

3533847.00026-02 33 01

Подпрограмма YP333 предназначена для описания поверхности вращения по грани и оси путем вращения данной грани вокруг данной оси на данный угол.

Обращение:

```
CALL YP333(KI,KJ,X0,Y0,Z0,X1,Y1,Z1,AL,L)
```

где KI - имя создаваемого или дополняемого набора граней;

KJ - имя начально данного объекта (грани);

X0, Y0, Z0 - начальная точка оси вращения;

X1, Y1, Z1 - конечная точка оси вращения;

AL - ориентированный угол поворота (в градусах);

L - число равных частей, на которое разбивается угол поворота при аппроксимации набором граней.

Предполагается, что ось вращения не перпендикулярна плоскости грани. Если внешняя нормаль к грани противоположна направлению вращения, то внешние нормали к создаваемым граням будут направлены вовне "трубы".

Результат:

KIEL(10)=0: операция выполнена;

KIEL(10)!=0: операция не выполнена.

Подпрограмма YP334 предназначена для описания замкнутой призмы, полученной переносом данной грани на данный вектор.

Обращение:

```
CALL YP334(KI,KJ,X1,Y1,Z1)
```

где KI - имя создаваемого или дополняемого набора граней;

KJ - имя данного объекта (грани);

X1, Y1, Z1 - координаты вектора переноса.

Результат:

KIEL(10)=0: операция выполнена;

KIEL(10)!=0: операция не выполнена.

Подпрограмма YP335 предназначена для описания замкнутой пирамиды по данной точке вращения (вершина пирамиды) и данной грани (основание пирамиды).

Обращение:

```
CALL YP335(KI,KJ,X1,Y1,Z1)
```

где KI - имя создаваемого или дополняемого набора граней;

KJ - имя данного объекта (грани);

X1, Y1, Z1 - данная точка (вершина пирамиды).

Предполагается, что точка (X1, Y1, Z1) не лежит в плоскости грани основания.

Результат:

KIEL(10)=0: операция выполнена;

KIEL(10)!=0: операция не выполнена.

Подпрограмма YP336 предназначена для описания многогранного тела вращения по заданной грани и оси вращения путем вращения данной грани вокруг данной оси на данный угол.

Обращение:

```
CALL YP336(KI,KJ,X1,Y1,Z1,X2,Y2,Z2,AL,L)
```

3533847.00026-02 33 01

где KI - имя создаваемого или дополняемого набора граней;  
 KJ - имя данного объекта (грани);  
 X1, Y1, Z1 )  
 > - координаты двух точек, определяющие ось  
 X2, Y2, Z2 ) вращения;  
 AL - величина угла вращения, задаваемая в градусах;  
 L - число равных частей, на которые разбивается угол вращения  
 при аппроксимации тела многогранником.  
 При  $AL > 0$  вращение осуществляется против часовой стрелки, если  
 смотреть из точки (X2, Y2, Z2) в точку (X1, Y1, Z1).  
 В противном случае это вращение осуществляется в противопо-  
 ложном направлении. Предполагается, что ось вращения не перпенди-  
 кулярна плоскости исходной грани.

Результат:

KIEL(10)=0: операция выполнена;

KIEL(10)!=0: операция не выполнена.

Подпрограмма YP337 предназначена для описания многогранника  
параллелепипеда.

Обращение:

CALL YP337(KI, X, Y, Z)

где KI - имя создаваемого или дополняемого набора граней;  
 X, Y, Z - координаты самой удаленной вершины параллелепипеда от  
 вершины (0,0,0).

При обращении к этой подпрограмме создается параллелепипед с  
 ребрами, параллельными осям координат, главная диагональ которого  
 определяется точками (0,0,0), (X, Y, Z).

Результат:

KIEL(10)=0: операция выполнена;

KIEL(10)!=0: операция не выполнена.

Подпрограмма YP338 предназначена для описания замкнутого  
двойного кругового цилиндра.

Обращение:

CALL YP338(KI, R1, R2, H, N)

где KI - имя создаваемого или дополняемого набора граней;  
 R1 - радиус внутренней поверхности кругового цилиндра;

R2 - радиус внешней поверхности кругового цилиндра;

H - высота кругового цилиндра;

N - число частей, на которые разбиваются окружности основания  
 при ее аппроксимации вписанным многоугольником.

Подпрограмма строит двойной цилиндр с осью, расположенной на  
 оси OZ, и основаниями, расположенными в плоскостях Z=0 и Z=H. Если  
 R1=0, то будет построен обычный правильный цилиндр. Предполагается  
 $R2 > R1 > 0$ .

Результат:

KIEL(10)=0: операция выполнена;

KIEL(10)!=0: операция не выполнена.

Подпрограмма YP339 предназначена для описания замкнутого  
двойного кругового конуса.

Обращение:

CALL YP339(KI, R1, R2, H, N)

3533847.00026-02 33 01

где KI - имя создаваемого или дополняемого набора граней;  
 R1 - радиус внутренней поверхности конуса;  
 R2 - радиус внешней поверхности конуса;  
 H - высота конуса; его вершина находится в точке (0,0,H);  
 N - число частей, на которое разбивается окружность основания при ее аппроксимации вписанным многоугольником.

Подпрограмма строит двойной конус с осью и вершиной, расположенных на оси OZ, и основанием, расположенным в плоскости XOY. Если R1=0, то будет построен правильный конус. Предполагается, что R2>R1>=0.

Результат:

KIEL(10)=0: операция выполнена;  
 KIEL(10)!=0: операция не выполнена.

#### 2.14.5. Подпрограммы вычисления характеристик

В этом пункте описываются подпрограммы, предназначенные для вычисления характеристик сформированных объектов.

Подпрограмма YP260 предназначена для вычисления периметра, площади и центра тяжести однородного многоугольника.

Обращение:

CALL YP260(KI,P,S,X,Y)

где KI - имя многоугольника;  
 P - переменная, в которую заносится значение периметра;  
 S - переменная, в которую заносится значение площади;  
 X,Y - переменные, в которые помещаются координаты центра тяжести однородного многоугольника.

Результат:

KIEL(10)=0: операция выполнена;  
 KIEL(10)!=0: операция не выполнена.

Подпрограмма YP361 предназначена для расчета характеристик набора граней как многогранной поверхности.

Обращение:

CALL YP361(KI,S,X,Y,Z,H)

где KI - имя набора граней;  
 S - площадь многогранной поверхности;  
 X,Y,Z - координаты центра многогранной поверхности;  
 H - момент инерции многогранной поверхности.

