация устанавливает соответствие между блоками и интерфейсными дугами с одной стороны и функциями и их интерфейсами – с другой.

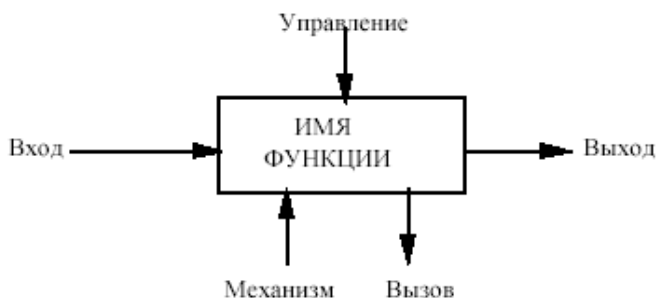
Семантика блоков и интерфейсных дуг

Поскольку IDEF0 есть методология функционального моделирования, имя блока, описывающее функцию, должно быть глаголом или глагольным оборотом; например, имя блока "Выполнить проверку", означает, что блок с таким именем превращает непроверенные детали в проверенные. После присваивания блоку имени, к соответствующим его сторонам присоединяются входные, выходные и управляющие интерфейсные дуги, а также интерфейсные дуги механизма, что и определяет наглядность и выразительность изображения блока IDEF0.

Чтобы гарантировать точность модели, следует использовать стандартную терминологию. Блоки именуются глаголами или глагольными оборотами и эти имена сохраняются при декомпозиции Интерфейсные дуги и их сегменты, как отдельные, так и связанные в «пучок», помечаются существительными или оборотами существительного. Метки сегментов позволяют конкретизировать данные или материальные объекты, передаваемые

мые этими сегментами, с соблюдением синтаксиса ветвлений и слияний.

Каждая сторона функционального блока имеет стандартное значение с точки зрения связи блок/интерфейсные дуги, В свою очередь, сторона блока, к которой присоединена интерфейсная дуга, однозначно определяет ее роль. Интерфейсные дуги, входящие в левую сторону блока - входы. Входы преобразуются или расходуются функцией, чтобы создать то, что появится на ее выходе. Интерфейсные дуги, входящие в блок сверху - управления. Управления определяют условия, необходимые функции, чтобы произвести правильный выход. Интерфейсные дуги, покидающие блок справа – выходы, т.е. данные или материальные объекты, произведенные функцией.



Интерфейсные дуги, подключенные к нижней стороне блока, представляют механизмы. Интерфейсные дуги, направленные вверх, идентифицируют средства, поддерживающие выполнение функции. Другие средства могут наследоваться из родительского блока. Интерфейсные дуги механизма, направленные вниз, являются интерфейсными дугами вызова. Интерфейсные дуги вызова обозначают обращение из данной модели или из данной части модели к блоку, входящему в состав другой модели или другой части модели, обеспечивая их связь, т.е. раз-

ные модели или разные части одной и той же модели могут совместно использовать один и тот же элемент (блок).

Имена и метки

Как указывалось, имена функций – глаголы или глагольные обороты. Примеры таких имен:

производить детали	планировать ресурсы	наблюдать
наблюдать за выполнением	проектировать систему	эксплуатировать
разработать детальные чертежи	изготовить компонент	проверять деталь

Интерфейсные дуги идентифицируют данные или материальные объекты, необходимые для выполнения функции или производимые ею. Каждая интерфейсная дуга должна быть помечена существительным или оборотом существительного, например:

Спецификации	отчет об испытаниях	бюджет
Конструкторские требования	конструкция детали	директива
Инженер-конструктор	плата в сборе	требования

Семантические правила блоков и интерфейсных дуг:

- имя блока должно быть активным глаголом или глагольным оборотом;
- входные интерфейсные дуги должны связываться с левой стороной блока;
- управляющие интерфейсные дуги должны связываться с верхней стороной блока;
- выходные интерфейсные дуги должны связываться с правой стороной блока;
- интерфейсные дуги механизма (кроме интерфейсных дуг вызова) должны указывать вверх и подключаться к нижней стороне блока;
- интерфейсные дуги вызова механизма должны указывать вниз, подключаться к нижней стороне блока, и помечаться ссылкой на вызываемый блок;



- сегменты интерфейсных дуг, за исключением интерфейсных дуг вызова, должны помечаться существительным или оборотом существительного, если только единственная метка интерфейсной дуги не относится к ней в целом;

- чтобы связать интерфейсную дугу с меткой, следует использовать «тильду» (~);

- в метках интерфейсных дуг не должны использоваться следующие термины: функция, вход, управление, выход, механизм, вызов.

