

ТЕХТРАН[®]

СИСТЕМА АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ
УПРАВЛЯЮЩИХ ПРОГРАММ для СТАНКОВ с ЧПУ

Техтран[®]

Версия 9

Описание языка

Сopyright © 1993-2019 НИП-Информатика с сохранением всех прав

Техтран является зарегистрированным товарным знаком ООО "НИП-Информатика"

ООО "НИП-Информатика"
192102, С.-Петербург
ул. Фучика, д.4 лит. К
tehtran@nipinfor.ru
<http://www.tehtran.com>

Оглавление

Глава I Описание языка Техтран	12
1 Основные понятия	13
Алфавит	13
Синтаксис.....	14
Программа.....	15
Символические имена.....	15
Операторы.....	16
Метки	16
Комментарии.....	16
Числа	17
Служебные слова.....	18
Переменные.....	18
Служебные символы.....	18
Литералы	18
Типы данных.....	19
Массивы	20
Единицы измерения.....	21
Выражения.....	21
Арифметические выражения.....	21
Логические выражения.....	22
Строки	24
Условные выражения.....	24
Стандартные функции.....	24
Оператор присваивания.....	30
Система координат.....	31
2 Геометрические определения	32
Внутреннее представление геометрических объектов	33
Определение точки	35
Точка, определенная прямоугольными координатами.....	36
Точка, определенная пересечением двух прямых.....	36
Точка, определенная пересечением прямой и окружности.....	37
Точка, определенная пересечением двух окружностей.....	38
Точка пересечения окружности и ее радиуса.....	39
Точка, являющаяся центром окружности.....	40
Точка окружности на известном расстоянии от другой точки.....	41
Точка прямой, имеющая заданную координату.....	41
Точка прямой на определенном расстоянии от другой точки этой же прямой	42
Точка, заданная приращениями координат известной точки.....	43
Точка на заданном расстоянии от данной точки под углом к оси X.....	44

Точка, для которой ранее определен вектор является радиус-вектором	45
Точка, полученная в результате преобразования координат известной точки	45
Точка, определенная ранее заданной точкой	46
Точка пересечения прямой и контура, ближайшая к заданной точке	46
Точка пересечения прямой и сегмента контура, заданного номером	47
Точка пересечения окружности и контура, ближайшая к заданной точке	48
Точка пересечения окружности и сегмента контура, заданного номером	48
Точка пересечения сегментов двух контуров	49
Опорная точка контура	50
Текущая точка контура или траектории	51
Точка контура, ближайшая к заданной точке	51
Точка контура, ближайшая к другому контуру	52
Точка прямой, ближайшая к заданной точке	53
Точка окружности, ближайшая к заданной точке	54
Задание координаты Z в геометрических определениях точек	54
Определение прямой	56
Прямая, совпадающая с ранее определенной прямой	57
Прямая, проходящая через две точки, заданные координатами	57
Прямая, проходящая через две точки	58
Прямая, проходящая через точку и касающаяся окружности	59
Прямая, касающаяся двух окружностей	59
Прямая, проходящая через точку под углом к оси X	60
Прямая, проходящая через точку под углом к другой прямой	61
Прямая, проходящая через точку и перпендикулярная к другой прямой	62
Прямая, проходящая через точку и параллельная другой прямой	63
Прямая параллельная другой прямой на заданном расстоянии	63
Прямая параллельная другой прямой и касающаяся окружности	64
Прямая, касающаяся окружности под заданным углом к оси X	65
Прямая, проходящая через точку параллельно вектору	66
Прямая, параллельная одной из осей на заданном расстоянии	67
Прямая, параллельная одной из осей и имеющая заданную координату	68
Прямая, полученная в результате преобразования координат	68
Прямая, заданная отрезком контура	69
Прямая, проходящая через точку и касающаяся контура в точке, ближайшей к заданной	69
Прямая, проходящая через точку и касающаяся контура	70
Прямая, проходящая через точку и перпендикулярная к контуру в точке, ближайшей к заданной	71
Прямая, проходящая через точку и перпендикулярная к контуру	72
Прямая, ограничивающая контур и параллельная прямой	72
Прямая, ограничивающая контур и параллельная оси координат	73
Определение окружности	74
Окружность, совпадающая с ранее определенной окружностью	75
Окружность, определенная координатами центра и радиусом	76
Окружность, определенная радиусом и точкой центра	76
Окружность, определенная точкой центра, и касающаяся прямой	77

Окружность, определенная точкой центра, и проходящая через точку...	78
Окружность, проходящая через три точки.....	78
Окружность, определенная точкой центра, и касающаяся окружности..	79
Окружность заданного радиуса, касающаяся двух пересекающихся прямых	80
Окружность заданного радиуса, касающаяся прямой и проходящая через точку	81
Окружность заданного радиуса, проходящая через точку, с центром, лежащим на прямой.....	82
Окружность заданного радиуса, касающаяся прямой и окружности.....	83
Окружность заданного радиуса, касающаяся двух окружностей.....	85
Окружность, касающаяся трех прямых.....	87
Окружность заданного радиуса, проходящая через две точки.....	88
Окружность заданного радиуса, проходящая через точку и касающаяся окружности	89
Окружность, концентричная заданной окружности.....	90
Окружность, заданная дугой, являющейся сегментом контура.....	91
Окружность, полученная в результате преобразования координат.....	92
Определение вектора.....	92
Вектор, равный ранее определенному вектору.....	92
Вектор, определенный проекциями на оси и идущий из начала координат	93
Вектор, определенный своими начальной и конечной точками.....	93
Вектор, равный сумме двух других векторов	94
Вектор, равный разности двух других векторов.....	95
Вектор, равный произведению заданного вектора и числа.....	96
Вектор, являющийся радиус-вектором заданной точки.....	96
Вектор, заданный координатами своих начальной и конечной точек....	97
Вектор, параллельный заданной прямой.....	98
Вектор, равный произведению двух векторов.....	98
Вектор, полученный в результате преобразования координат.....	99
Определение матрицы.....	99
Матрица, равная ранее определенной матрице.....	101
Матрица, определяющая преобразование, обратное заданному преобразованию.....	101
Матрица, определяющая систему координат, полученную поворотом на угол вокруг оси	102
Матрица, определяющая систему координат, полученную параллельным переносом.....	103
Матрица, определяющая координаты, полученные параллельным переносом и поворотом.....	104
Матрица, являющаяся произведением двух других матриц.....	105
Матрица, определяющая перенос начала координат в точку, заданную явно или вектором.....	105
Матрица, определяющая поворот с последующим переносом начала координат	106
Матрица, определяющая симметрию относительно заданной прямой.	107

Матрица, определяющая систему координат, направления осей которой заданы векторами.....	107
Матрица, определяющая систему координат с началом в точке и с осями-векторами.....	108
Матрица, являющаяся зеркальным отображением исходной относительно плоскости.....	109
Матрица, определяющая масштабирование объекта.....	110
Определение плоскости.....	110
Плоскость, совпадающая с ранее определенной плоскостью.....	110
Плоскость, полученная преобразованием заданной плоскости в соответствии с матрицей.....	111
Плоскость, параллельная одной из координатных плоскостей и проходящая через точку.....	111
Плоскость, параллельная одной из координатных плоскостей на заданном расстоянии от нее.....	112
Плоскость, параллельная другой плоскости и проходящая через заданную точку.....	112
Плоскость, определенная коэффициентами ее нормального уравнения.....	112
Плоскость, проходящая через три точки, не лежащие на одной прямой.....	113
Плоскость, проходящая через заданную точку перпендикулярно вектору.....	113
Плоскость, параллельная двум векторам и проходящая через заданную точку.....	114
Вложенное определение.....	115
Массивы точек.....	115
Точки, равномерно расположенные между двумя точками.....	116
Перенос точек заданное число раз.....	117
Точки, полученные переносом точки под определенным углом с заданным интервалом.....	118
Точки, расположенные по контуру с заданным интервалом.....	119
Точки, равномерно расположенные по контуру.....	120
Точки, равномерно расположенные по окружности.....	120
Точки, равномерно расположенные по сектору окружности.....	121
Точки, расположенные по окружности с заданным угловым интервалом.....	122
Все опорные точки контура.....	123
Массив точек, упорядоченных контуром.....	123
Включение точек в массив точек.....	124
Исключение точек из массива точек.....	125
Преобразование координат точек.....	125
Перенос массива точек в каждую точку другого массива.....	126
Отсечение точек массива ограниченной областью.....	127
Точки пересечения контура и геометрического объекта.....	128
Точки пересечения геометрических объектов.....	129
Точки в узлах прямоугольной сетки, заполняющие прямоугольную область.....	129
Точки в узлах прямоугольной сетки, заполняющие ограниченную область.....	131
Массивы прямых.....	132

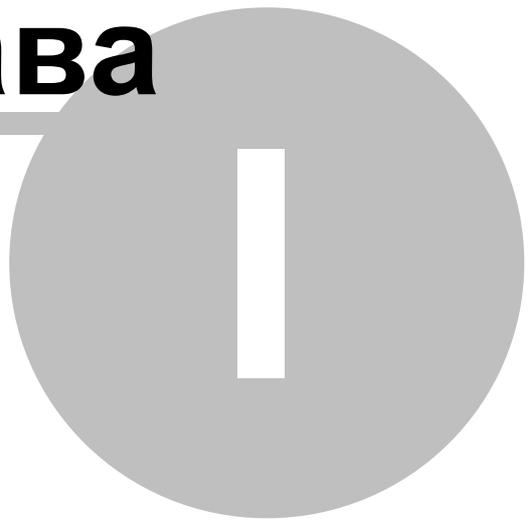
Заданное число прямых, параллельных данной прямой.....	133
Прямые, проходящие через все точки массива под заданным углом....	134
Прямые, соединяющие все точки массива с заданной точкой.....	135
Прямые, определяющие все отрезки контура.....	135
Прямые, ограничивающие контур и параллельные прямой.....	136
Прямые, ограничивающие контур и параллельные координатной оси	137
Прямые, равномерно расположенные по всем направлениям.....	137
Прямые, равномерно расположенные в диапазоне углов.....	138
Прямые, расположенные с заданным угловым интервалом.....	139
Массивы окружностей.....	139
Заданное число окружностей, концентричных данной с указанным интервалом	139
Окружности заданного радиуса с центрами в точках массива.....	140
Окружности, задающие все дуги контура.....	141
Массивы контуров.....	142
Семейство внутренних эквидистант к замкнутому контуру.....	142
Контур, полученные преобразованием контура заданное число раз..	143
Контур, полученные переносом контура заданное число раз.....	144
Контур, полученные переносом контура во все точки массива с поворотом на заданный угол.....	146
Заданное число последовательно построенных эквидистант к контуру	147
Массив контуров, заданный текстом.....	148
Текст, расположенный по дуге.....	149
Объединение областей.....	153
Исключение области из области.....	153
Пересечение областей.....	153
Построение границ следа контура.....	154
3 Описание инструмента.....	154
4 Программирование перемещений инструмента.....	156
Участок траектории, построенный из элементов.....	156
Исходное положение инструмента.....	159
Поточечное движение.....	159
Движение в точку.....	160
Движение, заданное приращением координат.....	161
Непрерывное движение.....	162
Начало непрерывного движения.....	164
Задание направления подвода.....	166
Продолжение непрерывного движения.....	169
Неявное определение ограничивающей поверхности.....	173
Сочетание поточечного и непрерывного движения.....	174
Фиктивное движение.....	175
Управление поверхностью детали.....	176
Отводы и возвраты инструмента.....	177
Преобразование траектории движения инструмента.....	178
Преобразование управляющих поверхностей.....	179
Преобразование точек траектории.....	182
Управление формированием эквидистанты и траектории.....	183

	Сопряжение соседних сегментов при построении эквидистанты.....	184
	Анализ наличия петель.....	185
	Сглаживание эквидистанты.....	186
	Сглаживание траектории.....	187
	Задание метода интерполяции и режима аппроксимации.....	187
	Точность аппроксимации.....	191
	Траектория, полученная объединением участков траектории.....	192
	Копирование участка траектории с преобразованием координат.....	193
	Перенос участка траектории заданное число раз.....	193
	Перенос участка траектории во все точки массива точек.....	194
5	Управляющие операторы.....	194
	Оператор безусловного перехода.....	195
	Условный оператор.....	195
	Условный оператор с альтернативой.....	196
	Условный оператор с несколькими альтернативами.....	198
	Оператор цикла со счетчиком.....	199
	Оператор цикла с условием.....	200
	Вложенные циклы.....	202
	Операторы прерывания выполнения.....	203
6	Специальные операторы.....	203
	Синонимы.....	204
	Вставка текста.....	204
	Управление версиями.....	205
7	Контур.....	205
	Определение контура.....	206
	Контур построенный из элементов.....	207
	Оператор поточечного движения в описании контура.....	208
	Операторы непрерывного движения в описании контура.....	209
	Оператор включения контура.....	212
	Контур являющийся сплайном.....	216
	Контур эквидистантный заданному.....	219
	Контур, имеющий форму прямоугольника.....	220
	Контур, имеющий форму окружности.....	221
	Контур, имеющий форму эллипса.....	222
	Контур, имеющий форму прямого паза.....	223
	Контур, имеющий форму радиусного паза.....	224
	Контур, имеющий форму сектора.....	225
	Контур, имеющий форму правильного многоугольника.....	226
	Габаритный контур.....	227
	Изменение направления обхода контура.....	227
	Замыкание контура путем сопряжения его концов.....	228
	Движение по контуру.....	228
	Подход к контуру.....	229
	Применение операторов непрерывного движения.....	230
	Движение по незамкнутому контуру.....	232
	Движение по табулированной кривой.....	233

Встраивание фасок и скруглений в контур	236
Траектория как контур	238
8 Макросы	239
Структура макросов	239
Параметры макроса	240
Заголовок макроса	242
Вызов макроса	243
Вложенные вызовы	246
Локализация переменных в макросе	247
Использование меток в макросах	248
9 Управление выводом результатов	248
Трассировка программы	249
Установка режима вывода траектории движения инструмента	251
Вывод значений переменных или параметров геометрических объектов	252
10 Операторы постпроцессора	252
Начало и конец программы	255
Вызов постпроцессора	255
Управление программноносителем	256
Разбиение программы.....	256
Изменение порядка нумерации кадров.....	257
Пропуск кадров.....	258
Задание длины лидера.....	258
Перемотка ленты.....	258
Создание маркера в программе.....	258
Управление сменой инструмента	259
Выбор инструмента.....	259
Смена инструмента.....	260
Разгрузка инструмента.....	260
Возврат инструмента.....	261
Ввод и отмена коррекции	261
Коррекция на радиус.....	261
Коррекция на длину.....	262
Коррекция по осям.....	262
Парная коррекция.....	263
Отмена коррекции.....	263
Управление подачей	264
Ускоренная подача.....	264
Рабочая подача.....	265
Управление шпинделем	266
Включение и выключение шпинделя.....	266
Скорость шпинделя.....	266
Вспомогательные функции.....	267
Поворот шпинделя на фиксированный угол.....	268
Управление охлаждением	268

Стандартные циклы	269
Задание безопасного расстояния.....	269
Сверление	270
Глубокое сверление.....	270
Цекование и зенкование.....	271
Коническое зенкование.....	271
Нарезание резьбы.....	272
Растачивание	273
Развертывание	273
Зенкерование	274
Включение и выключение цикла.....	274
Останов станка	275
Подготовительные и вспомогательные функции	276
Управление поворотным столом	277
Вставка текста	278
Вставка текста в управляющую программу.....	278
Вставка текста в выходной документ.....	279
Комментарий	279
Система координат станка	279
Задание местной системы координат смещениями.....	280
Задание местной системы координат функцией.....	280
Дополнительная ось.....	280
Сдвиг системы координат станка.....	281
Способ расчета координат.....	281
Управление резкой	282
Управление электроэрозионной обработкой.....	282
Управление термической резкой.....	282
Задание ширины реза.....	283
Включение и выключение резаков.....	283
Управление боковыми резаками.....	283
Управление разметкой.....	284
Специальные функции постпроцессора	284
Пример использования операторов постпроцессора	285
11 Примеры программирования обработки	286
Пример программирования простейшей траектории	287
Пример использования матриц	288
Пример использования контуров	289
Пример использования циклов и преобразования траектории	290
Пример программирования сплайнов	293
Пример использования макросов	294
Предметный указатель	297

Глава



1 Описание языка Техтран

О чем эта книга? В книге описан язык программирования обработки на станках с ЧПУ Техтран. Язык позволяет описывать геометрию обрабатываемой детали, траекторию перемещения инструмента и технологию обработки. Программа на Техтране представляет собой множество операторов, состоящих из элементов языка (служебных слов, чисел, имен), строящихся по заданным синтаксическим правилам. Синтаксис Техтрана подобен синтаксису языка АРТ (Automatically Programmed Tools).

Для чего нужен язык? Несмотря на неоспоримое преимущество интерактивного построения модели обработки перед текстовым описанием, отказ от языка программирования во многих современных САМ системах, представляется не оправданным. Начальный этап разработки УП – получение правильной геометрии детали, действительно, не требует работы с текстом. Но далее следует этап доработки программы, заключающийся в корректировке режимов обработки, вводе и удалении технологических команд, вставке фрагментов текста в кодах системы ЧПУ. Как правило, подобные изменения вносятся не в модель, а непосредственно в УП, что приводит к неадекватности УП и модели. Наличие файла с исходным текстом программы позволяет, во-первых, легко внести подобные изменения в модель обработки и, во-вторых, сохранить соответствие УП и модели. Кроме того, язык программирования включает такие стандартные средства организации логики программы как циклы, переходы, условные операторы, которые позволяют не только легко программировать вычисления, но и существенно сократить объем модели. Важным свойством современных САМ-систем является наличие средств программирования макрокоманд, что требует того или иного текстового представления. Макросредства позволяют описывать обработку типовых элементов в параметрическом виде и хранить такие фрагменты программ на диске в виде отдельных файлов.

В данном документе приводится подробное описание базовой версии Техтрана, включающей универсальные средства, которые могут применяться при любом виде обработки.

Подробное описание расширений Техтрана используемых при программировании фрезерной, токарной и электроэрозионной обработки приведено в книгах Фрезерная обработка, Токарная обработка, Электроэрозионная обработка.