Результат:

KIEL(10)=0: операция выполнена;  
 KIEL(10)!=0: операция не выполнена.

Подпрограмма YP362 предназначена для расчета характеристик набора граней как многогранника (как тела).

Обращение:

CALL YP362(KI,V,X,Y,Z,H)

где KI - имя набора граней;  
 V - объем многогранника;  
 X,Y,Z - координаты его центра инерции;  
 H - момент инерции многогранника (как тела).

3533847.00026-02 33 01

**Результат:**

KIEL(10)=0: операция выполнена;

KIEL(10)!=0: операция не выполнена.

Подпрограмма YP901 предназначена для вычисления характеристик объекта, многоугольника или многогранника и печать их на АЦПУ с пояснениями.

**Обращение:**

CALL YP901(KI)

где KI - имя объекта типа 210 - многоугольника или 220 - многогранника.

Для многоугольника подпрограмма вычисляет и печатает на АЦПУ: периметр, площадь, центр инерции. Для многогранника: площадь поверхности, момент инерции поверхности относительно оси Z, объем, момент инерции тела относительно оси Z, центр инерции.

**2.15. Представление объектов с использованием Т-структур**

В компоненте АМТТ описываются программы построения и обработка трехмерных объектов и их 2Д-моделей с использованием иерархического представления. Представление объектов основано на Т-структурах.

Т-структура представляет собой куст объектов, архивные типы которых больше или равны 1000. Причем, объекты с типом 1000 являются порциями. Для рассматриваемых представлений характерно следующее: вся содержательная информация заключена в порциях, которые состоят из элементов. Объекты Т-структуры, не являющиеся порцией, содержат только атрибутную информацию.

**2.15.1. Представление трехмерных объектов**

Структура представления трехмерного объекта в комплекте АМТТ изображена на рис.21.

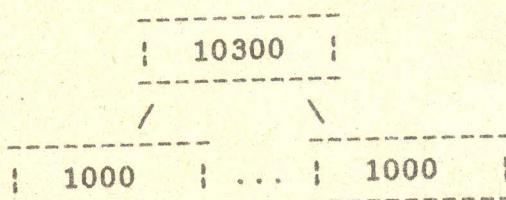
**Архивная структура трехмерного объекта**

Рис.21

В прямоугольниках (см.рис.21) даны условные обозначения Т-объектов, приведены значения их типов. Головной объект не имеет информационной части. В информационных частях объектов - порций (тип=1000) содержатся элементы трехмерного объекта. Изображенная структура выстраивается программами создания трехмерных объектов. Размер информационной части порции указывается при создании объекта. Элементом трехмерного объекта является одна грань, следо-

3533847.00026-02 33 01

вательно, любая грань должна помещаться в кванте (порции) и в оперативной памяти. Структура информационной части порции описана в ПС ИСД. Структура элемента - грань изображена на рис.22, где ДЕ - значение первого слова элемента - общее число слов им занимаемое; КА - значение 2-го слова элемента - число слов атрибута (возможно равное 0). Далее идут КА слов атрибута. Затем размещается обычное описание грани через ребра, по 6 слов на ребро. Вообще говоря, в конце поля элемента могут оставаться незадействованные слова.

#### Структура элемента - грань

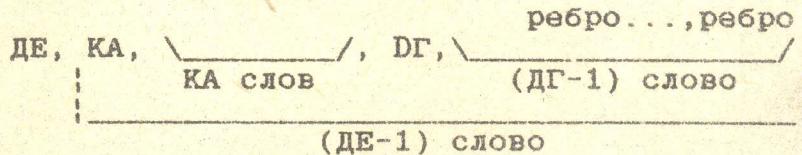


Рис.22

Описываемые ниже подпрограммы их никак не обрабатывают. Подпрограммы создания объекта приписывают каждой грани одно слово атрибута ( $KA=1$ ) и первоначально устанавливают его в соответствии с их параметрами или делают равным 0, если соответствующих параметров нет.

Следует отметить, что при создании объекта порядок элементов и, следовательно, граней может быть произвольный. Для обрабатываемых алгоритмов это несущественно.

#### 2.15.2. Представление плоских образов

Под плоскими образами понимаются наборы отрезков и плоских контуров. Эти образы формируются автоматически в программах удаления невидимых линий. На рис.23 изображена архивная структура плоского образа.

#### Архивная структура плоского образа

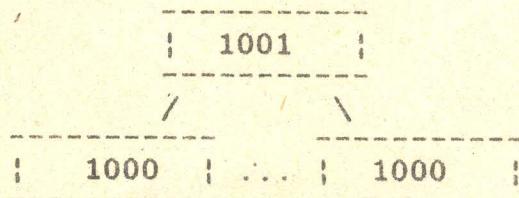


Рис.23

В прямоугольниках (см.рис.23) даны условное обозначение т-объектов, приведены значения их типов. Головной объект не имеет информационной части. В информационных частях объектов - порций (тип 1000) содержатся элементы - наборы отрезков или контура. Изображенная структура выстраивается автоматически подпрограммами построения плоских образов. Элементом плоского образа является набор отрезков. Это может быть либо контур (контур помеченной грани), либо набор видимых отрезков, невидимых отрезков или отрезков части внешнего контура. Ограничение - контур помеченной грани

3533847.00026-02 33 01

- один элемент и, следовательно, должен помещаться в оперативной памяти. Размер порции устанавливается либо по параметру (если он указан), либо берется равным заданному при инициализации архива. Структура элемента - набор отрезков изображена на рис.24, где DE - значение первого слова элемента - общее число слов в нем; далее размещаются координаты концов отрезков, по 4 слова на отрезок. Эта структура элемента используется для представления видимых отрезков, невидимых отрезков и отрезков внешнего контура, которые формируются в подпрограмме SPT200.

Структура элемента - набор отрезков

DE, OTR1, OTR2, ..., OTR  
|  
(DE-1) слово

Рис.24

Структура элемента - помеченного контура изображена на рис.25.

Структура элемента - помеченная грань

...  
DE, KA \ / , DK, \ / ...  
KA слов | (DK-1) слово  
|  
(DE-1) слово

Рис.25

Конкретно в подпрограмме SPT200 при формировании помеченных контуров значение KA=4, но используется только последнее слово атрибута, в которое заносится значение Z-координаты вектора нормали к грани, по которой получен данный контур.

С элементами плоских образов работа ведется только в подпрограммах SPT200, SPT101. Вообще говоря, эта структура может измениться при развитии компонента АМТТ. В результате работы программы SPT330 формируется структура - набор отрезков, но каждый отрезок задается 6-ю словами. Эта структура преобразуется в структуру элемента - набор отрезков с помощью подпрограммы SPT205.