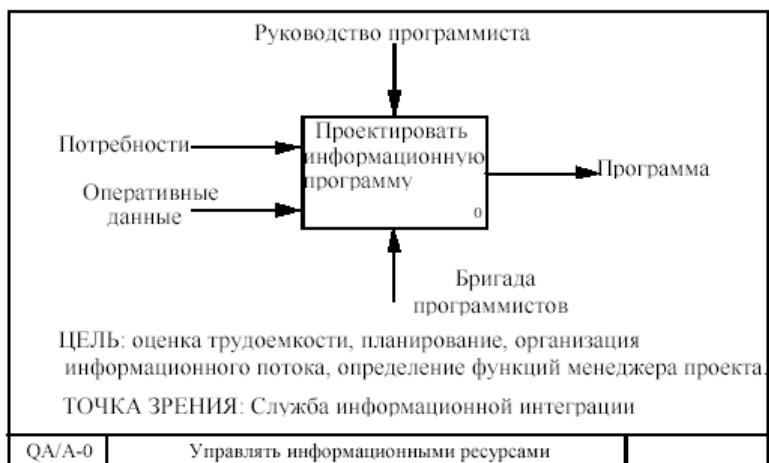
Диаграммы IDEF0

IDEF0-модели состоят из трех типов документов: графических диаграмм, текста и глоссария. Эти документы имеют перекрестные ссылки друг на друга. Графическая диаграмма – главный компонент IDEF0-модели, содержащий блоки, интерфейсные дуги, соединения блоков и интерфейсных дуг и ассоциированные с ними отношения. Блоки представляют основные функции моделируемого объекта. Эти функции могут быть разбиты (декомпозированы) на составные части и представлены в виде более подробных диаграмм; процесс декомпозиции про-

должается до тех пор, пока объект не будет описан на уровне детализации, необходимом для достижения целей конкретного проекта. Диаграмма верхнего уровня обеспечивает наиболее общее или абстрактное описание объекта моделирования. За этой диаграммой следует серия дочерних диаграмм, дающих более детальное представление об объекте.

Контекстная диаграмма верхнего уровня

Каждая модель должна иметь контекстную диаграмму верхнего уровня, на которой объект моделирования представлен единственным блоком с граничными интерфейсными дугами. Эта диаграмма называется А-0 (А минус нуль). Интерфейсные дуги на этой диаграмме отображают связи объекта моделирования с окружающей средой. Поскольку единственный блок представляет весь объект, его имя общее для всего проекта. Это же справедливо и для всех интерфейсных дуг диаграммы, поскольку они представляют полный комплект внешних интерфейсов объекта. Диаграмма А-0 устанавливает область моделирования и ее границу.



Контекстная диаграмма A-0 также должна содержать краткие утверждения, определяющие точку зрения должностного лица или подразделения, с позиций которого создается модель, и цель, для достижения которой ее разрабатывают. Эти утверждения помогают руководить разработкой модели и ввести этот процесс в определенные рамки. Точка зрения определяет, что и в каком разрезе можно увидеть в пределах контекста модели. Изменение точки зрения, приводит к рассмотрению других аспектов объекта. Аспекты, важные с одной точки зрения, могут не появиться в модели, разрабатываемой с другой точки зрения на тот же самый объект.

Формулировка цели выражает причину создания модели, т.е. содержит перечень вопросов, на которые должна отвечать модель, что в значительной мере определяет ее структуру. Наиболее важные свойства объекта обычно выявляются на верхних уровнях иерархии; по мере декомпозиции функции верхнего уровня и разбиения ее на подфункции, эти свойства уточняются. Каждая подфункция, в свою очередь, декомпозируется на элементы следующего уровня, и так происходит до тех пор, пока не будет получена релевантная структура, позволяющая ответить на вопросы, сформулированные в цели моделирования. Каждая подфункция моделируется отдельным блоком. Каждый родительский блок подробно описывается дочерней диаграммой на более низком уровне. Все дочерние диаграммы должны быть в пределах области контекстной диаграммы верхнего уровня.