Темы этого раздела:

- [Основные понятия](#)^[13]
- [Геометрические определения](#)^[32]
- [Описание инструмента](#)^[154]
- [Программирование перемещений инструмента](#)^[156]
- [Управляющие операторы](#)^[194]
- [Специальные операторы](#)^[203]
- [Контуры](#)^[205]
- [Макросы](#)^[239]
- [Операторы постпроцессора](#)^[252]
- [Примеры программирования обработки](#)^[286]

1.1 Основные понятия

В данном разделе вводится терминология, используемая в дальнейшем, определяются основополагающие элементы языка и языковые конструкции.

Темы этого раздела:

- [Алфавит](#)^[13]
- [Синтаксис](#)^[14]

- [Программа](#)^[15]
- [Символические имена](#)^[15]
- [Операторы](#)^[16]
- [Метки](#)^[16]
- [Комментарии](#)^[16]
- [Числа](#)^[17]
- [Единицы измерения](#)^[21]
- [Служебные слова](#)^[18]
- [Переменные](#)^[18]
- [Служебные символы](#)^[18]
- [Литералы](#)^[18]

- [Типы данных](#)^[19]
- [Массивы](#)^[20]
- [Выражения](#)^[21]
- [Стандартные функции](#)^[24]
- [Оператор присваивания](#)^[30]
- [Система координат](#)^[31]

1.1.1 Алфавит

Для записи программы используются следующие символы:

буквы русского алфавита **A – Я**

буквы латинского алфавита **A – Z**

цифры **0, 1, 2, 3, 4, 5, 6, 7, 8, 9**

символы

–	пробел
,	запятая
.	точка
:	двосточие
=	равно

+	плюс
-	минус
*	звездочка
'	апостроф
#	номер
%	процент
(левая скобка
)	правая скобка
&	коммерческое И
;	точка с запятой
/	наклонная черта
_	подчеркивание

1.1.2 Синтаксис

Каждый оператор языка формируется из операндов в соответствии с синтаксическими правилами языка.

При описании форматов операторов использованы следующие правила и условные обозначения:

- Служебные слова записываются прописными буквами обычным полужирным шрифтом:

ТОЧКА, ВПЕРЕД, ПОДАЧА

- Параметры записываются строчными буквами курсивом:

прямая, окружность, число

- Необязательные элементы заключены в квадратные скобки []:

[**ПЕРЕСЕЧ**], [, **МАКСЛИН**, *длина*]

- Альтернативные варианты записи параметров и модификаторов заключены в фигурные скобки { }:

{ **ВНЕ**
 ВНУТРИ }

- Если необязательный элемент имеет варианты, то внутренние фигурные скобки опускаются:

[**НЕКВАДР**
 ' **КВАДР**]

1.1.3 Программа

Программа на Техтране представляет собой последовательность операторов и комментариев, описывающих геометрию детали, траекторию перемещения инструмента и технологию обработки.

Программа на Техтране имеет следующую структуру:

ДЕТАЛЬ [название программы] текст программы КОНЕЦ

Программе на Техтране может соответствовать одна или несколько управляющих программ.

Название программы – заключенный в апострофы текст, предназначенный для идентификации программы. Его наличие необязательно, т.к. он имеет характер комментария.

Текст программы состоит из последовательности строк. В одной строке может быть записан один или несколько операторов, между которыми должен стоять символ “точка с запятой”(;). Часть оператора может быть перенесена на следующую строку. Для этого, после прерванной части оператора, необходимо ввести символ продолжения – “процент”(%). Число строк продолжения не ограничивается.

Текст оператора или комментария может начинаться с любой позиции. Элементы оператора отделяются друг от друга пробелами или запятыми. Число пробелов между элементами оператора не ограничено.

В программу могут быть вставлены из внешних файлов фрагменты текста и макросы.

Для удобства чтения программ и ускорения их отладки рекомендуются оформлять текст придерживаясь следующих правил:

- выделять отступами, пустыми строками, комментариями функциональные части программы, соответствующие отдельным переходам обработки, циклам повторения операторов и т.д.;
- размещать операторы геометрических определений в той же части программы, где описывается траектория;
- присваивать имена переменным, геометрическим объектам, меткам, учитывая их назначение.

1.1.4 Символические имена

Для идентификации в программе переменных величин (геометрических, вещественных, логических переменных, меток операторов, макросов) используют символические имена.

Символическое имя - это последовательность букв и цифр, начинающаяся с буквы. Значащими являются только первые 8 символов. Символическое имя должно быть уникальным в программе и описывать один объект или класс объектов. Пробелы внутри имени не разрешены.

Примеры символических имен:

АБ	верно;
КР15	верно;
К152АБ	верно;
К_1	верно;
К12345678910	верно;
МК<1	неверно;
123К22	неверно;

Имена **М2345678ККК** и **М2345678ТТТ** являются допустимыми, но они не могут быть использованы для идентификации различных объектов, т.к. первые 8 символов у них совпадают.

1.1.5 Операторы

Оператор – основная функциональная единица языка. Он предназначен как для описания операций, которые следует выполнить, так и для установления свойств данных, идентифицируемых с помощью символических имен.

1.1.6 Метки

Метка является символическим именем и предназначена для отметки любого оператора программы, для того чтобы на него можно было сослаться в других операторах данной программы. Метка должна предшествовать помечаемому оператору и может находиться либо в одной строке с оператором, либо на отдельной строке. После метки должен находиться служебный символ "двоеточие"(:). Переход на отмеченный оператор выполняется по оператору **НАМЕТКУ** *имя метки*.

1.1.7 Комментарии

Комментарий используется для пояснения действий, выполняемых в программе. Комментарии не обрабатываются системой, а только выводятся на экран или на печать при просмотре и распечатке исходного текста или результатов его трансляции. Комментарий может находиться как в одной строке с операторами исходной программы, так и занимать отдельную строку. Признаком начала комментария является служебный символ "номер"(**#**)

Возможно задание многострочного комментария. Он должен начинаться с комбинации символов (**/#**) и заканчиваться комбинацией (**#/**).

Пример:

```
/#  
    Это  
    многострочный  
    комментария  
#/
```

1.1.8 Числа

В Техтране не существует разницы между целыми и вещественными числами. Число представляется в виде последовательности цифр, среди которых может находиться точка или буква E. Точка отделяет целую часть от дробной, а буква E -мантиссу числа от его порядка. Перед самым числом и перед его порядком может стоять знак "минус"(-). Пробелы и другие символы внутри числа недопустимы, например:

1.23 -верно;
 2.15E5 -верно;
 -3E-6 -верно;
 3_E2 -неверно;
 E4 -неверно.

Число может включать в себя учет класса точности, задаваемого на чертеже. Для этого пользователь при вводе размеров может задавать их с указанием условного обозначения класса точности (квалитета).

Формат:

*размер * класс точности*

Параметры:

Параметр	Описание
<i>размер</i>	Значение (величина) размера.
<i>класс точности</i>	Литерал с указанием условного обозначения класса точности (квалитета).

 В обозначении квалитета должны использоваться только латинские буквы.

Размер, введенный с указанием условного обозначения класса точности, будет пересчитан системой на размер, находящийся в середине поля допуска указанного класса точности. Это позволяет избежать трудоемких вычислений и необходимости использования справочников или ГОСТ. Система проверяет соответствие введенного размера и условного обозначения класса точности (квалитета) значениям, имеющимся в базе данных. Таблицы диапазонов размеров и условных обозначений классов точности сделаны открытыми и могут корректироваться и дополняться пользователями.

Примеры:

110*^h7⁷ для системы вала или 110*^H7⁷ для системы отверстия.

32*^e8⁸ - задание размера 32 с квалитетом e8 (системы вала). Система пересчитает этот размер на размер, находящийся в середине поля допуска указанного класса точности:

$$32+(-0.050+(-0.089))/2=31.9305$$

Для ввода диаметрального размера можно задать 32*^e8⁷/2. В этом случае система произведет расчет по следующей формуле:

$$(32+(-0.050+(-0.089))/2)/2= 15,96525$$

1.1.9 Служебные слова

Служебные слова являются зарезервированными словами и составляют словарь языка Техтран. Они используются для написания операторов программы и состоят только из букв русского алфавита. Служебные слова делятся на:

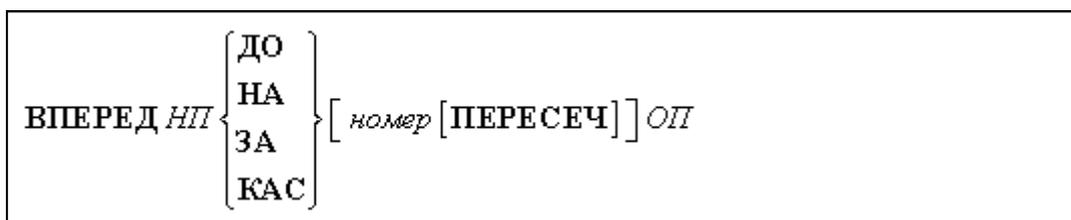
- *главные*
- *вспомогательные*
- *модификаторы*

Главное слово определяет оператор.

Вспомогательное слово поясняет параметр оператора.

Модификатор задает параметр, имеющий predetermined набор значений.

Пример:



В этом операторе движения главное слово – **ВПЕРЕД**, модификаторы – **ДО**, **НА**, **ЗА**, **КАС**, вспомогательное слово – **ПЕРЕСЕЧ**.

Служебное слово не может быть использовано в качестве метки, переменной или имени. Список служебных слов приведен в Приложении 1.

1.1.10 Переменные

Переменная в Техтране- это символическое изображение величины, которая может принимать любые допустимые для ее типа значения. Основными характеристиками переменной являются ее *имя* и *тип*.

1.1.11 Служебные символы

Служебные символы. Применение служебных символов в программе зависит от контекста. Описание их функций приводится по мере надобности в соответствующих разделах. В данную группу входят следующие символы: () (.) (=) (+) (-) (*) (/) (&) (;) ((:) (') (#) (%).

1.1.12 Литералы

Литералом является заключенный в апострофы произвольный текст, не содержащий символа #. В литералах могут быть использованы любые символы (например, латинские буквы, !, >, < и т.п.). При необходимости использования апострофа в качестве одного из символов литерала, его нужно записать два раза подряд. Например: 'ABCDE'РОН' определяет литерал, состоящий из символов ABCDE'РОН.

Литералы используются, в основном, для задания текстовых параметров макросов, а также в операторах **ДЕТАЛЬ**, **СТАНОК**, **ПЕЧ**, **ППФУН** и др.

1.1.13 Типы данных

Возможности языка программирования в значительной степени определяются набором допустимых типов данных. Язык Техтран допускает применение следующих типов данных:

- *вещественный*
- *логический*
- *точка*
- *прямая*
- *вектор*
- *окружность*
- *плоскость*
- *матрица*
- *контур*

Тип каждой используемой в программе переменной фиксирован и не допускает переопределения, т.е. на протяжении всей программы каждая переменная обозначает данное только одного типа.

Тип переменной описывается явно строкой вида:

```
тип имя1, имя2, имя3,...
```

Параметры:

Параметр	Описание
<i>тип</i>	ВЕЩ ЛОГИЧ ТОЧКА ПРЯМАЯ ВЕКТОР ОКРУЖН ПЛОСК МАТР КОНТУР
<i>имя₁, имя₂, имя₃</i>	Имена переменных.

В Техтране для переменных имеющих тип **ВЕЩ**, **ТОЧКА**, **ПРЯМАЯ**, **ОКРУЖН** предусмотрена возможность назначения типа неявно (по умолчанию). Переменные с именами, начинающимися с символов **ТЧ**, считаются неявно описанными как точки, с символов **ПР** – как прямые, с символов **КР** – как окружности. Все остальные переменные определяются неявным образом как вещественные числа.

Допускается совмещать описание типа переменной и оператор присваивания, например,

оператор **ТОЧКА Т1=-10,0,25** равносильен двум операторам:

ТОЧКА Т1
Т1 = -10,0,25

1.1.14 Массивы

Данные одного типа могут объединяться в массивы. Массивы могут быть одномерными и двумерными. Каждый, используемый в программе, массив должен быть описан с указанием типа, имени, числа индексов (размерности массива) и максимального значения каждого из индексов.

Формат:

```
тип имя(индекс1 [индекс2])
```

Параметры:

Параметр	Описание
<i>тип</i>	Тип переменных, объединенных в массив. Объединение в массив допускается для любых объектов, кроме объектов типа матрица (МАТР).
<i>имя</i>	Имя массива (любое допустимое в Техтроне имя).
<i>индекс₁, индекс₂</i>	Индексы массива- одно или два числа, заключенные в круглые скобки. Индексы определяют размерность массива (одно- или двумерный, соответственно) и максимальное количество элементов в массиве. Минимальным значением индексов является единица; максимальная величина индекса, а следовательно, и число элементов в массиве, не ограничены.

Например, оператор **ВЕЩА (16)** описывает одномерный массив, состоящий из 16 вещественных чисел, каждому из которых присвоено имя переменной **А**. Оператор **ПРЯМАЯ ПР11 (5,6)** описывает двумерный массив 30ти прямых с именем **ПР11**.

Элемент массива имеет имя массива и конкретный индекс (индексы), указанный за именем в круглых скобках. Например, **А(12), ПР11(3,2)**.

Индекс элемента массива не является частью его имени, поэтому запись вида **М2345678(1+22*А)** не является ошибочной с точки зрения допустимости длины имени. Индекс не может быть опущен, т.е. одно имя не представляет первый элемент массива. В качестве индекса может быть использовано любое арифметическое выражение. Число индексов и их значения должны соответствовать описанию массива. Например, если значение переменной **И=2**, то запись **ПР11(3,И+3)** именуется элемент массива **ПР11(3,5)**, а запись вида **ПР11(3)** также, как и запись вида **ПР11(1,6+И)**, вызывает ошибку трансляции.

Кроме стандартного, существует специальный формат объявления массива без указания размерности:

`тип имя()`

Этот формат применяется только для одномерных массивов, в случае когда размерность заранее не известна, например, в геометрических определениях массивов: **ТОЧКА T()**. Истинная размерность массива в таком случае определяется количеством построенных объектов.

1.1.15 Единицы измерения

Линейные размеры измеряются в миллиметрах, угловые – в градусах. Параметры, назначающие линейные и угловые размеры, задаются целыми и десятичными долями миллиметров и градусов.

1.1.16 Выражения

Выражение – это формула для вычисления значения. Выражение представляет собой последовательность чисел, переменных, элементов массивов и функций, разделенных знаками операций ('+', '—', '*' и т.д.) и круглыми скобками. Выражение имеет тип соответствующий вычисляемому значению. В Техтране различают арифметические (вещественные), логические и строковые выражения.

Темы этого раздела:

- [Арифметические выражения](#)^[21]
- [Логические выражения](#)^[22]
- [Строки](#)^[24]
- [Условные](#)^[24] [выражения](#)^[24]

1.1.16.1 Арифметические выражения

Вместо чисел в программе могут быть использованы арифметические выражения. Простейшими арифметическими выражениями могут быть:

- *число*
- *вещественная переменная*
- *элемент вещественного массива*
- *обращение к стандартной вещественной функции*
- *арифметическое выражение, заключенное в скобки*

Более сложные арифметические выражения могут быть образованы из простейших при помощи следующих знаков арифметических операций:

- + сложение
- вычитание
- * умножение
- / деление

** возведение в степень

// остаток от деления

Арифметические выражения выполняются слева направо в соответствии с приводимыми ниже приоритетами операций:

1 вычисление функций

2 возведение в степень

3 умножение и деление

4 сложение и вычитание

Приоритет используется для того, чтобы определить, какая из двух последовательных операций выполняется первой. Для изменения порядка вычислений следует применять круглые скобки.

Например, выражение

$$6a^2+2b(c^3-d^3)$$

$$(a^2+b^2)\sin\varphi$$

записывается следующим образом:

$$(6*A**2+2*B*(C**3-D**3))/((A**2+B**2)*\text{СИН}(\Phi))$$

 Следует помнить, что отрицательное число допускается возводить только в целую степень.

1.1.16.2 Логические выражения

Язык Техтран допускает использование логических выражений в *условных операторах ЕСЛИ*, *операторах цикла ПОВТОР*, *операторах логического присваивания*.

Значением логического выражения является **ЛОЖЬ** или **ИСТИНА**. Простейшим логическим выражением может быть:

- логическая константа **ЛОЖЬ** или **ИСТИНА**
- логическая переменная
- элемент логического массива
- оператор корректности
- отношение
- логическое выражение, заключенное в скобки

Отношения образуются путем объединения двух арифметических выражений с помощью операций отношения, которые изображаются следующим образом:

БЛ или **>** больше

Ш

БРВ или **>=** больше или равно

МН или $<$ меньше
Ш

МР или \leq меньше или равно
В

РВН или $=$ равно

НРВ или \neq не равно

Например, выражение $(K+1) \text{ РВН } 5$ принимает значение **ИСТИНА** при $K=4$, в остальных случаях оно принимает значение **ЛОЖЬ**.

Отношения при действиях со строками могут быть заданы операциями '=' или '<>'.
Например, выражение $(\text{ТипИнстр} = \text{'ФРЕЗА'})$ принимает значение **ИСТИНА** только тогда, когда строковая переменная **ТипИнстр** имеет значение **'ФРЕЗА'**, в остальных случаях это выражение принимает значение **ЛОЖЬ**.

Логические выражения образуются из простейших путем использования логических операций. В языке Техтран существуют три логические операции:

НЕ отрицание

ЛИ логическое умножение

ИЛ логическое сложение
И

Выполнение логического выражения производится слева направо, после арифметических операций и в соответствии с приводимыми ниже приоритетами:

- 1 вычисление функций
- 2 возведение в степень
- 3 умножение и деление
- 4 сложение и вычитание
- 5 операции отношения **БЛШ**, **БРВ**, **МНШ**, **МРВ**, **РВН**, **НРВ**
- 6 логическая операция **НЕ**
- 7 логическая операция **ЛИ**
- 8 логическая операция **ИЛИ**

Для изменения порядка выполнения логического выражения следует применять круглые скобки. Логическое выражение, к которому относится операция **НЕ**, должно заключаться в скобки, если оно содержит не менее двух операндов.

1.1.16.3 Строки

В качестве строковых параметров можно использовать следующие элементы языка:

- литерал
- строковую переменную
- обращение к стандартной строковой функции

1.1.16.4 Условные выражения

Вместо выражений в Техтране могут быть использованы условные выражения.

Условное выражение - это конструкция вида:

ЕСЛИ логическое выражение **ТО** выражение₁ **ИНАЧЕ** выражение₂

Если логическое выражение принимает значение ИСТИНА, то условное выражение принимает значение выражение₁, иначе выражение₂. В зависимости от контекста условные выражения могут быть вещественными, логическими или строковыми.