### 2.15.3. Специальное представление 2Д-моделей трехмерных объектов

Для удобства работы с плоскими образами была написана группа подпрограмм, которые используют специальное представление плоских образов. Структура общей 2Д-модели изображена на рис.26.

Архивная структура 2Д-модели

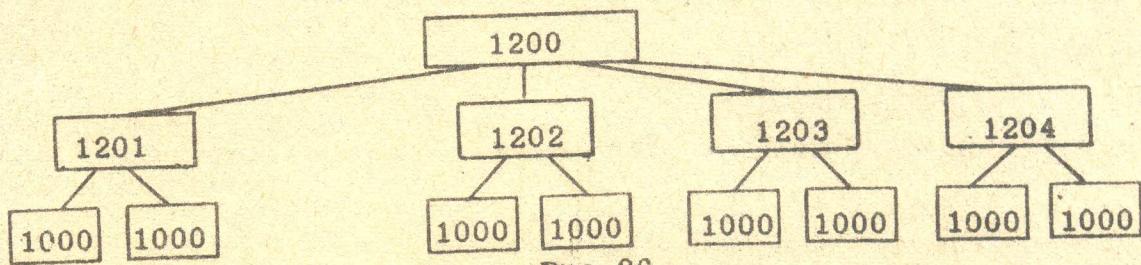


Рис.26

В прямоугольниках (см. рис. 26) дано условное обозначение Т-объектов, приведены значения их типов. Головной объект в информационной части содержит ряд параметров, описываемых ниже. Т-структуры с головными объектами следующего уровня являются плоскими образами, со структурой элементов, описанной в п. 2.15.2. Т-структура с типом головного объекта 1201 - набор видимых отрезков; 1202 - набор невидимых отрезков; 1203 - набор отрезков внешнего контура и 1204 - набор контуров помеченных граней. Вообще говоря, некоторые из этих плоских образов могут отсутствовать в 2Д-модели.

Длина информационной части головного объекта с типом 1200 может быть вычислена по следующей формуле:

$$25 + 3 * LK \quad (20)$$

где LK - число помеченных контуров. Структура информационной части следующая (в словах):

1 - общая длина информационной части;

2 - резервное слово;

3-10 - параметры, определяющие проекцию  
(X0, Y0, X1, Y1, X2, Y2, X3, Y3);

11-16 - габариты (DX, UX, DY, UY, DZ, UZ) исходного трехмерного объекта;

17-19 - резервные слова;

20 - тип линии визуализации видимых отрезков;

21 - тип линии визуализации невидимых отрезков;

22 - тип линии визуализации внешнего контура;

23-24 - параметры штриховки внешнего контура (угол наклона, расстояние);

25 - число помеченных контуров.

Далее размещаются параметры для каждого из помеченных контуров по три слова:

1 - тип линии визуализации контура;

2-3 - параметры штриховки контура.

Описанная выше информационная часть формируется при создании Т-структуры 2Д-модели (подпрограмма TD1200) и используется в подпрограммах визуализации. 2Д-модель-сечение (формируемая подпрограммой TD1400) рассматривается как 2Д-модель с одним помеченным контуром - сечением.

## 2.16. Описание программ для создания и обработки иерархических представлений трехмерных объектов и их плоских образов

В этом подразделе описываются программы, предназначенные для обработки 3Д-объектов в иерархическом представлении. Предполагается, что первоначально объекты создаются с помощью средств компонента ПА.

Для большинства описываемых подпрограмм может быть осуществлен контроль корректности выполнения. Контроль заключается в проверке значения десятого слова общего блока с именем /KIEL/ непосредственно после обращения к подпрограмме. Нулевое значение этого слова свидетельствует о правильной работе подпрограммы. В противном случае подпрограмма проработала некорректно. Как правило, это вызывается недостатком памяти.

2.16.1. Формирование иерархического представления 3Д-объектов

Подпрограмма SPT300 предназначена для преобразования объекта, созданного на оперативной памяти с помощью средств компонента ПА (многогранника) к иерархическому представлению.

Обращение:

CALL SPT300(KI,KJ,LP,KV,KA)

где KI - целая программная переменная - идентификатор нового объекта (тип 10300);

KJ - идентификатор многогранника, созданного средствами компонента ПА (тип 350);

LP - начальное значение атрибута, присваиваемое каждой грани - целое число;

KV - максимальный размер порции элементов для создаваемого объекта - длина кванта порции.

KA - количество слов атрибута, выделяемое каждой грани, значение атрибута заносится в последнее из этих слов; если KA=0, то LP=0. Рекомендуется KA=1;

Контроль корректности выполнения - значение десятого слова общего блока /KIEL/.

Результат:

KIEL(10)!=0: были препятствия для выполнения, при этом KI=0.

Подпрограмма SPT301 предназначена для линейного преобразования всех вершин многогранника, определяемого матрицей третьего порядка.

Обращение:

CALL SPT301(KI,A,L)

где KI - идентификатор многогранника (тип 10300);

A=A(3,3) - двумерный массив элементов матрицы преобразования;

L - целое - режим работы: если L=0, то вершины (векторы-столбцы) преобразуются матрицей A, если L=1, то транспонированной матрицей.

KIEL(10)!=0: преобразование не выполнилось.

Подпрограмма SPT302 предназначена для преобразования сдвига или масштабирования вершин многогранника.

Обращение:

CALL SPT302(KI,X,Y,Z,L)

где KI - идентификатор многогранника (тип 10300);

X,Y,Z - координаты вектора сдвига или масштабирования по осям OX,OY,OZ;

L - режим работы, целое: L=0 - сдвиг; L=1 - масштабирование.

Результат:

KIEL(10)!=0: преобразование не выполнено, объект может быть испорчен.

Подпрограмма SPT303 предназначена для вращения многогранника на заданный угол вокруг заданной оси.

Обращение:

3533847.00026-02 33 01

CALL SPT303(KI,X0,Y0,Z0,X1,Y1,Z1,F)

где KI - идентификатор многогранника (тип 10300);  
X0,Y0,Z0 )> начальная и конечная точки вектора,  
X1,Y1,Z1 ) определяющего ось вращения;  
F - угол вращения в градусах.Вращение осуществляется против часовой стрелки (при F>0),  
если смотреть из конца вектора оси в начало.

Результат:

KIEL(10)!=0: преобразование не выполнилось, объект KI испорчен.

Подпрограмма SPT304 предназначена для вычисления габаритного параллелепипеда для многогранника.