Дочерняя диаграмма

Единственная функция, представленная на контекстной диаграмме верхнего уровня, может быть разложена на основные подфункции посредством создания дочерней диаграммы. В

свою очередь, каждая из этих подфункций может быть разложена на составные части посредством создания дочерней диаграммы следующего, более низкого уровня, на которой некоторые или все функции также могут быть разложены на составные части. Каждая дочерняя диаграмма содержит дочерние блоки и интерфейсные дуги, обеспечивающие дополнительную детализацию родительского блока.

Дочерняя диаграмма, создаваемая при декомпозиции, охватывает ту же область, что и родительский блок, но описывает ее более подробно. Таким образом, дочерняя диаграмма как бы вложена в свой родительский блок.

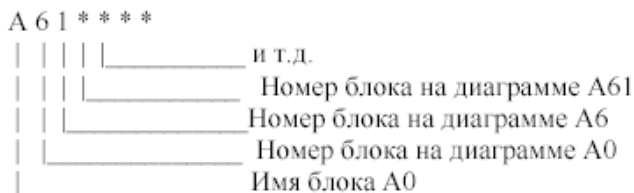
Родительская диаграмма

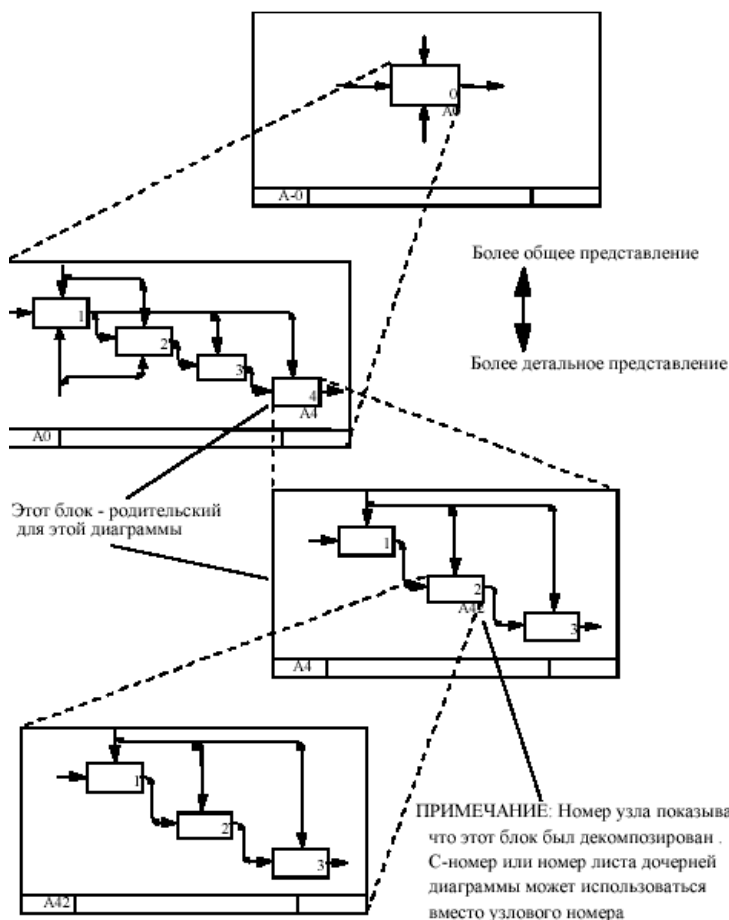
Родительская диаграмма – та, которая содержит один или более родительских блоков. Каждая обычная (не-контекстная) диаграмма является также дочерней диаграммой, поскольку, по определению, она подробно описывает некоторый родительский блок. Таким образом, любая диаграмма может быть как родительской диаграммой (содержать родительские блоки), так и дочерней (подробно описывать собственный родительский блок). Аналогично, блок может быть как родительским (подробно описываться дочерней диаграммой) так и дочерним (появляющимся на дочерней диаграмме). Основное иерархическое отношение существует между родительским блоком и дочерней диаграммой, которая его подробно описывает.