Пример:

ВЕЩ А = ЕСЛИ (Направление = ПОЧС) ТО 3 ИНАЧЕ 4
СТРОКА Б = ЕСЛИ (Направление = ПОЧС) ТО 'M03' ИНАЧЕ 'M04'
ЛОГИЧ В = ЕСЛИ (X > 5) ТО ИСТИНА ИНАЧЕ X=4

Сокращенная запись условного выражения имеет вид:

логическое выражение ? выражение₁ : выражение₂

В сокращенной записи предыдущий пример будет иметь вид:

ВЕЩ А = Направление = ПОЧС ? 3 : 4
СТРОКА Б = Направление = ПОЧС ? 'M03' : 'M04'
ЛОГИЧ В = (X > 5) ? ИСТИНА : X=4

1.1.17 Стандартные функции

Язык Техтран содержит ряд стандартных функций, используемых при вычислениях.

Функция выполняется указанием в выражении ее имени и аргументов (параметров).

имя(аргумент₁ [, аргумент₂ ...]).

Результат, вычисленный функцией, замещает в выражении обращение к ней. Стандартные функции могут быть вещественного, логического и строкового типа. Аргументы заключаются в круглые скобки и перечисляются через запятую. Вместо чисел и литералов в качестве аргументов могут использоваться выражения соответствующего типа, вместо имен геометрических объектов могут использоваться вложенные определения.

Стандартные функции вещественного типа.

Имя	Функция	Аргументы	Результат
СИН	Вычисление синуса	<i>выражение</i>	Синус
КОС	Вычисление косинуса	<i>выражение</i>	Косинус
ТАН	Вычисление тангенса	<i>выражение</i>	Тангенс
АСИН	Вычисление арксинуса	<i>выражение</i>	Арксинус
АКОС	Вычисление арккосинуса	<i>выражение</i>	Арккосинус
АТАН	Вычисление арктангенса	<i>выражение</i>	Арктангенс
АТАН2	Вычисление арктангенса	<i>y – координата y точки x – координата x точки</i>	АТАН2 (y,x) = АТАН (y/x). Угол между осью x и линией, проведенной из начала координат (0, 0) в точку с координатами (x, y).
КВКОР	Вычисление квадратного корня	<i>выражение</i>	Квадратный корень
ЛОГ10	Вычисление десятичного логарифма	<i>выражение</i>	Десятичный логарифм
ЛОГ	Вычисление натурального логарифма	<i>выражение</i>	Натуральный логарифм
EXP	Вычисление экспоненты	<i>выражение</i>	Экспонента
АБС	Вычисление абсолютной величины	<i>выражение</i>	Абсолютное значение числа или вещественной переменной
ЦЕЛ	Выделение целой части	<i>выражение</i>	Целая часть числа или веществен. переменной

Имя	Функция	Аргументы	Результат
УГОЛПР	Вычисление угла между двумя прямыми	<i>прямая1,</i> <i>прямая2</i>	Угол между двумя прямыми
УГОЛВ	Вычисление угла между двумя векторами	<i>вектор1,</i> <i>вектор2</i>	Угол между двумя векторами
УГОЛ180	Приведение угла к диапазону $(-180^\circ, 180^\circ]$	<i>угол</i>	Угол в диапазоне $(-180^\circ, 180^\circ]$
УГОЛ360	Приведение угла к диапазону $[0^\circ, 360^\circ)$	<i>угол</i>	Угол в диапазоне $[0^\circ, 360^\circ)$
МОДВ	Вычисление модуля вектора	<i>вектор</i>	Модуль вектора
СКАЛПР	Вычисление скалярного произведения	<i>вектор1,</i> <i>вектор2</i>	Скалярное произведение векторов
РАСТ	Расстояние между двумя геометрическими объектами	<i>объект1,</i> <i>объект2</i>	Кратчайшее расстояние от одного геометрического объекта до другого
КООРД	Извлечение составляющей канонической формы	<i>номер,</i> <i>имя</i>	Составляющая с заданным номером канонической формы объекта с заданным именем
ИНФОРМ	Информация об объектах, траектории и др. справки	<i>номер/</i> <i>оператор</i>	1 – номер оператора 2 – номер строки текста 3 – уровень вложения макро 4 – номер сегмента контура 5 – номер участка траектории 10 – координата X текущей точки 11 – координата Y текущей точки 12 – координата Z

Имя	Функция	Аргументы	Результат
			<p>текущей точки</p> <p><i>ПРИПУСК</i> – величина припуска в мм (оператор ПРИПУСК)</p> <p><i>ГЛУБИНА</i> – глубина обработки (оператор ГЛУБИНА)</p> <p><i>ПОВЗАГ</i> – уровень поверхности заготовки (оператор ПОВЗАГ)</p> <p><i>БЕЗОПЛ</i> – плоскость безопасности (оператор БЕЗОПЛ)</p> <p><i>ИНСТР</i> - диаметр инструмента</p> <p><i>ПЕРЕКР</i> – перекрытие в мм (оператор ПЕРЕКР)</p> <p><i>НЕДОХОД</i> – уровень недохода в мм (оператор НЕДОХОД)</p> <p><i>ПЕРЕБЕГ</i> – величина перебега в мм (оператор ПЕРЕБЕГ)</p> <p><i>БЕЗОПРСТ</i> – безопасное расстояние (оператор БЕЗОПРСТ)</p>
ДЛИНА	Вычисление длины контура или участка траектории	<i>контур, элемент массива ХОД</i>	Длина контура или участка траектории
ДЛИНАСТР	Длина строки	<i>строка</i>	Количество символов
ЗНАК	Определение знака переменной	<i>выражение</i>	1, если выражение > 0

Имя	Функция	Аргументы	Результат
			-1, если выражение <0 0, если выражение =0
КОЛЭЛЕМ	Определяет количество элементов массива, для которых определены значения	<i>переменная (массив)</i>	Количество элементов массива, для которых определены значения.
	Определяет количество сегментов контура	<i>переменная (контур)</i>	Количество сегментов контура
ОКРУГЛ	Округление	<i>выражение – округляемое значение точность – точность округления (0.1 –десятые, 0.01 – сотые, 0.001 – тысячные и т. д)</i>	Округленное значение
ВЕЩ	Преобразование строки в число	<i>строка – строковое представление вещественного или целого числа</i>	Вещественное число
ПОИСКСТР	Ищет вхождение подстроки в строку	<i>строка - строка, в которой производится поиск, подстрока - искомая строка n - номер символа начала поиска (по умолчанию</i>	Номер символа (начиная с 1) строки вхождения подстроки или 0, если вхождение не найдено

Имя	Функция	Аргументы	Результат
		1)	

 Угол между векторами или прямыми отсчитывается от элемента, заданного в качестве первого аргумента.

Пример использования стандартных функций вещественного типа:

$$\text{РАССТОЯНИЕ}=\text{КВКОР}(\text{СИН}(\text{УГОЛ})^{**2}+\text{КОС}(\text{УГОЛ})^{**2})$$

Стандартные функции логического типа:

Имя	Функция	Аргументы	Результат
КОРР	Корректность	Вложенное геометрическое определение	ИСТИНА , если объект может быть построен; ЛОЖЬ -в противном случае
НЕКОРР	Некорректность	Вложенное геометрическое определение	ИСТИНА , если объект не может быть построен; ЛОЖЬ -в противном случае

Пример использования логической функции **КОРР**:

$$\text{ЛОГИЧ А} = \text{КОРР ТОЧКА ТЧ1 (ПЕРЕСЕЧ,ПР1,КР1,УБ)}$$

$$\text{ЕСЛИ (НЕ А) ТЧ1=ТЧ0}$$

Переменной **ТЧ1** присваивается значение одной из точек пересечения окружности и прямой -если они пересекаются; в противном случае переменной **ТЧ1** присваивается значение переменной **ТЧ0**.

Стандартные функции строкового типа.

Имя	Функция	Аргументы	Результат
ПОДСТР	Выделение подстроки	<i>строка,</i> <i>номер</i> <i>первого</i> <i>символа,</i> <i>длина</i>	Подстрока, заданной длины, выделенная из исходной, начиная с заданной позиции
СИМВОЛ	Символ	<i>код символа</i> <i>длина</i>	Строка, полученная повторением символа, имеющего заданный ASCII код. По умолчанию длина равна 1.

Имя	Функция	Аргументы	Результат
РЕГИСТ Р	Преобразование строки в верхний или нижний регистр	<i>строка</i> , { ВЕРХ/НИЖН }	Строка, полученная преобразованием исходной в верхний или нижний регистр
СТРОКА с()	Определение массива строк, полученного из строки с разделителями	<i>с=ПОИСК</i> , <i>строка</i> - текст, подлежащий разделению на фрагменты, <i>разделители</i> - строка, в которой перечислены символы разделителей для выделения фрагментов строки	Массив строк

Пример использования стандартных функций строкового типа:

СТРОКА С=ПОДСТР('авджик',3,5), ' ',СИМВОЛ(10,5), СИМВОЛ(23)

1.1.18 Оператор присваивания

Для присваивания значения переменным используется оператор присваивания.

Он состоит из левой части, знака равенства и правой части.

<i>переменная = определение</i>

В результате выполнения оператора присваивания значение правой части присваивается переменной, находящейся в левой части. Тип левой части должен допускать такое присваивание: если в левой части стоит переменная вещественного (логического) типа, то и в правой части должно стоять арифметическое (логическое) выражение.

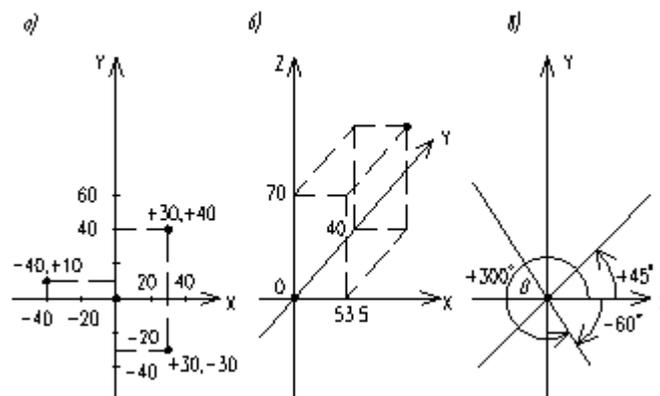
Например: оператор **X=X+1** -правильный оператор присваивания, в результате выполнения которого значение переменной **X** будет увеличено на 1. Оператор **A=ЛОЖЬ** является правильным оператором, только при условии, что до его использования определен логический тип переменной **A** оператором **ЛОГИЧ A**.

1.1.19 Система координат

В Техтране все геометрические объекты и траектория движения описываются в правосторонней прямоугольной системе координат.

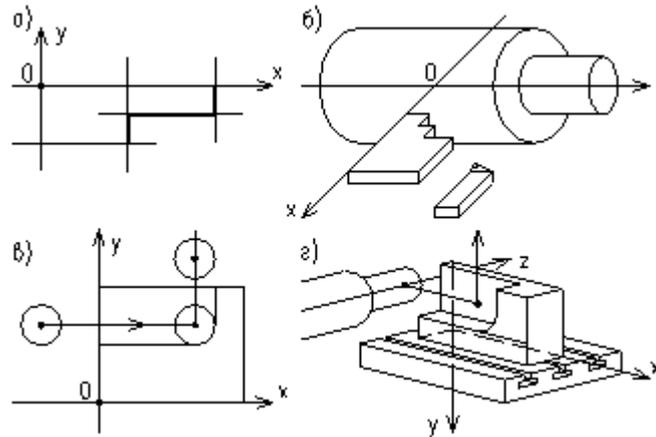


Следующий рисунок демонстрирует примеры отсчета координат на плоскости (а), в пространстве (б) и угловых размеров (в).



Определяемые прямые и окружности всегда находятся в горизонтальной плоскости xOy , поэтому описания геометрии, движения по контуру и поверхности детали не зависят от системы координат конкретного станка. Перемещения и координаты, рассчитанные процессором системы, приводятся в соответствие с фактическими направлениями и адресами координат станка при работе постпроцессора, формирующего УП.

На следующем рисунке приведены примеры перехода от системы координат детали к системе координат станка для токарных (а, б) и фрезерных (в, г) станков.



1.2 Геометрические определения

Геометрическое определение является оператором присваивания, в левой части которого находится символическое имя определяемого геометрического объекта, а в правой - один из вариантов задания этого объекта в соответствии с форматом, приведенным в данном разделе.

В общем виде геометрическое определение имеет следующий формат:

$$[тип] имя [(индекс)] = список параметров$$

Параметры:

Параметр	Описание
<i>тип</i>	Тип геометрического объекта: ТОЧКА, ПРЯМАЯ, ВЕКТОР, ОКРУЖН, ПЛОСК, МАТР, КОНТУР
<i>имя</i>	Символическое имя определяемого геометрического объекта.
<i>индекс</i>	Размерность массива.
<i>список параметров</i>	Параметры, определяющие геометрический объект: <ul style="list-style-type: none"> опорные элементы, заданные именем или вложенным геометрическим определением;

Параметр	Описание
	<ul style="list-style-type: none"> ▪ модификаторы; ▪ числа или заменяющие их арифметические выражения.

Служебное слово, определяющее тип геометрического объекта может отсутствовать, если тип объекта был указан ранее явно (отдельной строкой) или, если используется символическое имя, определяющее тип геометрического объекта неявно.

Если размерность массива опущена, то она будет определена по числу построенных объектов. Если количество построенных объектов, меньше размерности массива, то часть элементов массива остается неопределенной. Если массив не может вместить все объекты, т.е. размерность массива меньше числа объектов, то размерность массива переопределяется (становится равной фактическому числу построенных объектов).

Темы этого раздела:

- ▣ [Внутреннее представление геометрических объектов](#)^[33]
- ▣ [Определение точки](#)^[35]
- ▣ [Определение прямой](#)^[56]
- ▣ [Определение окружности](#)^[74]
- ▣ [Определение вектора](#)^[92]
- ▣ [Определение матрицы](#)^[99]
- ▣ [Определение плоскости](#)^[110]
- ▣ [Вложенное определение](#)^[115]
- ▣ [Определение массива точек](#)^[115]
- ▣ [Определение массива прямых](#)^[132]
- ▣ [Определение массива окружностей](#)^[139]
- ▣ [Определение массива контуров](#)^[142]

1.2.1 Внутреннее представление геометрических объектов

Для внутреннего представления геометрических объектов в Техтране используются канонические формы, которые представлены в следующей таблице:

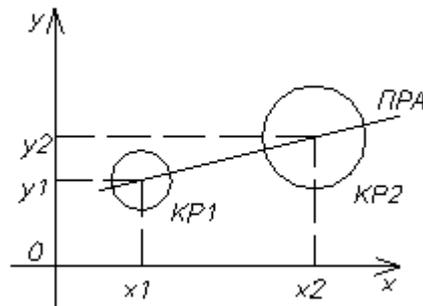
Тип объекта	Внутреннее представление	
<i>точка</i>	x, y, z	Координаты точки.
<i>прямая</i>	$cosa, sina, p$	Коэффициенты нормального уравнения прямой $xcosa + ysina - p = 0$.
<i>окружность</i>	x, y, R	Координаты центра окружности и ее радиус.
<i>вектор</i>	Dx, Dy, Dz	Проекции вектора на координатные оси.

Тип объекта	Внутреннее представление	
<i>плоскость</i>	$cosa, cosb, cosg, p$	Коэффициенты нормального уравнения плоскости $xcosa + ycosb + zcosg - p = 0$.
<i>матрица</i>	$a11, a12, \dots, a34$	12 коэффициентов нормальных уравнений координатных плоскостей системы координат, полученной при преобразовании исходной системы координат в соответствии с данной матрицей.

Внутреннее представление геометрических объектов используется при их контроле или при выводе на печать результатов интерпретации в случае использования отладочных операторов СЛЕД и ПЧТ1.

Для доступа к элементам канонической формы точек, прямых, окружностей и векторов следует использовать функцию **КООРД**.

Например, требуется определить прямую, проходящую через центры двух окружностей, координаты которых неизвестны, как показано на следующем рисунке.



Эти координаты определяются следующим образом:

$$X1 = \text{КООРД}(1, \text{КР1})$$

$$Y1 = \text{КООРД}(2, \text{КР1})$$

$$X2 = \text{КООРД}(1, \text{КР2})$$

$$Y2 = \text{КООРД}(2, \text{КР2})$$

Искомая прямая задается следующим геометрическим определением:

$$\text{ПРА} = X1, Y1, X2, Y2$$

1.2.2 Определение точки

В определениях точки используются следующие группы модификаторов.

- Модификаторы выбора точки из двух возможных:
 - **ХБ** – выбирается точка с большей координатой X;
 - **УБ** – выбирается точка с большей координатой Y;
 - **ХМ** – выбирается точка с меньшей координатой X;
 - **УМ** – выбирается точка с меньшей координатой Y.
- Модификаторы направления отсчета угла:
 - **ПОЧС** – задается отсчет угла по часовой стрелке;
 - **ПРЧС** – задается отсчет угла против часовой стрелки.
- Модификаторы, указывающие координату точки, заданную в геометрическом определении:
 - **ХКООРД** – координата X;
 - **УКООРД** – координата Y.