Обращение:

CALL SPT304(KI,DX,UX,DY,UY,DZ,UZ)

где KI - идентификатор многогранника (тип 10300);

DX,DY,DZ - минимальные значения координат объемлющего параллелепипеда;

UX,UY,UZ - максимальные значения координат объемлющего параллелепипеда; вещественные переменные, куда помещаются указанные значения.

Результат:

KIEL(10)!=0: результата нет.

Подпрограмма SPT305 предназначена для реализации теоретико-множественных операций над многогранниками.

Обращение:

CALL SPT305(KI,K1,K2,M1,M2,L,KV)

где KI - целая переменная - идентификатор результирующего многогранника; тип 10300, если после выполнения KI=0, то результат пуст (при KIEL(10) !=0);

K1,K2 - идентификаторы исходных многогранников (тип 10300);

M1,M2 - признаки установки меток граней результата, целые. Поскольку каждая грань результата является частью какой-либо грани одного из операндов, то ее метка в результате определяется исходной меткой и признаком M1 для грани K1 и M2 для грани K2. Признаки могут иметь следующие значения: -1 - соответствующие метки устанавливаются в 0; 0 - метки сохраняются; 1 - устанавливаются новые метки, равные 1.

L - тип операции: L=1 - пересечение, L=2 - объединение, L=3 - разность.

KV - максимальный размер порции элементов объекта-результата операции.

Результат:

KIEL(10)!=0: результирующий объект не создан.

Подпрограмма SPT306 предназначена для установки меток граней многогранника.

Обращение:

CALL SPT306(KI,L)

3533847.00026-02 33 01

где KI - идентификатор многогранника (тип 10300);  
 L - целое положительное число - значение метки, устанавливаемое каждой грани KI в поле атрибута (последнее слово).

Результат:

KIEL(10)!=0: метки не установлены.

Подпрограмма SPT331 предназначена для отсечения части многогранника плоскостью, двугранным или трехгранным углами.

Обращение:

CALL SPT331(KI,KJ,R,E1,E2,E3,L)

где KI - целая переменная, идентификатор создаваемого объекта (тип 10300) - результат отсечения исходного объекта KJ;

KJ - идентификатор исходного многогранника (тип 10300);

R=R(3) - вещественный массив длиной в три слова - координаты вершины угла отсечения;

E1=E1(3),E2=E2(3),E3=E3(3) - вещественные массивы длиной по три слова - координаты векторов, образующие правую тройку E1,E2,E3;

L - целое число, режим работы:

L=1 - отсечение плоскостью: отсекаемое полупространство - множество точек:  $\{xE1+yE2+zE3+R; x>0, y, z \text{ - любые}\}$

L=2 - отсечение двугранным углом: отсекаемое множество точек:  $\{xE1+yE2+zE3+R; x>0, y>0, z \text{ - любое}\}$

L=31 - отсечение трехгранным углом: отсекаемое множество точек:  $\{xE1+yE2+zE3+R; x>0, y>0, z>0\}$ .

Примечание. Если  $E1=(1.,0.,0.), E2=(0.,1.,0.), E3=(0.,0.,1.)$ ,  $R=(0.,0.,0.)$ , то при  $L=3$  из объекта вычитается "восьмушка" пространства ( $x>0, y>0, z>0$ ), при  $L=2$  - двугранный угол ( $x>0, y>0$ ) и при  $L=1$  - полупространство ( $x>0$ ).

Результат:

KIEL(10)!=0: операция не выполнилась.

## 2.16.2. Создание и обработка плоских образов

В комплекте АМТТ выделены два способа построения плоских образов трехмерного объекта: параллельное проецирование на плоскость и сечение объекта плоскостью. Произвольная аксонометрическая проекция определяется четырьмя двумерными векторами (двухэлементными массивами) E0, E1, E2, E3. Эти векторы задаются в двумерной системе координат и являются проекциями: E0 - начала трехмерной системы координат на плоскость проецирования, E1, E2, E3 - концов ортov осей OX, OY, OZ на плоскость проецирования.

Это задание существенно более удобно, чем отдельно задание плоскости проецирования и вектора проецирования. И кроме того, оно обеспечивает более широкий класс проекций. По векторам E1, E2, E3 строится матрица линейного преобразования P. Если преобразовать объект с помощью этой матрицы и затем сдвинуть его на вектор (E0(1), E0(2), 0), то мы получим 3D-объект (возможно искаженный по отношению к исходному), ортогональная проекция которого на плоскость XOY даст требуемый результат.

3533847.00026-02 33 01

Предусмотрены подпрограммы, позволяющие строить соответствующие матрицы преобразования.

Для сечения объекта плоскостью, последняя задается некоторой точкой и вектором нормали. В результате сечения получается набор трехмерных отрезков, который может быть преобразован в плоский образ, отбрасыванием Z координат (подпрограмма SPT205).

Подпрограмма SPT000 предназначена для построения матрицы преобразования трехмерного объекта.

Обращение:

```
CALL SPT000(E1,E2,E3,P,E)
```

где E1=E1(2),E2=E2(2),E3=E3(2) - двухэлементные массивы, определяющие требуемую проекцию;

P=P(3,3) - вещественный массив, в который будет помещена матрица требуемого линейного преобразования объекта;

E - положительная константа; если векторное произведение векторов, составленных из первых (вторых) элементов массивов E1,E2,E3 по длине меньше E, то P не вычисляется, т.к. аксонометрия вырожденная, в этом случае KIEL(10)!=0.

Подпрограмма SPT001 предназначена для вычисления обратной матрицы к матрице, полученной по подпрограмме SPT000.

Обращение:

```
CALL SPT001(P,Q,E)
```

где P=P(3,3) - исходная матрица;

Q=Q(3,3) - вещественный массив, куда будет записана матрица обратного по отношению к P преобразования;

E - положительный малый параметр, смысл тот же, что для подпрограммы SPT000.

Результат:

KIEL(10)!=0: матрица Q не вычислена.

Возможно преобразование "на месте", т.е. массив Q совпадает с P.

Преобразование объекта по заданной матрице возлагается на программиста, и может быть осуществлено с использованием подпрограмм, описанных в п. 2.16.2.

Подпрограмма SPT206 предназначена для вычисления габаритов проекции ЗД-объекта.

Обращение:

```
CALL SPT206(KI,DX,DY,DZ,E0,E1,E2,E3,GP)
```

где KI - идентификатор многогранника (тип 10300);

DX,DY,DZ - переменные, в которые будут помещены координаты вершины габаритного параллелепипеда, охватывающего исходный ЗД-объект, с минимальными значениями координат вдоль осей;

E0,E1,E2,E3 - двухэлементные вещественные массивы, задающие проекцию;

GP=GP(4) - вещественный массив, в который будут помещены координаты габаритного прямоугольника, охватывающего проекцию объекта: (GP(1),GP(2)) - левый нижний угол, (GP(3),GP(4)) - правый верхний - в координатной системе плоскости проецирования.