То, что блок является дочерним и раскрывает содержание родительского блока на диаграмме предшествующего уровня, указывается специальным ссылочным кодом, написанным ниже правого нижнего угла блока. Этот ссылочный код может формироваться несколькими способами, из которых самый простой заключается в том, что код, начинающийся с буквы А(по имени

диаграммы A-0), содержит цифры, определяемые номерами родительских блоков. Например, показанные на рис.7 коды означают, что диаграмма является декомпозицией 1-го блока диаграммы, которая, в свою очередь является декомпозицией 6-го блока диаграммы A0, а сами коды образуются присоединением номера блока.

Таким образом, код формируется так:





Создание контекстной диаграммы на основе IDEF0

1. В составе модели должна присутствовать контекстная диаграмма A-0, которая содержит только один блок. Номер единственного блока на контекстной диаграмме A-0 должен быть 0.

2. Блоки на диаграмме должны располагаться по диагонали – от левого верхнего угла диаграммы до правого нижнего в порядке присвоенных номеров. Блоки на диаграмме, расположенные вверху слева «доминируют» над блоками, расположенными

внизу справа. «Доминирование» понимается как влияние, которое блок оказывает на другие блоки диаграммы. Расположение блоков на листе диаграммы отражает авторское понимание доминирования.

Таким образом, топология диаграммы показывает, какие функции оказывают большее влияние на остальные.

3. Неконтекстные диаграммы должны содержать не менее трех и не более шести блоков. Эти ограничения поддерживают сложность диаграмм на уровне, доступном для чтения, понимания и использования.

Диаграммы с количеством блоков менее трех вызывают серьезные сомнения в необходимости декомпозиции родительской функции. Диаграммы с количеством блоков более шести сложны для восприятия читателями и вызывают у автора трудности при внесении в нее всех необходимых графических объектов и меток.

4. Каждый блок неконтекстной диаграммы получает номер, помещаемый в правом нижнем углу; порядок нумерации - от верхнего левого к нижнему правому блоку (номера от 1 до 6).

5. Каждый блок, подвергнутый декомпозиции, должен иметь ссылку на дочернюю диаграмму; ссылка (например, узловый номер, С-номер или номер страницы) помещается под правым нижним углом блока.

6. Имена блоков (выполняемых функций) и метки интерфейсных дуг должны быть уникальными. Если метки интерфейсных дуг совпадают, это значит, что интерфейсные дуги отображают тождественные данные.

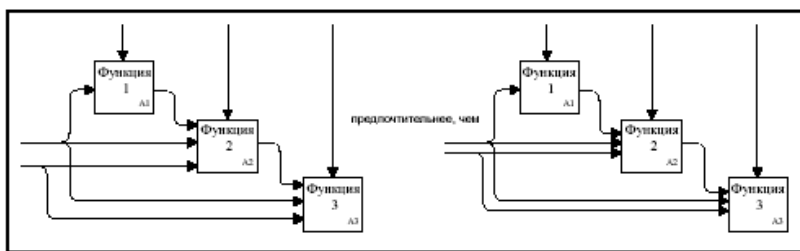
7. При наличии интерфейсных дуг со сложной топологией целесообразно повторить метку для удобства ее идентификации.

8. Следует обеспечить максимальное расстояние между блоками и поворотами интерфейсных дуг, а также между блоками и пересечениями интерфейсных дуг для облегчения чтения диаграммы. Одновременно уменьшается вероятность перепутать две разные интерфейсные дуги.

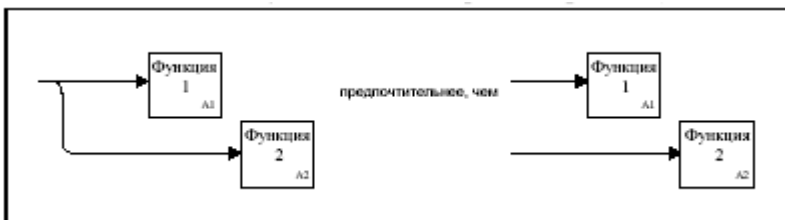
9. Блоки всегда должны иметь хотя бы одну управляющую, одну выходную интерфейсную дугу и одну интерфейсную дугу механизма, но могут не иметь входных интерфейсных дуг.

10. Если одни и те же данные служат и для управления, и для входа, вычерчивается только интерфейсная дуга управления. Этим подчеркивается управляющий характер данных и уменьшается сложность диаграммы.