Темы этого раздела:

- [Точка, определенная прямоугольными координатами](#)^[36]
- [Точка, определенная пересечением двух прямых](#)^[36]
- [Точка, определенная пересечением прямой и окружности](#)^[37]
- [Точка, определенная пересечением двух окружностей](#)^[38]
- [Точка пересечения окружности и ее радиуса](#)^[39]
- [Точка, являющаяся центром окружности](#)^[40]
- [Точка окружности на известном расстоянии от другой точки](#)^[41]
- [Точка прямой, имеющая заданную координату](#)^[41]
- [Точка прямой на определенном расстоянии от другой точки этой же прямой](#)^[42]
- [Точка, заданная приращениями координат известной точки](#)^[43]
- [Точка на заданном расстоянии от данной точки под углом к оси X](#)^[44]
- [Точка, для которой ранее определенный вектор является радиус-вектором](#)^[45]
- [Точка, полученная в результате преобразования координат известной точки](#)^[45]
- [Точка, определенная ранее заданной точкой](#)^[46]
- [Точка пересечения прямой и контура, ближайшая к заданной точке](#)^[46]
- [Точка пересечения прямой и сегмента контура, заданного номером](#)^[47]
- [Точка пересечения окружности и контура, ближайшая к заданной точке](#)^[48]
- [Точка пересечения окружности и сегмента контура, заданного номером](#)^[48]
- [Точка пересечения сегментов двух контуров](#)^[49]
- [Опорная точка контура](#)^[50]
- [Текущая точка контура или траектории](#)^[51]
- [Точка контура, ближайшая к заданной точке](#)^[51]
- [Точка контура, ближайшая к другому контуру](#)^[52]
- [Точка прямой, ближайшая к заданной точке](#)^[53]
- [Точка окружности, ближайшая к заданной точке](#)^[54]
- [Задание](#)^[54] [координаты Z в геометрических определениях точек](#)^[54]

1.2.2.1 Точка, определенная прямоугольными координатами

Формат:

имя = координата X, координата Y [, координата Z]

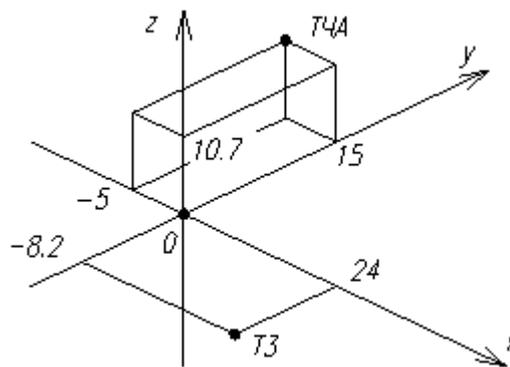
Параметры:

Параметр	Описание
<i>координата X</i>	Координата X.
<i>координата Y</i>	Координата Y.
<i>координата Z</i>	Координата Z.

Пример:

ТЧА = -5, 15, 10.7

ТОЧКА ТЗ = 24, -8.2



1.2.2.2 Точка, определенная пересечением двух прямых

Формат:

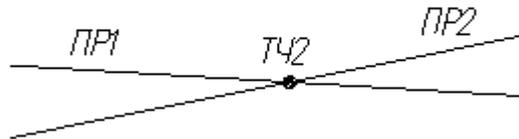
имя = [ПЕРЕСЕЧ,] прямая1, прямая2

Параметры:

Параметр	Описание
<i>прямая1</i>	Первая прямая.
<i>прямая2</i>	Вторая прямая.

Пример:

ТЧ2 = ПР1, ПР2



1.2.2.3 Точка, определенная пересечением прямой и окружности

Формат :

имя = [ПЕРЕСЕЧ,] *прямая, окружность*, $\left. \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}$

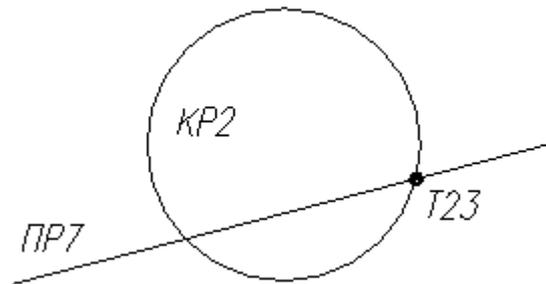
Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
<i>окружность</i>	Окружность.
ХБ	Точка с большей координатой X.
УБ	Точка с большей координатой Y.
ХМ	Точка с меньшей координатой X.
УМ	Точка с меньшей координатой Y.

 Касание прямой и окружности рассматривается как пересечение этих двух объектов. Модификатор в этом случае может быть любым.

Пример:

ТОЧКА Т23 = ПЕРЕСЕЧ, ПР7, КР2, УБ



1.2.2.4 Точка, определенная пересечением двух окружностей

Формат:

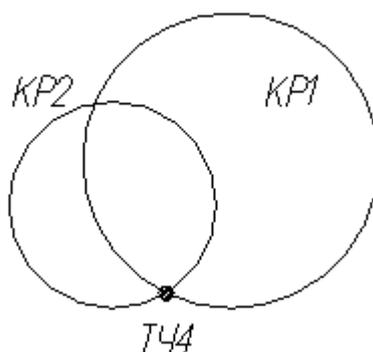
$$\text{имя} = [\text{ПЕРЕСЕЧ},] \text{окружность}_1, \text{окружность}_2, \left. \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>окружность1</i>	Первая окружность.
<i>окружность2</i>	Вторая окружность.
ХБ	Точка с большей координатой X.
УБ	Точка с большей координатой Y.
ХМ	Точка с меньшей координатой X.
УМ	Точка с меньшей координатой Y.

Пример:

ТЧ4 = КР1, КР2, УМ



1.2.2.5 Точка пересечения окружности и ее радиуса

Формат :

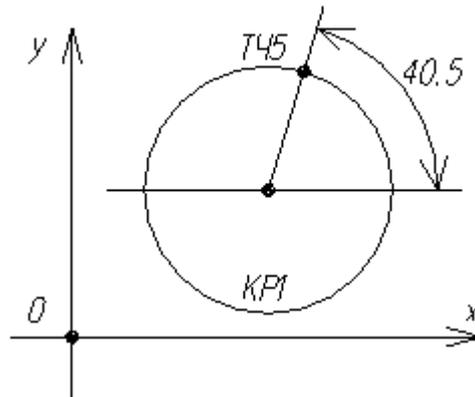
$$\text{имя} = \text{окружность}, \left\{ \begin{array}{l} \text{ПОЧС} \\ \text{ПРЧС} \end{array} \right\}, \text{угол}$$

Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
ПОЧС	Отсчет угла по часовой стрелке.
ПРЧС	Отсчет угла против часовой стрелки.
<i>угол</i>	Угол между радиусом окружности и осью X.

Пример:

TЧ5 = KP1, ПРЧС, 40.5



1.2.2.6 Точка, являющаяся центром окружности

Формат:

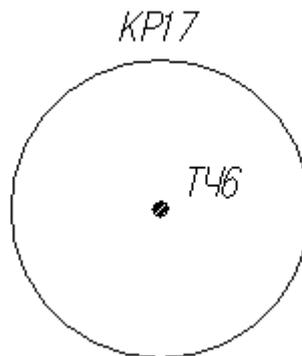
`имя = [ЦЕНТР,] окружность`

Параметры:

Параметр	Описание
<code>окружность</code>	Окружность.

Пример:

`ТЧ6 = КР17`



1.2.2.7 Точка окружности на известном расстоянии от другой точки

Формат:

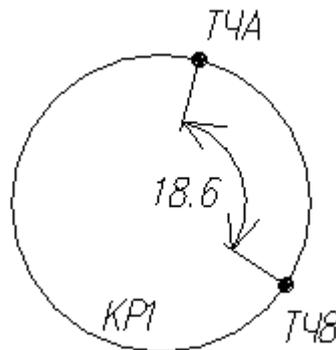
$$\text{имя} = \text{окружность}, \text{точка}, \left\{ \begin{array}{l} \text{ПРЧС} \\ \text{ПОЧС} \end{array} \right\}, \text{расстояние}$$

Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
<i>точка</i>	Точка.
ПОЧС	Отсчет угла по часовой стрелке.
ПРЧС	Отсчет угла против часовой стрелки.
<i>расстояние</i>	Расстояние между точками, измеряемое по дуге окружности.

Пример:

ТЧ8 = КР1, ТЧА, ПОЧС, 18.6



1.2.2.8 Точка прямой, имеющая заданную координату

Формат:

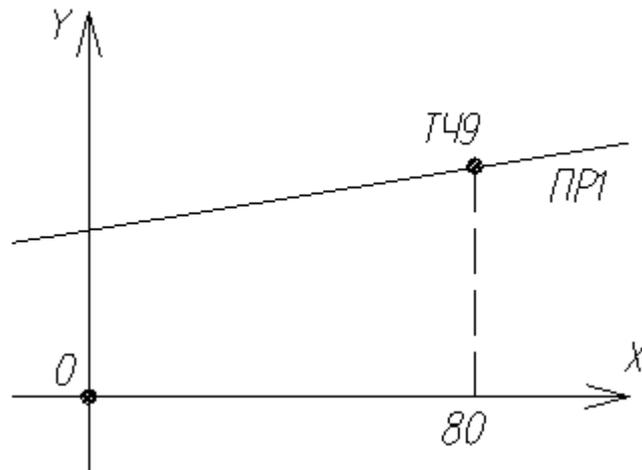
$$\text{имя} = \text{прямая}, \left\{ \begin{array}{l} \text{ХКООРД} \\ \text{УКООРД} \end{array} \right\}, \text{число}$$

Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
ХКООРД	Указание координаты X.
УКООРД	Указание координаты Y.
<i>число</i>	Значение координаты.

Пример:

ТЧ9 = ПР1, ХКООРД, 80



1.2.2.9 Точка прямой на определенном расстоянии от другой точки этой же прямой

Формат:

<i>имя = прямая, точка, расстояние,</i>	$\left. \begin{array}{l} \mathbf{ХБ} \\ \mathbf{УБ} \\ \mathbf{ХМ} \\ \mathbf{УМ} \end{array} \right\}$
-----------------------------------------	---------------------------------------------------------------------------------------------------------

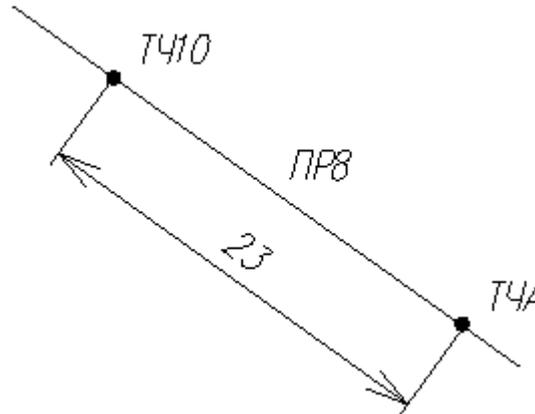
Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
<i>точка</i>	Точка.
<i>расстояние</i>	Расстояние между точками.
ХБ	Точка с большей координатой X.

Параметр	Описание
УБ	Точка с большей координатой Y.
ХМ	Точка с меньшей координатой X.
УМ	Точка с меньшей координатой Y.

Пример:

ТЧ10 = ПР8, ТЧА, 23, ХМ



1.2.2.10 Точка, заданная приращениями координат известной точки

Формат:

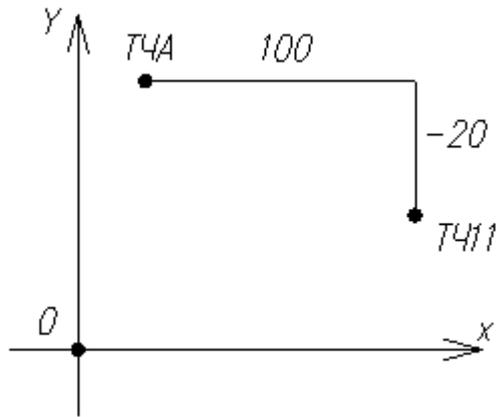
имя = точка, ПЕРЕНОС, приращение x, приращение y

Параметры:

Параметр	Описание
<i>точка</i>	Точка.
<i>приращение x</i>	Приращение по оси X
<i>приращение y</i>	Приращение по оси Y

Пример:

ТЧ11 = ТЧА, ПЕРЕНОС, 100, -20



1.2.2.11 Точка на заданном расстоянии от данной точки под углом к оси X

Формат:

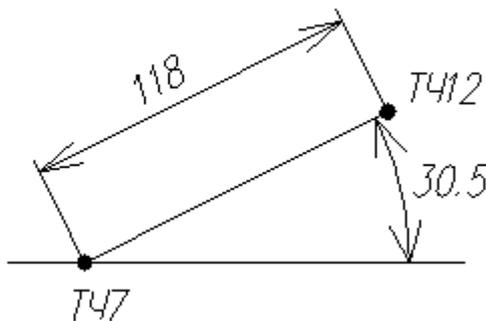
*имя = точка, { ПОВХУ
ПОВОРОТ }, угол, расстояние*

Параметры:

Параметр	Описание
<i>точка</i>	Точка.
<i>угол</i>	Угол между прямой, проходящей через точки, и осью X.
<i>расстояние</i>	Расстояние между точками.

Пример:

ТЧ12 = ТЧ7, ПОВХУ, 30.5, 118



1.2.2.12 Точка, для которой ранее определенный вектор является радиус-вектором

Формат:

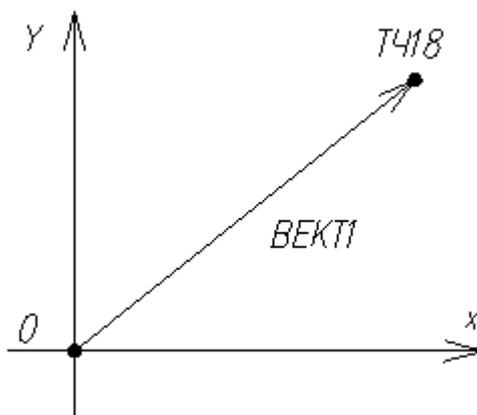
имя = вектор

Параметры:

Параметр	Описание
<i>вектор</i>	Вектор.

Пример:

ТЧ18 = ВЕКТ1



1.2.2.13 Точка, полученная в результате преобразования координат известной точки

Формат:

имя = точка, матрица

Параметры:

Параметр	Описание
<i>точка</i>	Точка.
<i>матрица</i>	Матрица преобразования.

Пример:

ТЧ16 = ТЧА, МК6

1.2.2.14 Точка, определенная ранее заданной точкой

Формат:

имя = *точка*

Параметры:

Параметр	Описание
<i>точка</i>	Точка.

Пример:

ТЧ17 = ТЧ13А

1.2.2.15 Точка пересечения прямой и контура, ближайшая к заданной точке

Формат:

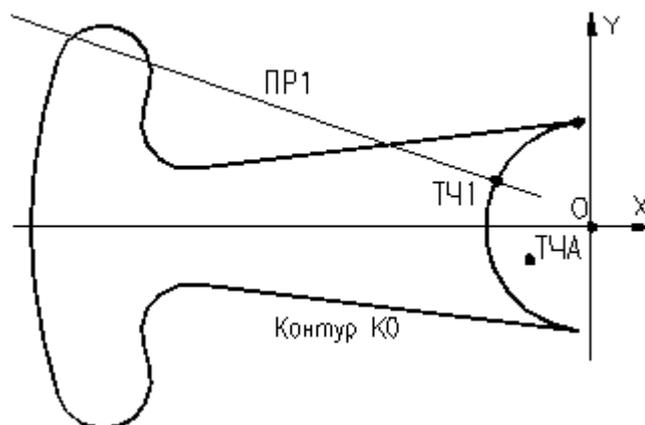
имя = [ПЕРЕСЕЧ,] *прямая*, *контур*, *точка*

Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
<i>контур</i>	Контур.
<i>точка</i>	Точка.

Пример:

ТЧ1 = ПР1,К0,ТЧА



1.2.2.16 Точка пересечения прямой и сегмента контура, заданного номером

Формат:

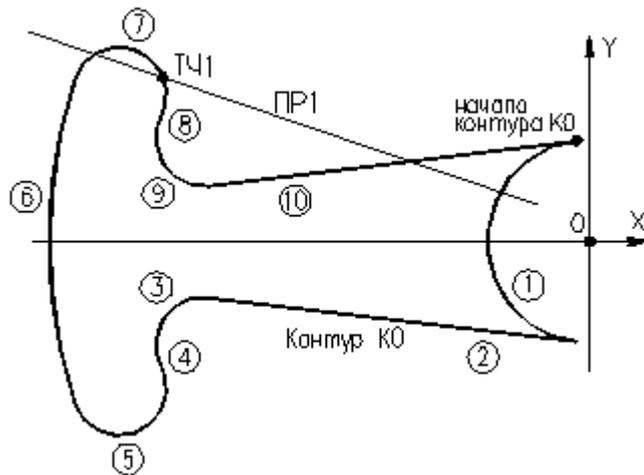
$$\text{имя} = [\text{ПЕРЕ СЕЧ,}] \text{ прямая, контур, число, } \begin{Bmatrix} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{Bmatrix}$$

Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
<i>контур</i>	Контур.
<i>число</i>	Номер сегмента контура.
ХБ	Точка с большей координатой X.
УБ	Точка с большей координатой Y.
ХМ	Точка с меньшей координатой X.
УМ	Точка с меньшей координатой Y.

Пример:

$$\text{ТЧ1} = \text{ПР1, К0, 7, ХБ}$$



1.2.2.17 Точка пересечения окружности и контура, ближайшая к заданной точке

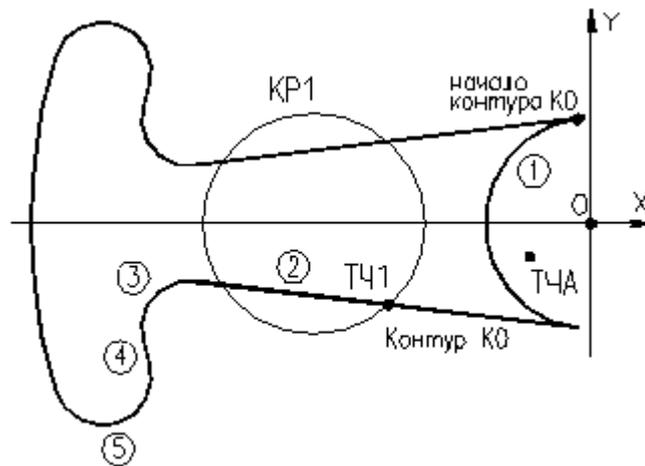
Формат:

$$\text{имя} = [\text{ПЕРЕСЕЧ,}] \text{окружность, контур, точка}$$

Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
<i>контур</i>	Контур.
<i>точка</i>	Точка.

Пример:

$$\text{ТЧ1} = \text{КР1, К0, ТЧА}$$


1.2.2.18 Точка пересечения окружности и сегмента контура, заданного номером

Формат:

$$\text{имя} = [\text{ПЕРЕСЕЧ,}] \text{окружность, контур, число, } \begin{Bmatrix} \text{XB} \\ \text{YB} \\ \text{XM} \\ \text{YM} \end{Bmatrix}$$

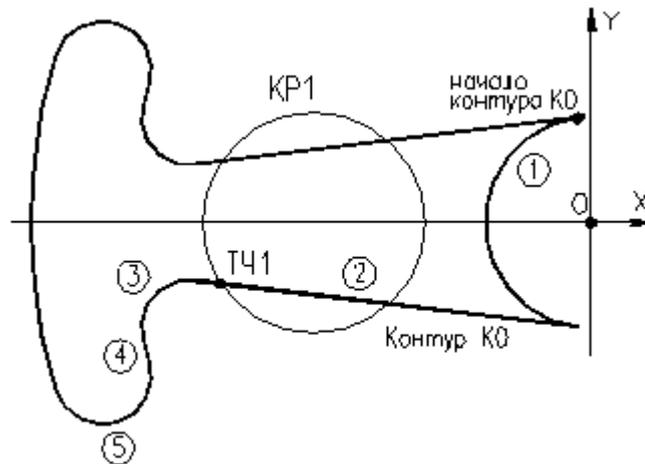
Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.