Результат:

KIEL(10)!=0: требуемые величины не вычислены.

Подпрограмма SPT200 предназначена для построения плоских образов 3D-объекта с использованием параллельного проецирования на плоскость XOY вдоль оси OZ.

Обращение:

CALL SPT200(KI,K1,K2,K3,K4,KV)

где KI - идентификатор многогранника (тип 10300);

K1 - целая переменная, если K1=0, то внешний контур проекции не строится;

K2 - целая переменная, если K2=0, то набор контуров проекций помеченных граней не строится;

K3 - целая переменная, если K3=0, то набор видимых отрезков проекции не строится;

K4 - целая переменная, если K4=0, то набор невидимых отрезков проекции не строится .

Если значения каких-либо переменных из K1,K2,K3,K4 равны 1, то эти переменные будут служить идентификаторами соответствующих плоских образов: внешнего контура, контуров проекций помеченных граней, набора видимых отрезков, набора невидимых отрезков, соответственно.

KV - целое, размер информационной части порции объектов результатов.

Результат:

KIEL(10)!=0: операция не выполнена.

Рекомендуется при обращении к подпрограмме SPT200 задавать ненулевым параметром только одну из переменных K1,K2,K3,K4, при этом уменьшаются требования к рабочей памяти. Число отрезков контуров помеченных граней не должно превышать KV/4 для каждого.

Подпрограмма SPT330 предназначена для построения сечения 3D-объекта плоскостью..

Обращение:

CALL SPT330(KI,KJ,R,H,KV)

где KI - целая переменная - идентификатор создаваемого набора отрезков сечения (отрезки трехмерные);

KJ - идентификатор исходного многогранника (тип 10300);

R=R(3) - вещественный массив, определяющий точку (любую) на плоскости сечения;

H=H(3) - вещественный массив, определяющий нормаль к плоскости сечения;

KV - максимальный размер порции набора отрезков сечения (T-объекта с именем KI).

Результат:

KIEL(10)!=0: сечение не построено.

Подпрограмма SPT205 предназначена для преобразования набора трехмерных отрезков (результат работы подпрограммы SPT330) в набор двумерных отрезков отбрасыванием Z-координаты; преобразование производится "на месте".

Обращение:

CALL SPT205(KI)

где KI - идентификатор набора трехмерных отрезков.

Результат:

3533847.00026-02 33 01

KIEL(10)!=0: преобразование не выполнено.

Ниже описываются две подпрограммы, предназначенные для модификации наборов отрезков. Эти подпрограммы удобно использовать непосредственно перед визуализацией набора отрезков.

Подпрограмма SPT101 предназначена для устранения взаимных наложений отрезков.

Обращение:

```
CALL SPT101(KI,E)
```

где KI - идентификатор набора двумерных отрезков, например, полученный после работы подпрограммы SPT200;

E - малое положительное число, определяющее совпадение отрезков: синус угла между ними меньше E и расстояние меньше E. Рекомендуется E = 0.01.

Результат:

KIEL(10)!=0: "склеивание" не выполнено.

Подпрограмма заменяет налагающиеся отрезки - одним. Преобразование набора отрезков осуществляется без образования новой Т-структурь.

Подпрограмма SPT1DM предназначена для "вычитания" одного набора отрезков из другого.

Обращение:

```
CALL SPT1DM(KI,K1,K2,E)
```

где KI - целая переменная-идентификатор создаваемого результирующего набора отрезков;

K1 - идентификатор исходного набора отрезков;

K2 - идентификатор "вычитаемого" набора отрезков;

E - малое положительное число, определяющее совпадение отрезков, смысл такой же, как и в SPT101.

Результат:

KIEL(10)!=0: операция не выполнилась, объект KI не создан.

Эту подпрограмму целесообразно использовать для вычитания набора видимых отрезков из набора невидимых, чтобы при последующей визуализации они не "накладывались" друг на друга.

### 2.16.3. Формирование 2Д-моделей

Все 2Д-модели, полученные на основании 3Д-моделей (виды, разрезы, сечения, аксонометрические проекции, виды с местными разрезами, аксонометрические проекции с разрезами) имеют с точки зрения пользователя компоненты АМТТ одну и ту же логическую структуру. 2Д-модель состоит из: группы видимых отрезков, группы невидимых отрезков, внешнего контура (обвода объекта) и нескольких, так называемых, помеченных контуров. В частных случаях в 2Д-модели может не быть помеченных контуров и/или внешнего контура, а также групп видимых и/или невидимых отрезков. Это зависит от типа программы формирования 2Д-модели, а также от желания самого пользователя.

Отметим, что группа видимых отрезков как множество точек на плоскости включает в себя внешний контур.

Помеченные контуры появляются в 2Д-модели при построении видов, разрезов и аксонометрических проекций (но не сечений). Они являются проекциями, так называемых, помеченных граней 3Д-модели.

3533847.00026-02 33 01

Такие грани получаются при специальном вычитании (отсечении) одного многогранника из другого.

Еще раз подчеркнем, что 2D-модель - это не модель всего чертежа. Это модель всегда только одного изображения, то есть либо это модель одного вида (основного, дополнительного, с местными разрезами или без), либо это модель одного разреза (пока только простого разреза), либо это модель одного сечения, либо это модель одной аксонометрической проекции (с разрезом или без).

Формирование видов и стандартных аксонометрических проекций осуществляется параллельным проецированием 3D-модели.

Для задания такого проецирования необходимо, вообще говоря, задать вектор направления взгляда и картинную плоскость. Везде в дальнейшем будет предполагаться, что картинная плоскость не пересекается с 3D-объектом и расположена за 3D-объектом относительно наблюдателя (рис. 27).

#### Параллельное проецирование 3D-модели



Рис. 27

Полученная в результате проецирования фигура в картинной плоскости будет являться множеством точек в трехмерном пространстве  $R^3$ . Чтобы получить двумерную фигуру, надо ввести на картинной плоскости некоторую двумерную систему координат, но обязательно такую, чтобы она при добавлении вектора, противоположного вектору направления взгляда, образовывала правую систему координат в  $R^3$ . Это требование необходимо для того, чтобы исключить возможность получения зеркально-симметричных проекций.