11. Максимально увеличенное расстояние между параллельными интерфейсными дугами облегчает размещения меток, их чтение и позволяет проследить пути интерфейсных дуг.



12. Интерфейсные дуги связываются (сливаются), если они представляют сходные данные и их источник не указан на диаграмме.



13. Обратные связи по управлению должны быть показаны как "вверх и над" (рис. 3.1, а):

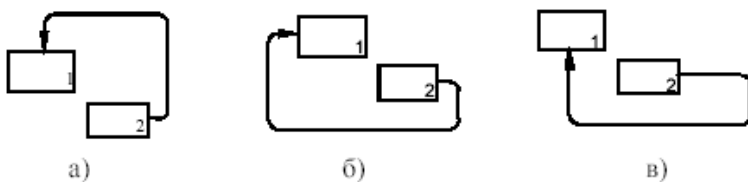
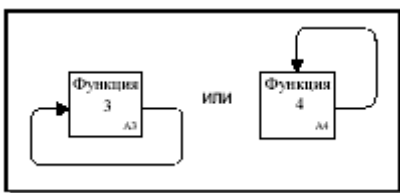


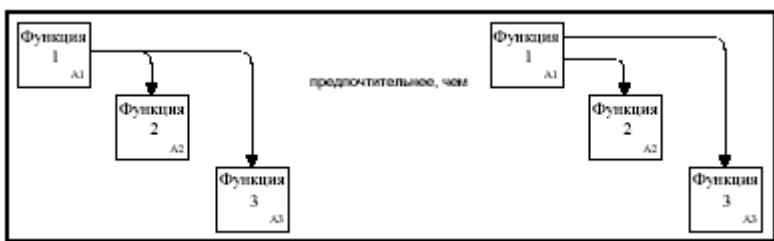
Рис. 3.1

Обратные связи по входу должны быть показаны как "вниз и под" (рис. 4.3,б). Так же показываются обратные связи посредством механизма. Таким образом, обеспечивается показ обратной связи при минимальном числе линий и пересечений.

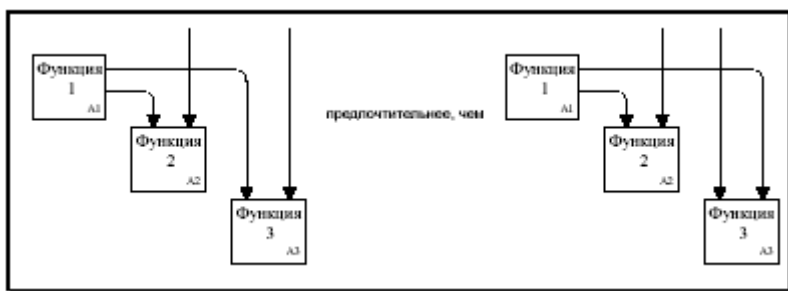
14. Циклические обратные связи для одного и того же блока изображаются только для того, чтобы их выделить. Обычно обратную связь изображают на диаграмме, декомпозирующей блок. Однако иногда требуется выделить повторно используемые объекты.



15. Интерфейсные дуги объединяются, если они имеют общий источник или приемник, или они представляют связанные данные. Общее название лучше описывает суть данных. Следует минимизировать число интерфейсных дуг, касающихся каждой стороны блока, если, конечно, природа данных не слишком разнородна.



16. Если возможно, интерфейсные дуги присоединяются к блокам в одной и той же позиции. Тогда соединение интерфейсных дуг конкретного типа с блоками будет согласованным и чтение диаграммы упростится.



17. При соединении большого числа блоков необходимо избегать необязательных пересечений интерфейсных дуг. Следует минимизировать число петель и поворотов каждой интерфейсной дуги.

Содержание лабораторной работы

В лабораторной работе требуется разработать диаграмму IDEF0 в соответствии с Вашим вариантом. Отчет о проделанной работе (пример оформления отчета по лабораторной работе смотри приложение 1) должен содержать: название и цель работы; номер варианта и описание задания; диаграмму соответствующую варианту; необходимые комментарии к диаграмме и выводы по проделанной лабораторной работе.

Варианты заданий

Для выполнения работы варианты задания берутся из лабораторной работы №1.