Параметр	Описание
<i>контур</i>	Контур.
<i>число</i>	Номер сегмента контура.
ХБ	Точка с большей координатой X.
УБ	Точка с большей координатой Y.
ХМ	Точка с меньшей координатой X.
УМ	Точка с меньшей координатой Y.

Пример:

ТЧ1 = КР1,К0,2,ХМ



1.2.2.19 Точка пересечения сегментов двух контуров

Формат:

$имя = [ПЕРЕСЕЧ,] контур1, число1, контур2, число2, \left. \begin{array}{l} ХБ \\ УБ \\ ХМ \\ УМ \end{array} \right\}$

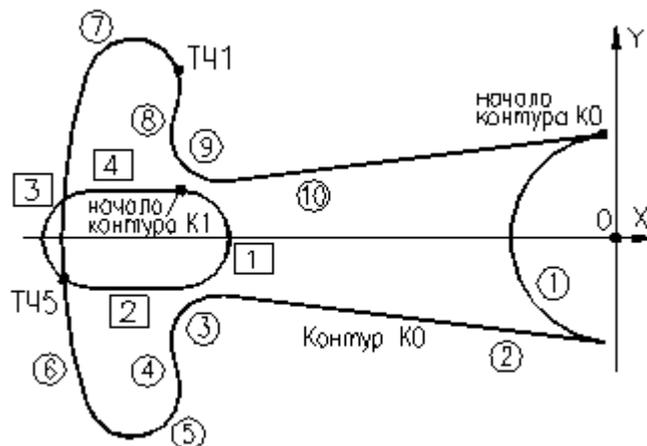
Параметры:

Параметр	Описание
<i>контур1</i>	Первый контур.

Параметр	Описание
<i>число1</i>	Номер сегмента первого контура.
<i>контур2</i>	Второй контур.
<i>число2</i>	Номер сегмента второго контура.
ХБ	Точка с большей координатой X.
УБ	Точка с большей координатой Y.
ХМ	Точка с меньшей координатой X.
УМ	Точка с меньшей координатой Y.

Пример:

ТЧ5 = ПЕРЕСЕЧ,К0,6,К1,3,УМ



1.2.2.20 Опорная точка контура

Формат:

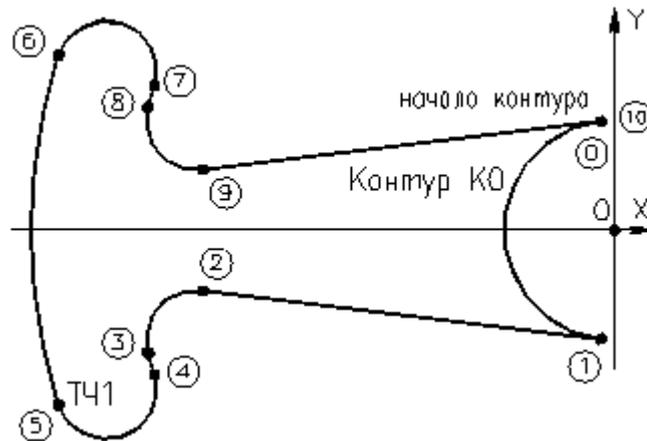
имя = контур, число

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>число</i>	Номер сегмента контура.

Пример:

ТЧ1 = К0,5



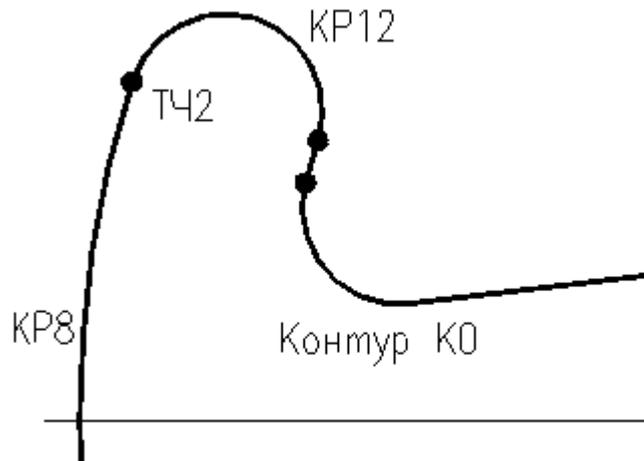
1.2.2.21 Текущая точка контура или траектории

Формат:

```
имя = ТЕКУЩ
```

Пример:

```
КОНТУР К0 = ИЗ.....
.....
ВПЕРЕД КР8 КАС КР12
ТЧ2 = ТЕКУЩ
```



1.2.2.22 Точка контура, ближайшая к заданной точке

Формат:

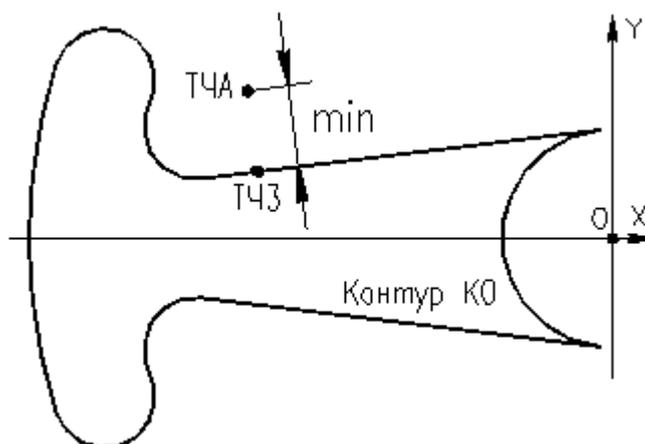
```
имя = контур, точка
```

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>точка</i>	Точка.

Пример:

ТЧЗ = К0,ТЧА



1.2.2.23 Точка контура, ближайшая к другому контуру

Формат:

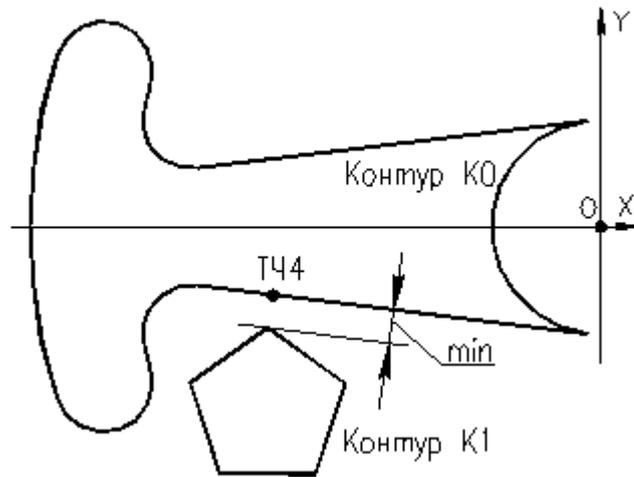
имя = контур1, контур2

Параметры:

Параметр	Описание
<i>контур1</i>	Первый контур.
<i>контур2</i>	Второй контур.

Пример:

ТЧ4 = К0,К1



1.2.2.24 Точка прямой, ближайшая к заданной точке

Формат:

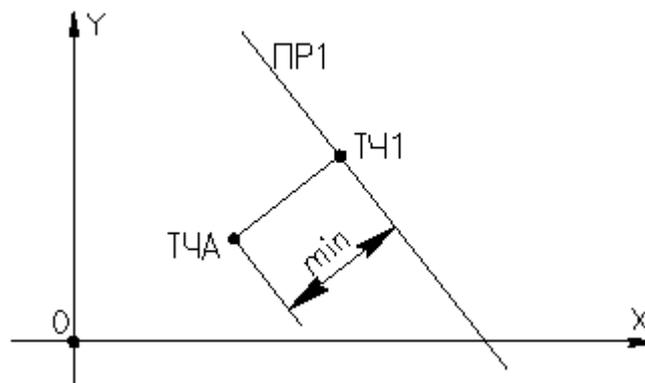
имя = прямая, точка

Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
<i>точка</i>	Точка.

Пример:

ТЧ1 = ПР1,ТЧА



1.2.2.25 Точка окружности, ближайшая к заданной точке

Формат:

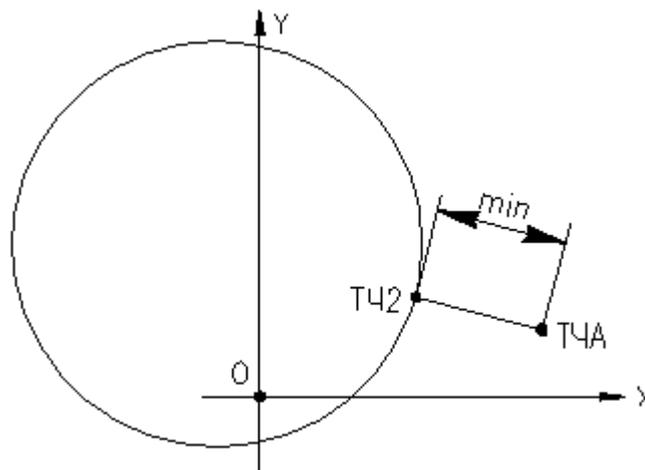
имя = окружность, точка

Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
<i>точка</i>	Точка.

Пример:

ТЧ2 = КР1, ТЧА



1.2.2.26 Задание координаты Z в геометрических определениях точек

Оператор ПЛЗ предназначен для задания координаты Z тех точек, в геометрических определениях которых нет явно заданной третьей координаты.

Формат:

ПЛЗ $\left\{ \begin{array}{l} \text{число} \\ \text{имя} \\ \text{ОТМЕН} \end{array} \right\}$

Параметры:

Параметр	Описание
<i>число</i>	Величина координаты Z.

Параметр	Описание
<i>имя</i>	Имя плоскости, определенной ранее.
ОТМЕН	Служебное слово, обозначающее отмену действия оператора ПЛЗ.

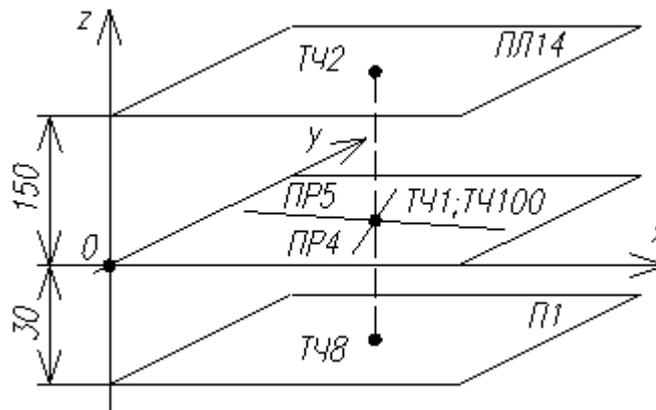
Действие оператора распространяется на точки, в геометрических определениях которых координата **Z** не задана явно. Для таких точек третья координата (**Z**) приравнивается числу, заданному в операторе **ПЛЗ**, или вычисляется исходя из того условия, что определяемая точка должна лежать в плоскости, заданной именем в операторе **ПЛЗ**.

Действие оператора **ПЛЗ** распространяется на все точки, определенные после него, до использования другого оператора **ПЛЗ** или оператора **ПЛЗ ОТМЕН**.

Если оператор **ПЛЗ** не был задан, или был выполнен оператор **ПЛЗ ОТМЕН**, то все координаты **Z** точек, не заданные явно, принимают нулевые значения.

☞ координата **Z** точки, в геометрическом определении которой в качестве опорного элемента используется другая точка, приравнивается координате **Z** опорной точки.

Следующий пример иллюстрирует управление координатой **Z** при построении точек.



ТЧ1 = ПР5, ПР4
ПЛОСК П1, ПЛ14
ПЛ14 = 150
П1 = -30
ПЛЗ ПЛ14
ТЧ2 = ПР5, ПР4
ПЛЗ ПЛ1
ТЧ8 = ПР5, ПР4
ПЛЗ ОТМЕН
ТЧ100 = ПР5, ПР4

Координаты **Z** точек **ТЧ2** и **ТЧ8** равны соответственно 150 и -30, так как они заданы оператором **ПЛЗ** с указанием плоскости, в которой лежит каждая из этих точек.

Координата **Z** точки **ТЧ1** по умолчанию равна нулю; координата точки **ТЧ100** также равна нулю, так как перед ее геометрическим определением задан оператор **ПЛЗ ОТМЕН**,

отменяющий действие последнего оператора ПЛЗ.

1.2.3 Определение прямой

В определениях прямой используются следующие группы модификаторов.

- Модификаторы, выбирающие одну прямую из двух:
 - **ХБ** – выбирается прямая в сторону большей координаты **X**;
 - **УБ** – выбирается прямая в сторону большей координаты **Y**;
 - **ХМ** – выбирается прямая в сторону меньшей координаты **X**;
 - **УМ** – выбирается прямая в сторону меньшей координаты **Y**.
- Модификаторы, указывающие положение точки касания прямой и опорной окружности:
 - **СПРАВА** – точка касания находится справа от центра окружности, если смотреть в направлении от первого опорного элемента ко второму;
 - **СЛЕВА** – точка касания находится слева от центра окружности, если смотреть в направлении от первого опорного элемента ко второму.
- Модификаторы, указывающие положение определяемой прямой относительно опорной:
 - **ПАРЛЕЛ** – параллельно;
 - **ПЕРП** – перпендикулярно.

Темы этого раздела:

- [Прямая, совпадающая с ранее определенной прямой](#)^[57]
- [Прямая, проходящая через две точки, заданные координатами](#)^[57]
- [Прямая, проходящая через две точки](#)^[58]
- [Прямая, проходящая через точку и касающаяся окружности](#)^[59]
- [Прямая, касающаяся двух окружностей](#)^[59]
- [Прямая, проходящая через точку под углом к оси X](#)^[60]
- [Прямая, проходящая через точку под углом к другой прямой](#)^[61]
- [Прямая, проходящая через точку и перпендикулярная к другой прямой](#)^[62]
- [Прямая, проходящая через точку и параллельная другой прямой](#)^[63]
- [Прямая параллельная другой прямой на заданном расстоянии](#)^[63]
- [Прямая параллельная другой прямой и касающаяся окружности](#)^[64]
- [Прямая, касающаяся окружности под заданным углом к оси X](#)^[65]
- [Прямая, проходящая через точку параллельно вектору](#)^[66]
- [Прямая, параллельная одной из осей на заданном расстоянии](#)^[67]
- [Прямая, параллельная одной из осей и имеющая заданную координату](#)^[68]
- [Прямая, полученная в результате преобразования координат](#)^[68]
- [Прямая, заданная отрезком контура](#)^[69]
- [Прямая, проходящая через точку и касающаяся контура в точке, ближайшей к заданной](#)^[69]
- [Прямая, проходящая через точку и касающаяся контура](#)^[70]
- [Прямая, проходящая через точку и перпендикулярная к контуру в точке, ближайшей к заданной](#)^[71]
- [Прямая, проходящая через точку и перпендикулярная к контуру](#)^[72]
- [Прямая, ограничивающая контур и параллельная прямой](#)^[72]

- [Прямая, ограничивающая контур и параллельная оси координат](#) ⁷³

1.2.3.1 Прямая, совпадающая с ранее определенной прямой

Формат:

```
имя = прямая
```

Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.

Пример:

ПР13 = ПР376

1.2.3.2 Прямая, проходящая через две точки, заданные координатами

Формат:

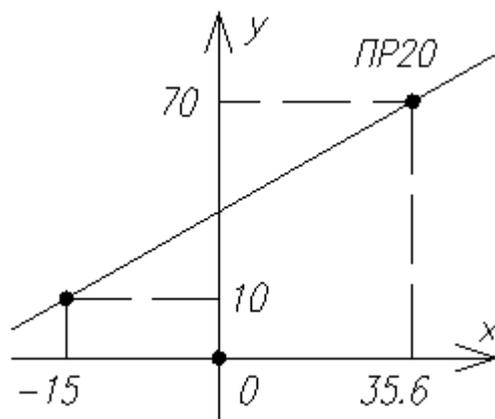
```
имя =  $x_1, y_1, x_2, y_2$ 
```

Параметры:

Параметр	Описание
x_1 y_1	Координаты первой точки.
x_2 y_2	Координаты второй точки.

Пример:

ПР20 = -15, 10, 35.6, 70



1.2.3.3 Прямая, проходящая через две точки

Формат:

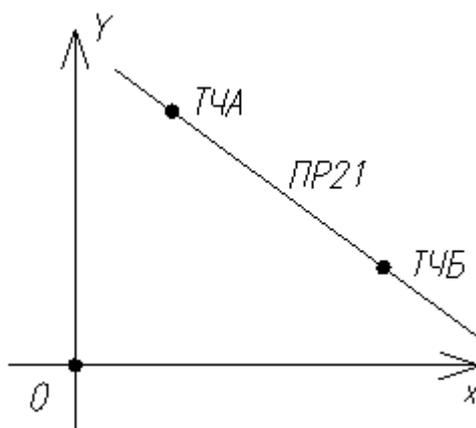
имя = точка1, точка2

Параметры:

Параметр	Описание
<i>точка1</i>	Первая точка.
<i>точка2</i>	Вторая точка.

Пример:

ПР21 = ТЧА, ТЧБ



1.2.3.4 Прямая, проходящая через точку и касающаяся окружности

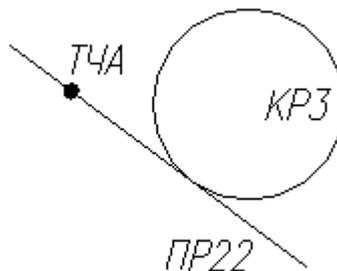
Формат:

$$\text{имя} = \text{точка}, \left\{ \begin{array}{l} \text{СЛЕВА} \\ \text{СПРАВА} \end{array} \right\} [, \text{КАСАТ}], \text{окружность}$$

Параметры:

Параметр	Описание
<i>точка</i>	Точка.
СЛЕВА	Точка касания находится слева от центра окружности, если смотреть в направлении от точки к центру окружности.
СПРАВА	Точка касания находится справа от центра окружности, если смотреть в направлении от точки к центру окружности.
<i>окружность</i>	Окружность.