Будем полагать, что система координат на картинной плоскости прямоугольная, с единичными масштабами по осям. В дальнейшем она будет называться собственной системой координат 2D-модели.

Непосредственное задание собственной системы координат в картинной плоскости обычно неудобно, поэтому часто мы будем поступать наоборот, задавая в собственной системе координат векторы  $\tilde{e}_1$ ,  $\tilde{e}_2$ ,  $\tilde{e}_3$ , являющиеся образами базисных векторов  $e_1$ ,  $e_2$ ,  $e_3$  и задавая координаты точки  $O$  - образа начала системы координат  $O$  в  $R^3$  (рис. 28).

#### Определение собственной системы координат

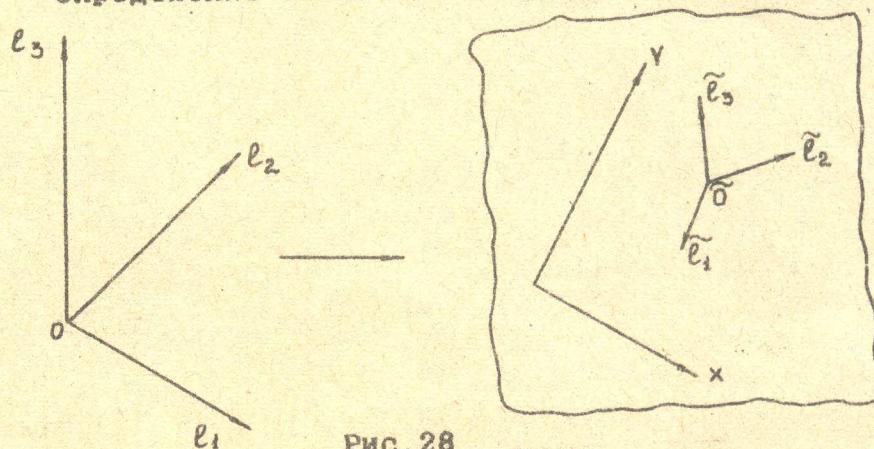


Рис. 28

Векторы  $e_1$ ,  $e_2$ ,  $e_3$  называются аксонометрическими осями.

Как известно, задание только одних аксонометрических осей  $e_1$ ,  $e_2$ ,  $e_3$  вполне достаточно для определения вектора направления взгляда и картинной плоскости с точностью до параллельного сдвига (т.е. нормали к картинной плоскости). В дальнейшем, описывая параметры подпрограмм, мы будем использовать оба способа задания параллельных проекций.

Во всех описываемых ниже подпрограммах будут использоваться следующие обозначения параметров, имеющие один и тот же смысл:

K13D - идентификатор исходного многогранника 3Д-объекта (тип 10300);

K12D - идентификатор формируемой или обрабатываемой 2Д-модели (тип 1200);

OXYZ(8) - вещественный массив, определяющий требуемую проекцию трехмерного пространства на плоскость: OXYZ(1),OXYZ(2) - координаты проекции начала системы координат; OXYZ(3),OXYZ(4) - координаты проекции конца орта оси OX; OXYZ(5),OXYZ(6) - координаты проекции конца орта оси OY; OXYZ(7), OXYZ(8) - координаты проекции конца орта OZ;

IB, IH, IVK, IK - признаки, определяющие нужно (>0) или нет (<=0) строить, соответственно: видимые, невидимые отрезки, внешний контур, контуры помеченных граней и включать их в 2Д-модель;

NP - резервный параметр, который не используется, может быть любая константа.

Далее при описании подпрограмм эти параметры не описываются.

Основные виды получаются прямоугольным проецированием на плоскость, параллельную одной из координатных плоскостей. Таких плоскостей три. Для каждой из них можно взять два перпендикулярных ей вектора направления взгляда, поэтому всего существует шесть основных видов.

Построение основных видов осуществляется подпрограммой TD1210.

Обращение:

CALL TD1210(K12D,K13D,N,DX,DY,IB,IH,IVK,IK)

где N=1,2,3,4,5,6 - номер основного вида, определяющий вектор направления взгляда. В табл.1 для каждого номера вида дано его название и вектор направления взгляда.

Таблица 1

Номер вида	Название вида	Направление взгляда
1	Главный вид (вид спереди)	-e2
2	Вид сверху	-e3
3	Вид слева	-e1
4	Вид справа	e1
5	Вид снизу	e3
6	Вид сзади	e2

$DX, DY$  - координаты точки образа начала системы координат при проецировании.

Результат:

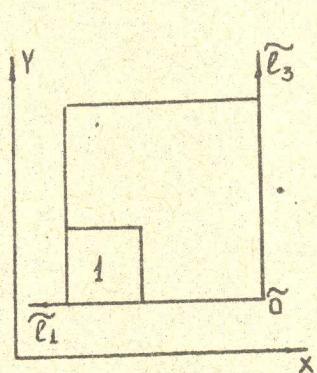
KIEL(10)!=0: 2D-модель не создана.

Перед обращением KI2D должно быть присвоено значение 0.

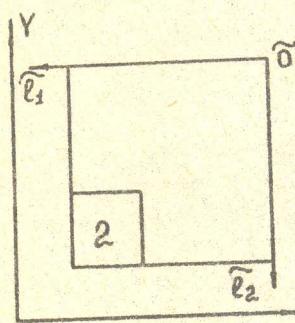
Так как задание вектора направления взгляда определяет (при прямоугольном проецировании) проекцию с точностью до поворота и сдвига в собственной системе координат, то на рис.29 приведено для каждого основного вида, формируемого подпрограммой TD1210, расположение аксонометрических осей  $\tilde{e}_1, \tilde{e}_2, \tilde{e}_3$  в собственной системе координат.

Одна из осей всегда нулевая, две другие параллельны координатным осям. ( $DX, DY$ ) - координаты точки  $B$  - образа начала системы координат в  $R^3$ .

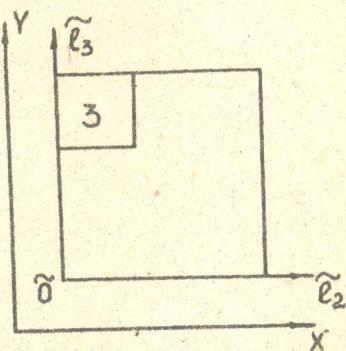
Расположение аксонометрических осей в собственной системе координат для каждого основного вида



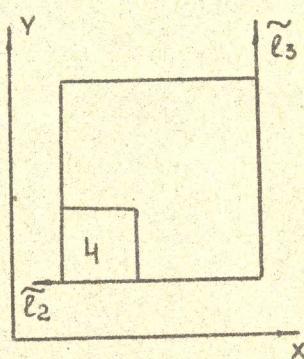
Вид спереди



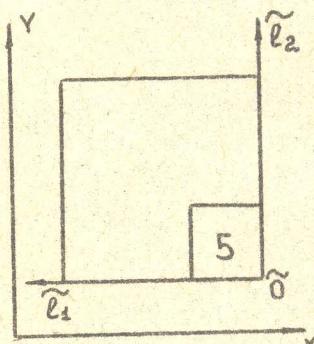
Вид сверху



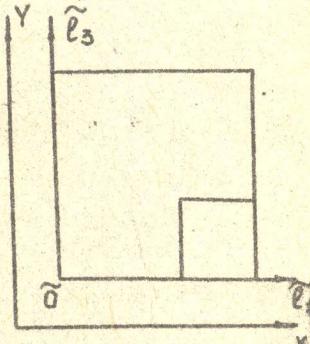
Вид слева



Вид справа



Вид снизу



Вид сзади

Рис.29

3533847.00026-02 33 01

Дополнительный вид формируется с помощью подпрограммы TD1220.  
Обращение:

```
CALL TD1220(KI2D,KI3D,I,X,Y,K,DX,DY,IB,IH,IBK,IK)
```

где I - параметр, определяющий координатную плоскость, в которой задается вектор проецирования: координатная плоскость всегда перпендикулярна I-ому орту ( $I=1,2,3$ );

X, Y - вектор направления взгляда, заданный в координатной плоскости;

DX, DY - те же, что и в TD1210;

K - целое, K=0,1,2,3,4 - параметр, задающий расположение аксонометрических осей.

Перед обращением к подпрограмме KI2D должно быть присвоено значение 0.

Дополнительный вид получается методом прямоугольного проецирования. Вектор направления взгляда всегда лежит в одной из трех координатных плоскостей и задается в ней координатами (X, Y).

Координатная плоскость определяется целым параметром I: она перпендикулярна оси e ( $I=1,2,3$ ).

Расположение аксонометрических осей при K=0 показано на рис. 30. Ось e параллельна вектору (X, Y) (если его расположить в

собственной системе координат) и направлена в ту же сторону. Две другие оси всегда ей перпендикулярны.

#### Дополнительный вид

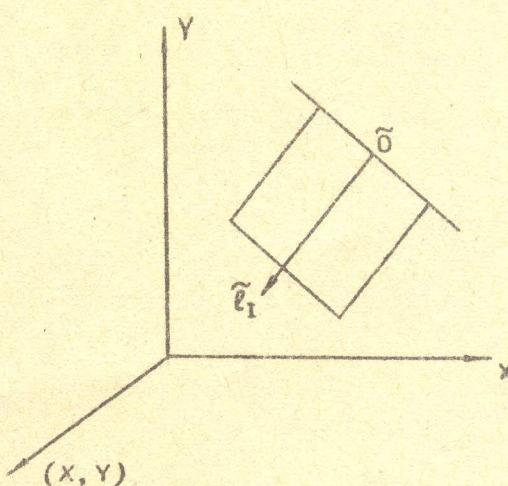


Рис. 30

(DX, DY) - координаты точки O. Значение параметра K=0 соответствует, так называемой, непосредственной проекционной связи дополнительного вида с одним из основных видов.

При K=1, 2, 3, 4 ось e располагается параллельно одной из координатных осей собственной системы координат (рис. 31).

3533847.00026-02 33 01

Параллельное расположение оси  $e$  в собственной  
системе координат

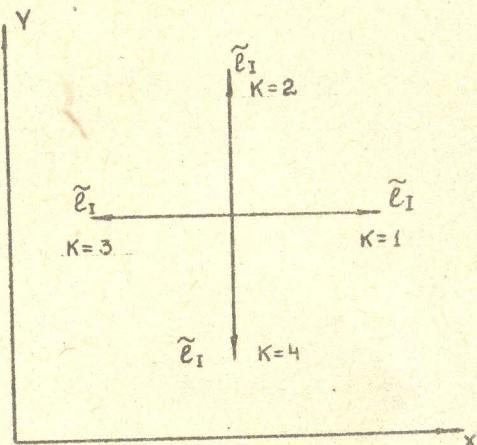


Рис. 31

Две подпрограммы построения сечений в какой-то мере очень похожи на подпрограммы построения основных и дополнительных видов. Только в отличие от видов, где картинная плоскость задается с точностью до параллельного сдвига, плоскость сечения должна обязательно быть задана точно и проходить через тело 3Д-объекта. Напомним, что в сечении изображается только то, что попадает в плоскость сечения. Следует учитывать, что при изменении вектора направления взгляда на противоположный, будет получаться зеркально-симметричная "картинка", что совершенно естественно, так как сечение - тело двумерное и "толщины" не имеет.

Отметим также, что в 2Д-модели сечения нет группы видимых и невидимых отрезков и внешнего контура, а всегда есть только один помеченный контур, т.е. сам контур сечения. Поэтому необходимость в параметрах IB, IH, IBK, IK отпадает.

Построение сечения плоскостью, параллельной одной из координатных осей, осуществляется подпрограммой TD1410.  
Обращение:

```
CALL TD1410(KI2D,KI3D,N,P,DX,DY)
```

где N - целое, N=1,2,3,4,5,6 - определяет так же, как и в TD1210, направление взгляда и соответствует номеру основного вида;  
P - расстояние (со знаком) от начала системы координат до плоскости сечения;

DX,DY - те же, что и в TD1210.

Перед обращением KI2D должно быть присвоено значение 0.

Подпрограмма TD1420 осуществляет построение сечений плоскостями, не параллельными координатным плоскостям.  
Обращение:

```
CALL TD1420(KI2D,KI3D,I,X,Y,X0,Y0,K,DX,DY)
```

Назначение параметров этой подпрограммы аналогично назначению параметров подпрограммы построения дополнительного вида.

Две подпрограммы построения разрезов так же, как и подпрограммы построения сечений, аналогичны подпрограммам построения основных и дополнительных видов. Для задания разреза надо задать вектор направления взгляда и перпендикулярную ему плоскость разреза. (Мы рассматриваем только простые разрезы). Это делается так же, как при задании плоскости сечения в подпрограммах TD1410, TD1420. Плоскость разреза делит тело на две части. Одна из частей тела удаляется. Остается та часть, в которую направлен вектор направления взгляда, если его начало расположить в плоскости разреза (рис. 32).