Пример:

$$\text{ПР22} = \text{ТЧА}, \text{СПРАВА}, \text{КР3}$$


1.2.3.5 Прямая, касающаяся двух окружностей

Формат:

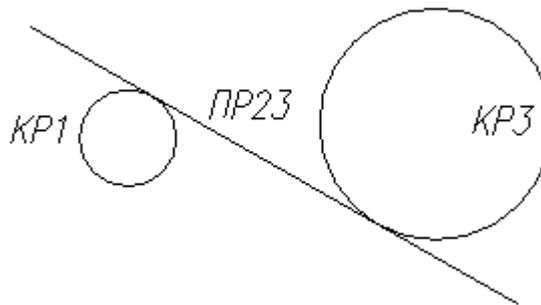
$$\text{имя} = \left\{ \begin{array}{l} \text{СЛЕВА} \\ \text{СПРАВА} \end{array} \right\} [, \text{КАСАТ}], \text{окружность } 1, \left\{ \begin{array}{l} \text{СЛЕВА} \\ \text{СПРАВА} \end{array} \right\} [, \text{КАСАТ}], \text{окружность } 2$$

Параметры:

Параметр	Описание
СЛЕВА	Точка касания находится слева от центра первой окружности, если смотреть в направлении от первой окружности ко второй.
СПРАВА	Точка касания находится справа от центра первой окружности, если смотреть в направлении от первой окружности ко второй.
<i>окружн₁</i>	Первая окружность.
СЛЕВА	Точка касания находится слева от центра второй окружности, если смотреть в направлении от первой окружности ко второй.
СПРАВА	Точка касания находится справа от центра второй окружности, если смотреть в направлении от первой окружности ко второй.
<i>окружн₂</i>	Вторая окружность.

Пример:

ПР23 = СЛЕВА, КР1, СПРАВА, КР3



1.2.3.6 Прямая, проходящая через точку под углом к оси X

Формат:

имя = точка, угол

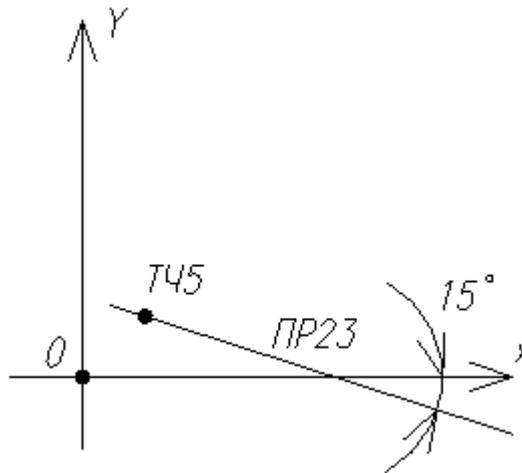
Параметры:

Параметр	Описание
<i>точка</i>	Точка.

Параметр	Описание
<i>угол</i>	Угол между прямой и осью X в градусах.

Пример:

ПР23 = ТЧ5,-15



1.2.3.7 Прямая, проходящая через точку под углом к другой прямой

Формат:

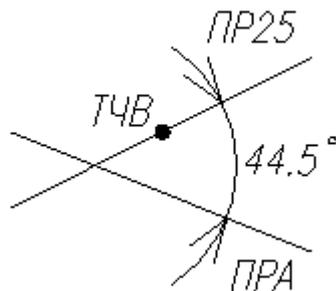
имя = прямая, точка, { ПОВОРОТ } , угол

Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
<i>точка</i>	Точка.
<i>угол</i>	Угол между прямыми в градусах.

Пример:

ПР25 = ПРА, ТЧВ, ПОВОРОТ, 44.5



1.2.3.8 Прямая, проходящая через точку и перпендикулярная к другой прямой

Формат:

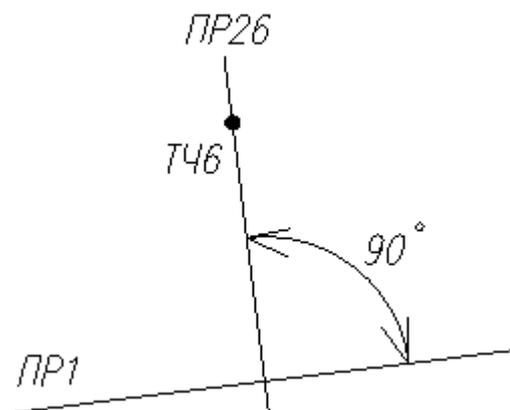
имя = точка, ПЕРП, прямая

Параметры:

Параметр	Описание
<i>точка</i>	Точка.
<i>прямая</i>	Прямая.

Пример:

PR26 = T46, ПЕРП, PR1



1.2.3.9 Прямая, проходящая через точку и параллельная другой прямой

Формат:

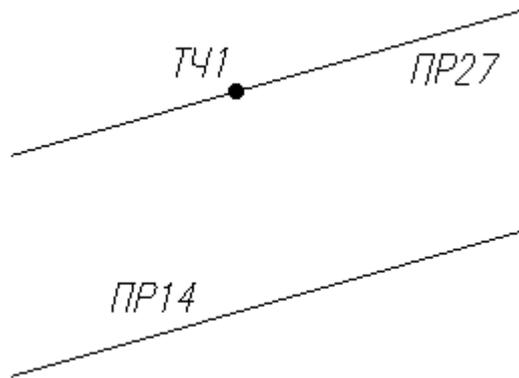
имя = *точка*, ПАРЛЕЛ, *прямая*

Параметры:

Параметр	Описание
<i>точка</i>	Точка.
<i>прямая</i>	Прямая.

Пример:

ПР27 = ТЧ1, ПАРЛЕЛ, ПР14



1.2.3.10 Прямая параллельная другой прямой на заданном расстоянии

Формат:

имя = [ПАРЛЕЛ,] *прямая*, *расстояние*, $\left. \begin{matrix} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{matrix} \right\}$

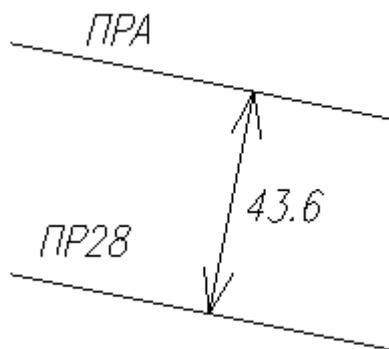
Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
<i>расстояние</i>	Расстояние между прямыми.

Параметр	Описание
ХБ	Смещение прямой в сторону большей координаты X.
УБ	Смещение прямой в сторону большей координаты Y.
ХМ	Смещение прямой в сторону меньшей координаты X.
УМ	Смещение прямой в сторону меньшей координаты Y.

Пример:

ПР28 = ПРА, 43.6, УМ



1.2.3.11 Прямая параллельная другой прямой и касающаяся окружности

Формат:

$\text{имя} = \text{КАС}, \text{окружность}, \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \text{прямая}$

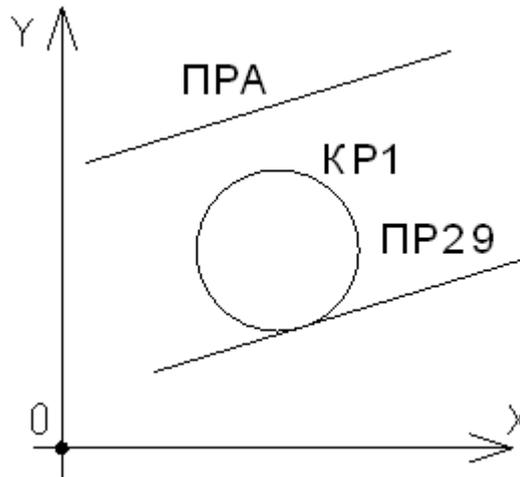
Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
<i>прямая</i>	Прямая.
ХБ	Смещение прямой в сторону большей координаты X.
УБ	Смещение прямой в сторону большей координаты Y.

Параметр	Описание
ХМ	Смещение прямой в сторону меньшей координаты X.
УМ	Смещение прямой в сторону меньшей координаты Y.

Пример:

ПР29 = КАС, КР1, УМ, ПРА



1.2.3.12 Прямая, касающаяся окружности под заданным углом к оси X

Формат:

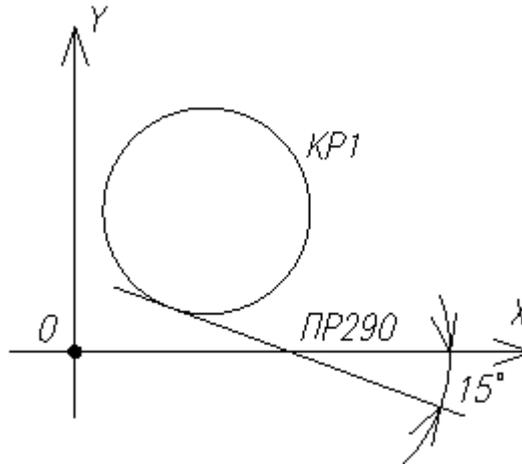
$$\text{имя} = [\text{КАС},] \text{окружность}, \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \text{угол}$$

Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
ХБ	Точка касания имеет большую координату X.
УБ	Точка касания имеет большую координату Y.
ХМ	Точка касания имеет меньшую координату X.
УМ	Точка касания имеет меньшую координату Y.
<i>угол</i>	Угол между прямой и осью X в градусах.

Пример:

ПР290 = КАС, КР1, УМ,-15



1.2.3.13 Прямая, проходящая через точку параллельно вектору

Формат:

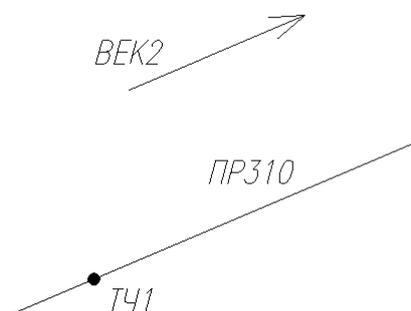
имя = точка, вектор

Параметры:

Параметр	Описание
<i>точка</i>	Точка.
<i>вектор</i>	Вектор.

Пример:

ПР310 = ТЧ1, ВЕК2



1.2.3.14 Прямая, параллельная одной из осей на заданном расстоянии

Формат:

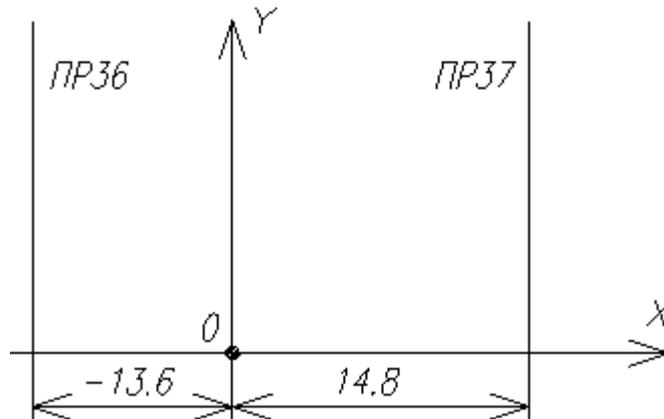
$$\text{имя} = \left\{ \begin{matrix} \text{ХПАР} \\ \text{УПАР} \end{matrix} \right\}, \left\{ \begin{matrix} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{matrix} \right\}, \text{расстояние}$$

Параметры:

Параметр	Описание
ХПАР	Ось координат X.
УПАР	Ось координат Y.
ХБ	Смещение прямой в сторону большей координаты X.
УБ	Смещение прямой в сторону большей координаты Y.
ХМ	Смещение прямой в сторону меньшей координаты X.
УМ	Смещение прямой в сторону меньшей координаты Y.
расстояние	Расстояние от оси координат.

Пример:

ПР37 = УПАР, ХБ, 14.8



1.2.3.15 Прямая, параллельная одной из осей и имеющая заданную координату

Формат:

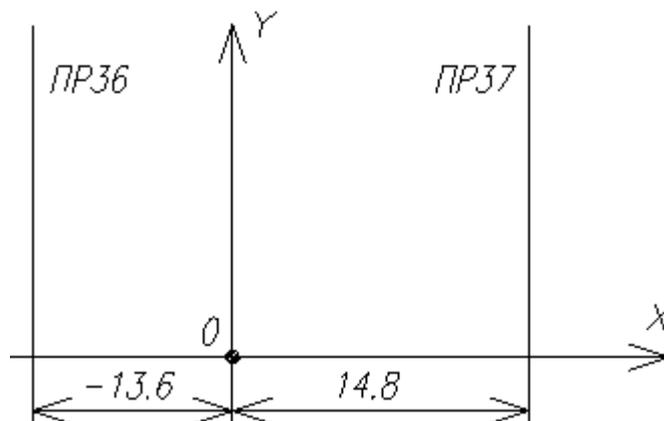
$$\text{имя} = \left\{ \begin{array}{l} \text{ХПАР} \\ \text{УПАР} \end{array} \right\}, \text{координата}$$

Параметры:

Параметр	Описание
<i>координата</i>	Заданная координата.
ХПАР	Ось координат X.
УПАР	Ось координат Y.

Пример:

$$\text{ПР36} = \text{УПАР}, -13.6$$



1.2.3.16 Прямая, полученная в результате преобразования координат

Формат:

$$\text{имя} = \text{прямая}, \text{матрица}$$

Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
<i>матрица</i>	Матрица преобразования координат.

Пример:

ПР32 = ПР10, МАТР1

1.2.3.17 Прямая, заданная отрезком контура

Формат:

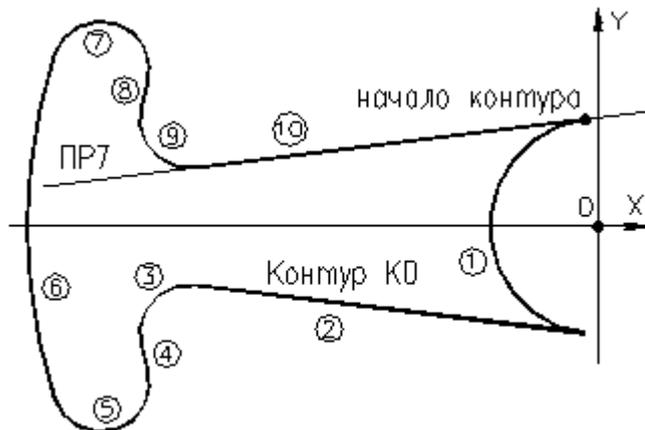
имя = контур, число

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>число</i>	Номер сегмента контура.

Пример:

ПР7 = К0,10



1.2.3.18 Прямая, проходящая через точку и касающаяся контура в точке, ближайшей к заданной

Формат:

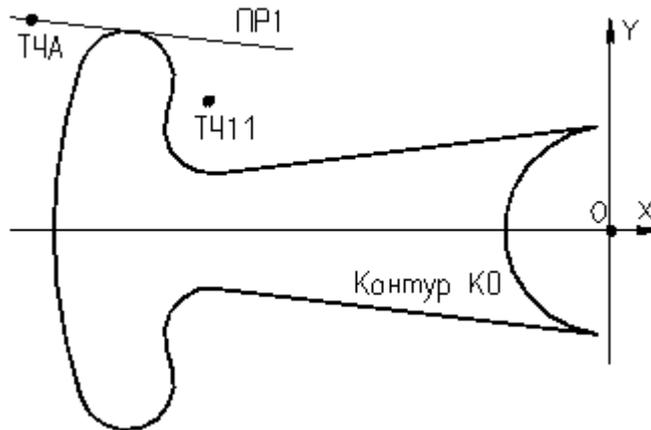
имя = точка1, КАС, контур, точка2

Параметры:

Параметр	Описание
<i>точка1</i>	Точка, через которую проходит прямая.
<i>контур</i>	Контур.
<i>точка2</i>	Точка, ближайшая к прямой.

Пример:

ПР1 = ТЧА,КАС,К0,ТЧ11



1.2.3.19 Прямая, проходящая через точку и касающаяся контура

Формат:

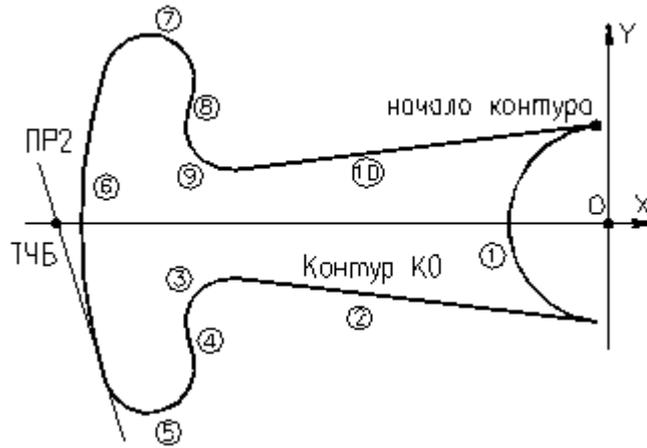
имя = *точка*, **КАС**, *контур*, *номер*, $\left. \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}$

Параметры:

Параметр	Описание
<i>точка</i>	Точка, через которую проходит прямая.
<i>контур</i>	Контур.
<i>номер</i>	Номер сегмента контура.
ХБ	Точка касания имеет большую координату X .
УБ	Точка касания имеет большую координату Y .
ХМ	Точка касания имеет меньшую координату X .
УМ	Точка касания имеет меньшую координату Y .

Пример:

ПР2 = ТЧБ,КАС,К0,6,УМ



1.2.3.20 Прямая, проходящая через точку и перпендикулярная к контуру в точке, ближайшей к заданной

Формат:

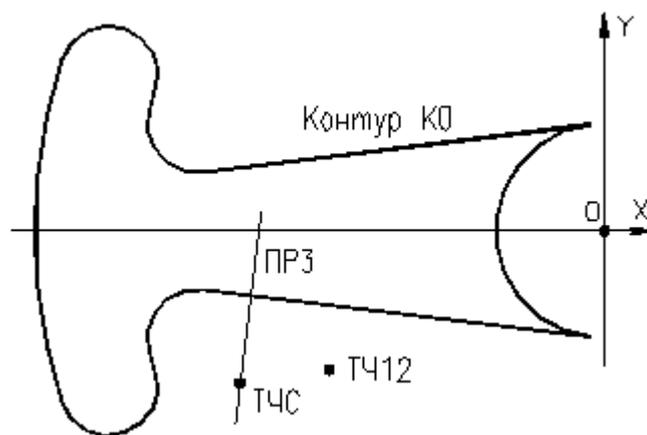
```
имя = точка1, ПЕРП, контур, точка2
```

Параметры:

Параметр	Описание
точка1	Точка, через которую проходит прямая.
контур	Контур.
точка2	Точка, ближайшая к прямой.