#### Определение плоскости разреза

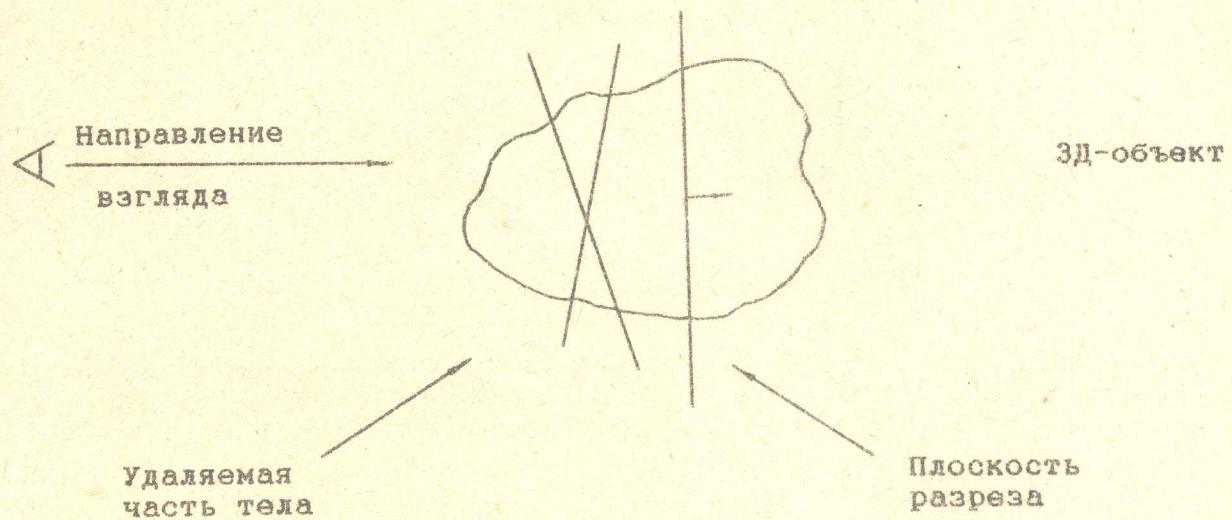


Рис. 32

После удаления части тела строится прямоугольная проекция оставшейся части тела по тому же вектору направления взгляда. Та грань тела, которая получилась в сечении, на этой проекции штрихуется. Таким образом, разрез является объединением сечения и проекции части тела, оставшейся за плоскостью сечения.

При построении 2Д-модели разреза так же, как при построении видов, формируются внешний контур и группы видимых и невидимых отрезков (если пользователь пожелает). Контур, полученный в сечении тела, формируется в 2Д-модели как помеченный контур. Этот контур получается с помощью обращения к теоретико-множественной операции отсечения, в результате которой грань, полученная отсечением, помечается. (Исходная 3Д-модель при построении по ней разреза не должна содержать помеченных граней).

Построение разреза плоскостью, параллельной координатной плоскости, осуществляется программой TD1310.

Обращение:

```
CALL TD1310(K12D,K13D,N,P,DX,DY,IB,IH,IVK,IK)
```

где  $N$  - целое,  $N=1,2,3,4,5,6$  определяет направление взгляда так же, как и в TD1210;

$P$  - расстояние (со знаком) от начала трехмерной системы координат до плоскости разреза;

$DX, DY$  - те же, что и в TD1210.

Перед обращением KI2D должно быть присвоено значение 0.

Подпрограмма TD1320 осуществляет построение разреза плоскостью, не параллельной координатной плоскости.

Обращение:

```
CALL TD1320(KI2D,KI3D,I,X,Y,X0,Y0,K,DX,DY,IB,IN,IBK,IK)
```

где  $I$  - целое,  $I=1,2,3$  определяет одну из осей OX, OY, OZ;

$X0, Y0$  - точка в координатной плоскости, перпендикулярной выбранной ( $I$ ) оси;

$X, Y$  - вектор в этой же плоскости;

$K$  - целое,  $K=0,1,2,3,4$ , определяет расположение проекции оси  $e$  (см. п. 2.16.3).

Плоскость разреза проходит через точку  $(X0, Y0)$  и перпендикулярна вектору  $(X, Y)$  - вектору направления взгляда.

Перед обращением KI2D должно быть присвоено значение 0.

Часто бывает необходимо на каком-нибудь виде сделать, так называемый, местный разрез, при котором отбрасывается не весь "кусок" тела, расположенный перед плоскостью разреза, а только некоторая его часть. В компоненте нет специальных подпрограмм построения видов с местными разрезами, поскольку местных разрезов на одном виде может быть несколько, но есть средства для их формирования с помощью нескольких обращений к подпрограммам комплекта. Пусть, например, требуется построить 2Д-модель главного вида с двумя местными разрезами (рис. 33). Для этого надо сначала построить два многогранника, являющихся прямоугольными призмами, в основании которых входит контур разреза (помеченный цифрами 1-2-3-4-1 на рис. 33), но лучше взять основание немного выступающим за габариты вида (помеченное цифрами 1-2-3-4-5-6-1, рис. 33). Призмы надо разместить так, чтобы они входили в тело на необходимую глубину, а затем вычесть их из тела с помощью операции отсечения (см. п. 2.16.1). В результате от каждой призмы в теле останется по четыре помеченных грани. После этого надо обратиться к подпрограмме построения основного вида (TD1210), которая построит 2Д-модель с двумя помеченными контурами, являющимися как раз нужными контурами разрезов. Остальные шесть помеченных граней в 2Д-модель не войдут, так как они параллельны вектору направления взгляда.

Построение вида с двумя местными разрезами

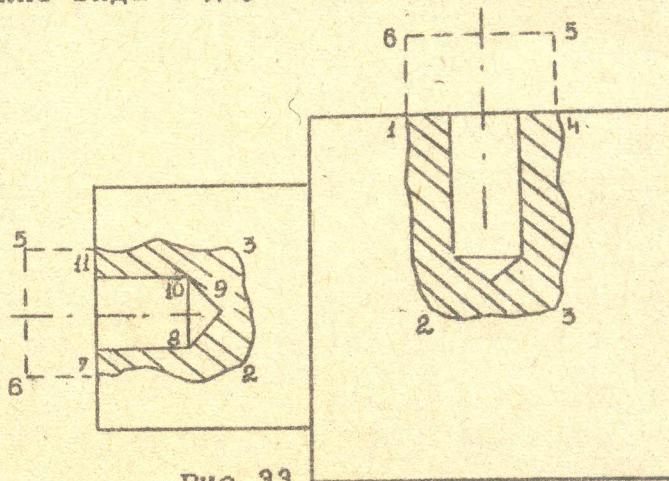


Рис. 33