Пример:

ПР3 = ТЧС, ПЕРП, К0, ТЧ12



1.2.3.21 Прямая, проходящая через точку и перпендикулярная к контуру

Формат:

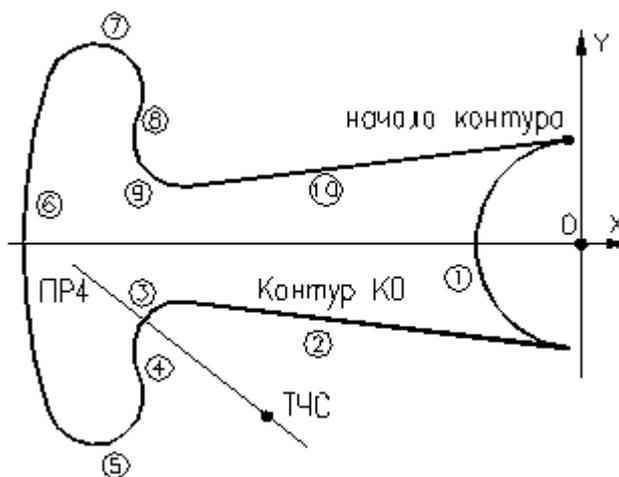
имя = точка, ПЕРП, контур, номер

Параметры:

Параметр	Описание
<i>точка</i>	Точка, через которую проходит прямая.
<i>контур</i>	Контур.
<i>номер</i>	Номер сегмента контура.

Пример:

ПР4 = ТЧС,ПЕРП,К0,3



1.2.3.22 Прямая, ограничивающая контур и параллельная прямой

Формат:

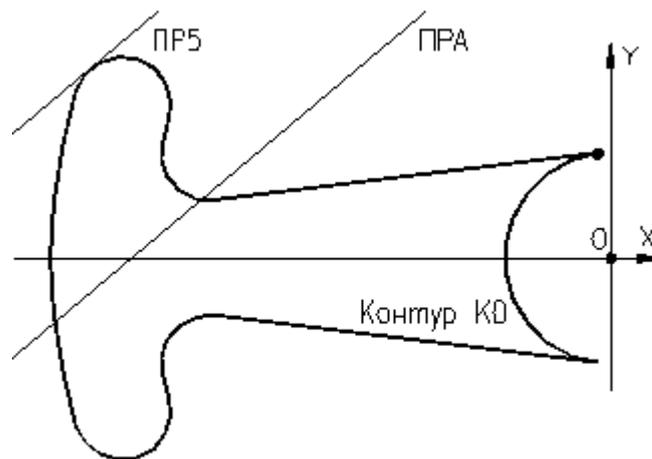
имя = контур, прямая, $\left\{ \begin{array}{l} \text{XB} \\ \text{УБ} \\ \text{XM} \\ \text{УМ} \end{array} \right\}$

Параметры:

Параметр	Описание
<i>контур</i>	Конгур.
<i>прямая</i>	Прямая.
ХБ	Смещение прямой в сторону большей координаты X.
УБ	Смещение прямой в сторону большей координаты Y.
ХМ	Смещение прямой в сторону меньшей координаты X.
УМ	Смещение прямой в сторону меньшей координаты Y.

Пример:

ПР5 = К0,ПРА,ХМ



1.2.3.23 Прямая, ограничивающая контур и параллельная оси координат

Формат:

$$ИМЯ = \left\{ \begin{matrix} \mathbf{ХПАР} \\ \mathbf{УПАР} \end{matrix} \right\}, \text{КОНТУР}, \left\{ \begin{matrix} \mathbf{ХБ} \\ \mathbf{УБ} \\ \mathbf{ХМ} \\ \mathbf{УМ} \end{matrix} \right\}$$

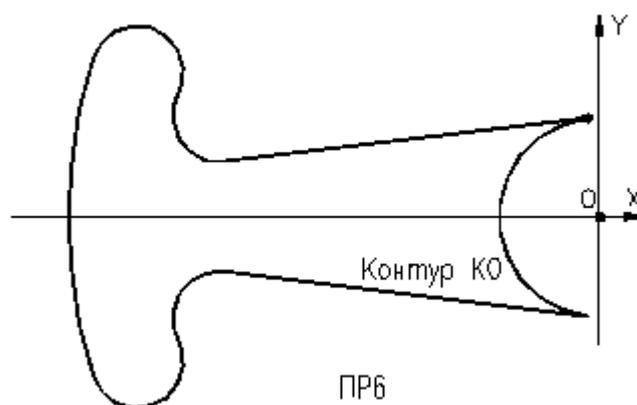
Параметры:

Параметр	Описание
ХПАР	Ось координат X.
УПАР	Ось координат Y.

Параметр	Описание
<i>контур</i>	Контур.
ХБ	Смещение прямой в сторону большей координаты X.
УБ	Смещение прямой в сторону большей координаты Y.
ХМ	Смещение прямой в сторону меньшей координаты X.
УМ	Смещение прямой в сторону меньшей координаты Y.

Пример:

ПР6 = ХПАР, К0, ХМ



1.2.4 Определение окружности

В определениях окружности используются следующие группы модификаторов.

- Модификаторы **БОЛШ** и **МЕНШ**, служащие для выбора из двух окружностей одной - с большим или меньшим радиусом соответственно;
- Модификаторы, указывающие положение определяемой окружности относительно другой, являющейся опорным элементом:
 - **ВНЕ** – окружность находится вне заданной окружности;
 - **ВНУТРИ** – окружность находится внутри заданной окружности.
- Модификаторы, указывающие положение определяемой окружности относительно прямой:
 - **ХМ** – центр окружности находится слева от прямой;
 - **ХБ** – справа от прямой;
 - **УБ** – выше прямой;
 - **УМ** – ниже прямой.
- Те же модификаторы **ХМ, ХБ, УМ, УБ**, но служащие для выбора из двух окружностей одной по соответствующей координате ее центра.

Модификаторы 2-ой и 3-ей групп относятся к следующим непосредственно за ними опорным элементам геометрического определения. Применение модификаторов **ВНУТРИ** и **ВНЕ** имеет особенности, отмеченные в примечаниях к конкретным геометрическим определениям.

Темы этого раздела:

- [Окружность, совпадающая с ранее определенной окружностью](#)^[75]
- [Окружность, определенная координатами центра и радиусом](#)^[76]
- [Окружность, определенная радиусом и точкой, являющейся ее центром](#)^[76]
- [Окружность, определенная точкой, являющейся ее центром, и касающаяся прямой](#)^[77]
- [Окружность, определенная точкой, являющейся ее центром, и проходящая через другую точку](#)^[78]
- [Окружность, проходящая через три точки](#)^[78]
- [Окружность, определенная точкой, являющейся ее центром, и касающаяся другой окружности](#)^[79]
- [Окружность заданного радиуса, касающаяся двух пересекающихся прямых](#)^[80]
- [Окружность заданного радиуса, касающаяся прямой и проходящая через точку](#)^[81]
- [Окружность заданного радиуса, проходящая через точку с центром, лежащим на прямой](#)^[82]
- [Окружность заданного радиуса, касающаяся прямой и другой окружности](#)^[83]
- [Окружность заданного радиуса, касающаяся двух других окружностей](#)^[85]
- [Окружность, касающаяся трех прямых](#)^[87]
- [Окружность заданного радиуса, проходящая через две точки](#)^[88]
- [Окружность заданного радиуса, проходящая через точку и касающаяся другой окружности](#)^[89]
- [Окружность, концентричная заданной окружности](#)^[90]
- [Окружность, заданная дугой, являющейся сегментом контура](#)^[91]
- [Окружность, полученная в результате преобразования координат](#)^[92]

1.2.4.1 Окружность, совпадающая с ранее определенной окружностью

Формат:

имя = окружность

Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.

Пример:

КР44 = КР100

1.2.4.2 Окружность, определенная координатами центра и радиусом

Формат:

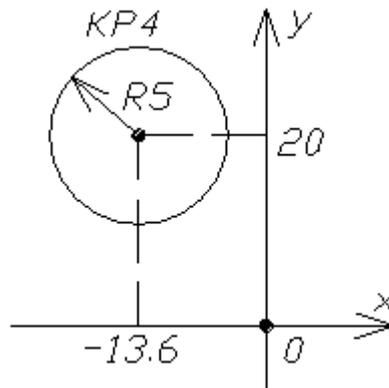
имя = [ЦЕНТР,] *координата x*, *координата y*, [РАДИУС,] *радиус*

Параметры:

Параметр	Описание
<i>координата x</i>	Координата X центра окружности.
<i>координата y</i>	Координата Y центра окружности.
<i>радиус</i>	Радиус окружности.

Пример:

КР4 = -13.6, 20, 5



1.2.4.3 Окружность, определенная радиусом и точкой центра

Формат:

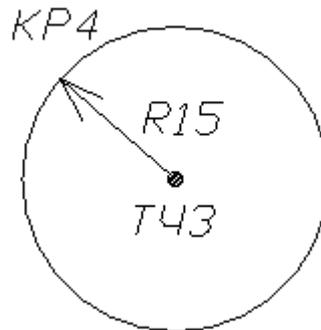
имя = [ЦЕНТР,] *точка*, [РАДИУС,] *радиус*

Параметры:

Параметр	Описание
<i>точка</i>	Центр окружности.
<i>радиус</i>	Радиус окружности.

Пример:

КР4 = ТЧ3, 15



1.2.4.4 Окружность, определенная точкой центра, и касающаяся прямой

Формат:

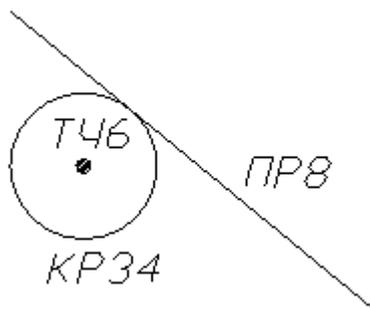
`имя = [ЦЕНТР,] точка, [КАС,] прямая`

Параметры:

Параметр	Описание
<i>точка</i>	Центр окружности.
<i>прямая</i>	Прямая.

Пример:

КР34 = ЦЕНТР, ТЧ6, КАС, ПР8



1.2.4.5 Окружность, определенная точкой центра, и проходящая через точку

Формат:

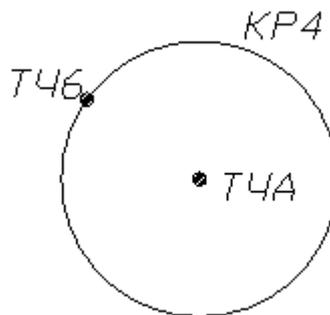
имя = [ЦЕНТР,] *точка1*, *точка2*

Параметры:

Параметр	Описание
<i>точка1</i>	Центр окружности.
<i>точка2</i>	Заданная точка.

Пример:

КР4 = ТЧА, ТЧ6



1.2.4.6 Окружность, проходящая через три точки

Формат:

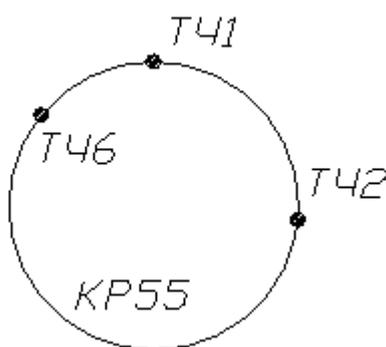
имя = *точка1*, *точка2*, *точка3*

Параметры:

Параметр	Описание
<i>точка1</i>	Первая точка.
<i>точка2</i>	Вторая точка.
<i>точка3</i>	Третья точка.

Пример:

КР55 = ТЧ1, ТЧ6, ТЧ2



1.2.4.7 Окружность, определенная точкой центра, и касающаяся окружности

Формат:

$$\text{имя} = [\text{ЦЕНТР},] \text{точка}, \left\{ \begin{array}{l} \text{МЕНШ} \\ \text{БОЛШ} \end{array} \right\}, [\text{КАС},] \text{окружность}$$

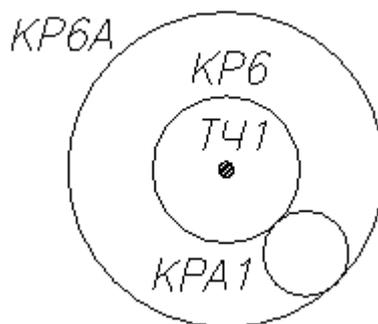
Параметры:

Параметр	Описание
<i>точка</i>	Центр окружности.
МЕНШ	Выбор окружности меньшего радиуса.
БОЛШ	Выбор окружности большего радиуса.
<i>окружность</i>	Окружность.

Пример:

KP6 = T41, МЕНШ, KPA1

KP6A = T41, БОЛШ, KPA1



1.2.4.8 Окружность заданного радиуса, касающаяся двух пересекающихся прямых

Формат:

$$\text{имя} = \begin{Bmatrix} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{Bmatrix}, \text{прямая1}, \begin{Bmatrix} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{Bmatrix}, \text{прямая2}, [\text{РАДИУС},] \text{радиус}$$

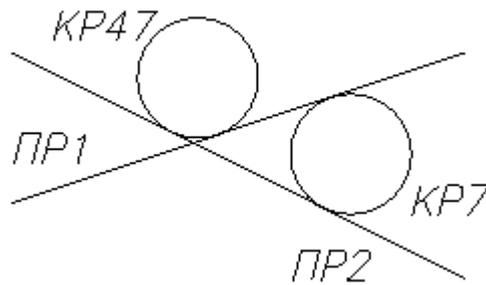
Параметры:

Параметр	Описание
ХБ	Центр окружности смещен в сторону большей координаты X относительно соответствующей прямой.
УБ	Центр окружности смещен в сторону большей координаты Y относительно соответствующей прямой.
ХМ	Центр окружности смещен в сторону меньшей координаты X относительно соответствующей прямой.
УМ	Центр окружности смещен в сторону меньшей координаты Y относительно соответствующей прямой.
<i>прямая1</i>	Первая прямая.
<i>прямая2</i>	Вторая прямая.
<i>радиус</i>	Радиус окружности.

Пример:

КР47 = УБ, ПР1, ХБ, ПР2, 18

КР7 = УМ, ПР1, ХБ, ПР2, 18



1.2.4.9 Окружность заданного радиуса, касающаяся прямой и проходящая через точку

Формат:

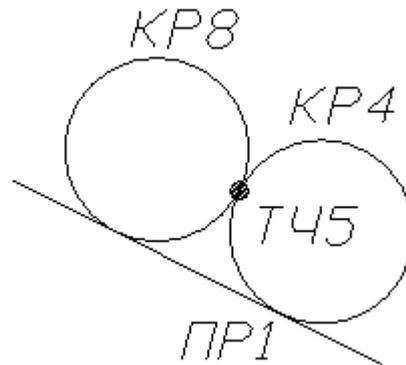
$$\text{имя} = [\text{КАС},] \text{прямая}, \begin{cases} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{cases}, \text{точка}, [\text{РАДИУС},] \text{радиус}$$

Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
ХБ	Центр окружности имеет большую координату X.
УБ	Центр окружности имеет большую координату Y.
ХМ	Центр окружности имеет меньшую координату X.
УМ	Центр окружности имеет меньшую координату Y.
<i>точка</i>	Центр окружности.
<i>радиус</i>	Радиус окружности.

Пример:

КР8 = ПР1, УБ, ТЧ5, 20



1.2.4.10 Окружность заданного радиуса, проходящая через точку, с центром, лежащим на прямой

Формат:

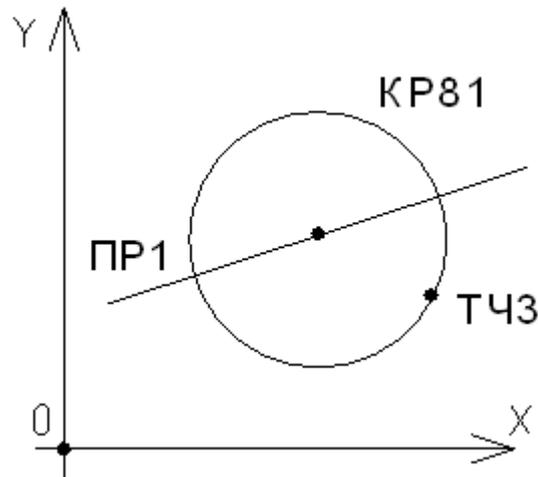
$$\text{имя} = \text{ЦЕНТР}, \text{прямая}, \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \text{точка}, [\text{РАДИУС},] \text{радиус}$$

Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
ХБ	Центр окружности имеет большую координату X.
УБ	Центр окружности имеет большую координату Y.
ХМ	Центр окружности имеет меньшую координату X.
УМ	Центр окружности имеет меньшую координату Y.
<i>точка</i>	Центр окружности.
<i>радиус</i>	Радиус окружности.

Пример:

KP81 = PP1, UB, T43, 20



1.2.4.11 Окружность заданного радиуса, касающаяся прямой и окружности

Формат:

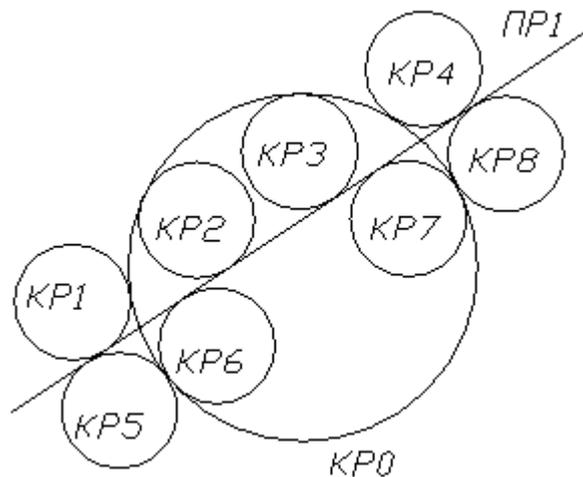
$$\text{имя} = \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \text{прямая}, \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \left\{ \begin{array}{l} \text{ВНЕ} \\ \text{ВНУТРИ} \end{array} \right\}, \text{окружность, радиус}$$

Параметры:

Параметр	Описание
ХБ	Центр окружности смещен в сторону большей координаты X относительно прямой.
УБ	Центр окружности смещен в сторону большей координаты Y относительно прямой.
ХМ	Центр окружности смещен в сторону меньшей координаты X относительно прямой.
УМ	Центр окружности смещен в сторону меньшей координаты Y относительно прямой.
<i>прямая</i>	Прямая.
ХБ	Точка пересечения прямой и окружности имеет большую координату X.
УБ	Точка пересечения прямой и окружности имеет большую координату Y.

Параметр	Описание
ХМ	Точка пересечения прямой и окружности имеет меньшую координату X.
УМ	Точка пересечения прямой и окружности имеет меньшую координату Y.
ВНЕ	Окружность вне опорной окружности.
ВНУТРИ	Окружность внутри опорной окружности.
<i>окружность</i>	Окружность.
<i>радиус</i>	Радиус окружности.

Пример:



KP1 = УБ, ПР1, ХМ, ВНЕ, КР0, 10
KP2 = УБ, ПР1, ХМ, ВНУТРИ, КР0, 10
KP3 = УБ, ПР1, ХБ, ВНУТРИ, КР0, 10
KP4 = УБ, ПР1, ХБ, ВНЕ, КР0, 10
KP5 = УМ, ПР1, ХМ, ВНЕ, КР0, 10
KP6 = УМ, ПР1, ХМ, ВНУТРИ, КР0, 10
KP7 = УМ, ПР1, ХБ, ВНУТРИ, КР0, 10
KP8 = УМ, ПР1, ХБ, ВНЕ, КР0, 10

1.2.4.12 Окружность заданного радиуса, касающаяся двух окружностей

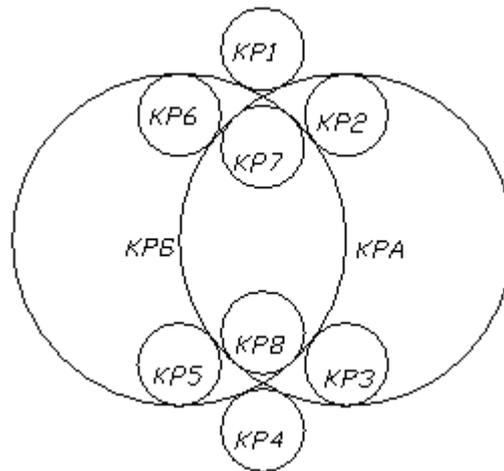
Формат:

$$\text{имя} = \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \left\{ \begin{array}{l} \text{ВНЕ} \\ \text{ВНУТРИ} \end{array} \right\}, \text{окружность1}, \left\{ \begin{array}{l} \text{ВНЕ} \\ \text{ВНУТРИ} \end{array} \right\}, \text{окружность2}, \text{радиус}$$

Параметры:

Параметр	Описание
ХБ	Точка пересечения окружностей имеет большую координату X.
УБ	Точка пересечения окружностей имеет большую координату Y.
ХМ	Точка пересечения окружностей имеет меньшую координату X.
УМ	Точка пересечения окружностей имеет меньшую координату Y.
ВНЕ	Окружность вне окружности1.
ВНУТРИ	Окружность внутри окружности1.
<i>окружность1</i>	Первая окружность.
ВНЕ	Окружность вне окружности2.
ВНУТРИ	Окружность внутри окружности2.
<i>окружность2</i>	Вторая окружность.
<i>радиус</i>	Радиус окружности.

Пример:



КР1 = УБ, ВНЕ, КРА, ВНЕ, КРБ, 15

КР2 = УБ, ВНЕ, КРА, ВНУТРИ, КРБ, 15

КР3 = УМ, ВНЕ, КРА, ВНУТРИ, КРБ, 15

КР4 = УМ, ВНЕ, КРА, ВНЕ, КРБ, 15

КР5 = УМ, ВНУТРИ, КРА, ВНЕ, КРБ, 15

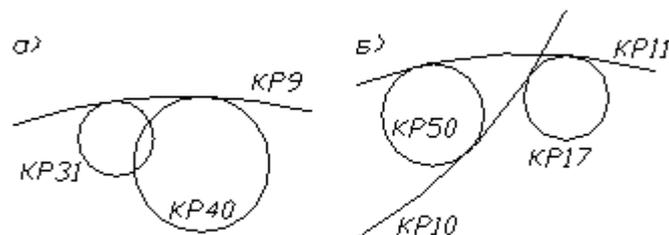
КР6 = УБ, ВНУТРИ, КРА, ВНЕ, КРБ, 15

КР7 = УБ, ВНУТРИ, КРА, ВНУТРИ, КРБ, 15

КР8 = УМ, ВНУТРИ, КРА, ВНУТРИ, КРБ, 15

 В тех случаях, когда определяемая окружность не может быть построена внутри опорных (слишком велик диаметр или опорные окружности не пересекаются), модификатор **ВНУТРИ** указывает положение опорных окружностей относительно определяемой окружности.

Пример:



КР9=УМ,ВНУТРИ,КР31,ВНУТРИ,КР40,150

КР10 = УБ, ВНУТРИ, КР50, ВНЕ, КР17, 60

КР11 = УМ, ВНУТРИ, КР50, ВНУТРИ, КР17, 60

1.2.4.13 Окружность, касающаяся трех прямых

Формат:

$$\text{имя} = \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \text{прямая } 1, \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \text{прямая } 2, \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \text{прямая } 3$$

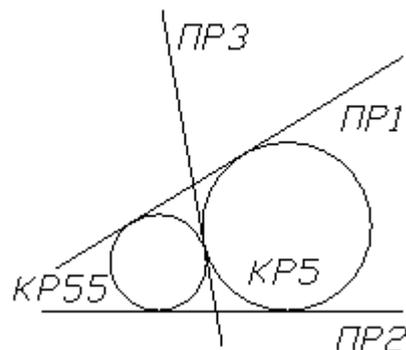
Параметры:

Параметр	Описание
ХБ	Центр окружности смещен в сторону большей координаты X относительно соответствующей прямой.
УБ	Центр окружности смещен в сторону большей координаты Y относительно соответствующей прямой.
ХМ	Центр окружности смещен в сторону меньшей координаты X относительно соответствующей прямой.
УМ	Центр окружности смещен в сторону меньшей координаты Y относительно соответствующей прямой.
<i>прямая1</i>	Первая прямая.
<i>прямая2</i>	Вторая прямая.
<i>прямая3</i>	Третья прямая.

Пример:

KP5 = УМ, ПР1, УБ, ПР2, ХБ, ПР3

KP55 = УМ, ПР1, УБ, ПР2, ХМ, ПР3



1.2.4.14 Окружность заданного радиуса, проходящая через две точки

Формат:

$$\text{имя} = \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \text{точка1}, \text{точка2} [,\text{РАДИУС}], \text{радиус}$$

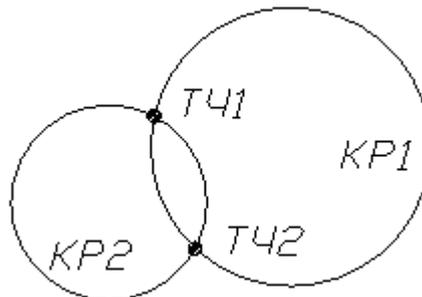
Параметры:

Параметр	Описание
ХБ	Центр окружности имеет большую координату X.
УБ	Центр окружности имеет большую координату Y.
ХМ	Центр окружности имеет меньшую координату X.
УМ	Центр окружности имеет меньшую координату Y.
<i>точка1</i>	Первая точка.
<i>точка2</i>	Вторая точка.
<i>радиус</i>	Радиус окружности.

Пример:

КР1 = ХБ, ТЧ1, ТЧ2, 50

КР2 = ХМ, ТЧ1, ТЧ2, 30



1.2.4.15 Окружность заданного радиуса, проходящая через точку и касающаяся окружности

Формат:

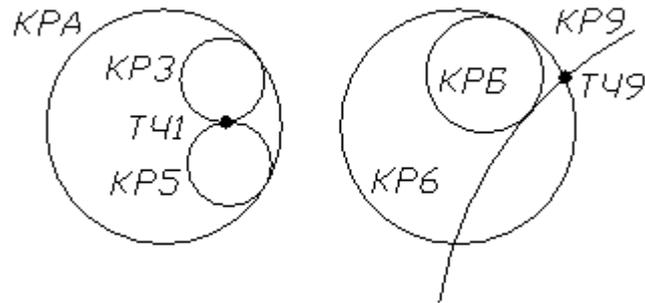
$$\text{имя} = \text{точка}, \left\{ \begin{array}{l} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{array} \right\}, \left\{ \begin{array}{l} \text{ВНЕ} \\ \text{ВНУТРИ} \end{array} \right\}, \text{окружность} [\text{,РАДИУС}], \text{радиус}$$

Параметры:

Параметр	Описание
ХБ	Центр окружности имеет большую координату X.
УБ	Центр окружности имеет большую координату Y.
ХМ	Центр окружности имеет меньшую координату X.
УМ	Центр окружности имеет меньшую координату Y.
ВНЕ	Окружность вне опорной окружности.
ВНУТРИ	Окружность внутри опорной окружности.
<i>точка</i>	Точка.
<i>окружность</i>	Окружность.
<i>радиус</i>	Радиус окружности.

 В том случае, когда точка, указанная в определении, находится внутри опорной окружности, использование модификатора **ВНЕ** не имеет смысла. Если опорная точка находится вне опорной окружности, возможно как внутреннее, так и внешнее касание. Модификатор **ВНУТРИ** в этом случае указывает положение опорной окружности относительно определяемой (окружность **КР6** в приведенном ниже примере).

Пример:



КР3 = ТЧ1, УБ, ВНУТРИ, КРА, 25
КР5 = ТЧ1, УМ, ВНУТРИ, КРА, 25
КР6 = ТЧ9, УМ, ВНУТРИ, КРБ, 14
КР9 = ТЧ9, УБ, ВНЕ, КРБ, 14

1.2.4.16 Окружность, концентричная заданной окружности

Формат:

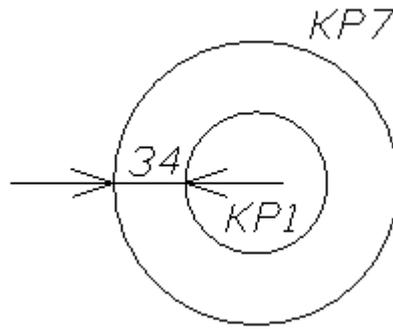
$$\text{имя} = [\text{ПАРЛЕЛ},] \left\{ \begin{array}{l} \text{МЕНШ} \\ \text{БОЛШ} \end{array} \right\}, \text{окружность}, \text{расстояние}$$

Параметры:

Параметр	Описание
МЕНШ	Окружность имеет меньший радиус.
БОЛШ	Окружность имеет больший радиус.
<i>окружность</i>	Окружность.
<i>расстояние</i>	Разность радиусов окружностей.

Пример:

КР7 = ПАРЛЕЛ, КР1, БОЛШ, 34



1.2.4.17 Окружность, заданная дугой, являющейся сегментом контура

Формат:

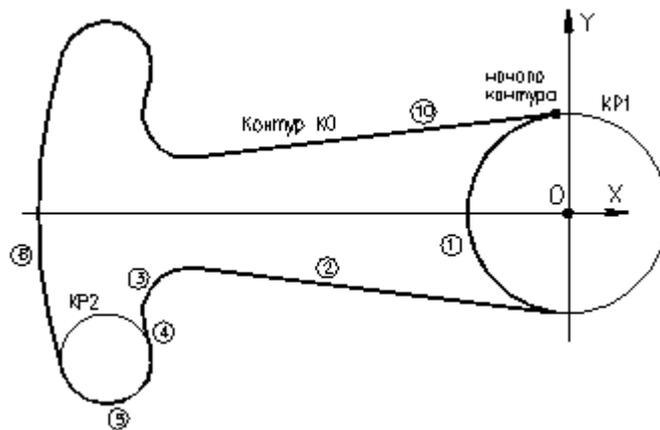
имя = контур, число

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>число</i>	Номер сегмента контура.

Пример:

KP1 = K0,1
KP2 = K0,5



1.2.4.18 Окружность, полученная в результате преобразования координат

Формат:

```
имя = окружность, матрица
```

Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
<i>матрица</i>	Матрица преобразования координат.

Пример:

KP77 = KP2, MATP6

1.2.5 Определение вектора

В определении вектора используют модификаторы, указывающие направление вектора.

- **XB** – вектор направлен в сторону увеличения координаты **X**;
- **YB** – вектор направлен в сторону увеличения координаты **Y**;
- **XM** – вектор направлен в сторону уменьшения координаты **X**.
- **YM** – вектор направлен в сторону уменьшения координаты **Y**.

Темы этого раздела:

- [Вектор, равный ранее определенному вектору](#)^[92]
- [Вектор, определенный проекциями на оси и идущий из начала координат](#)^[93]
- [Вектор, определенный своими начальной и конечной точками](#)^[93]
- [Вектор, равный сумме двух других векторов](#)^[94]
- [Вектор, равный разности двух других векторов](#)^[95]
- [Вектор, равный произведению заданного вектора и числа](#)^[96]
- [Вектор, являющийся радиус-вектором заданной точки](#)^[96]
- [Вектор, заданный координатами своих начальной и конечной точек](#)^[97]
- [Вектор, параллельный заданной прямой](#)^[98]
- [Вектор, равный произведению двух векторов](#)^[98]
- [Вектор, полученный в результате преобразования координат](#)^[99]

1.2.5.1 Вектор, равный ранее определенному вектору

Формат:

```
имя = вектор
```

Параметры:

Параметр	Описание
<i>вектор</i>	Вектор.

Пример:

ВЕР4 = ВКР6

1.2.5.2 Вектор, определенный проекциями на оси и идущий из начала координат

Формат:

имя = проекция x, проекция y [, проекция z]

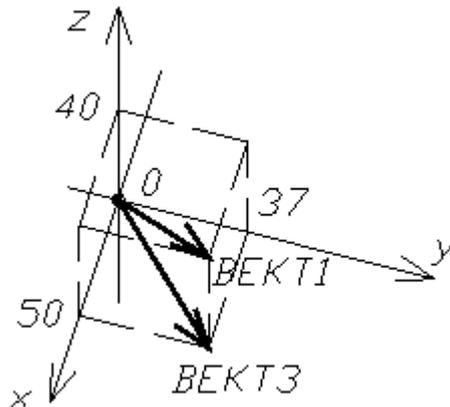
Параметры:

Параметр	Описание
<i>проекция x</i>	Проекция вектора на ось X .
<i>проекция y</i>	Проекция вектора на ось Y .
<i>проекция z</i>	Проекция вектора на ось Z .

Пример:

ВЕКТ1 = 50,37,40

ВЕКТ3 = 50,37



1.2.5.3 Вектор, определенный своими начальной и конечной точками

Формат:

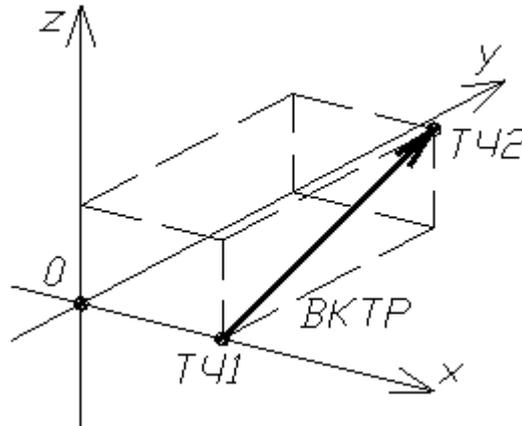
имя = точка1, точка2

Параметры:

Параметр	Описание
<i>точка1</i>	Начальная точка вектора.
<i>точка2</i>	Конечная точка вектора.

Пример:

ВКТР = ТЧ1, ТЧ2



1.2.5.4 Вектор, равный сумме двух других векторов

Формат:

имя = вектор1, ПЛЮС, вектор2

Параметры:

Параметр	Описание
<i>вектор1</i>	Первый вектор.
<i>вектор2</i>	Второй вектор.

Пример:

ВЕКТ4 = ВЕКР1, ПЛЮС, ВКТ2



1.2.5.5 Вектор, равный разности двух других векторов

Формат:

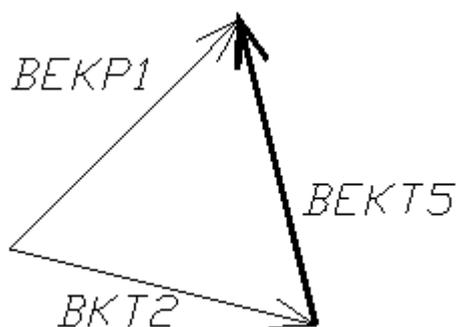
имя = *вектор1*, МИНУС, *вектор2*

Параметры:

Параметр	Описание
<i>вектор1</i>	Первый вектор.
<i>вектор2</i>	Второй вектор.

Пример:

ВЕКТ5 = ВЕКР1, МИНУС, ВКТ2



1.2.5.6 Вектор, равный произведению заданного вектора и числа

Формат:

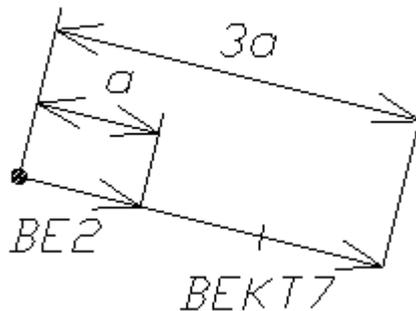
<i>имя</i> = число, вектор

Параметры:

Параметр	Описание
<i>число</i>	Число.
<i>вектор</i>	Вектор.

Пример:

$$\text{ВЕКТ7} = 3, \text{ВЕ2}$$



1.2.5.7 Вектор, являющийся радиус-вектором заданной точки

Формат:

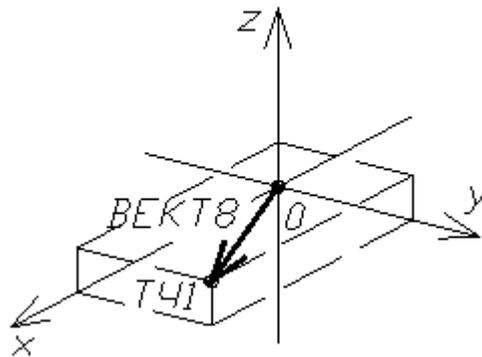
<i>имя</i> = точка

Параметры:

Параметр	Описание
<i>точка</i>	Точка.

Пример:

$$\text{ВЕКТ8} = \text{ТЧ1}$$



1.2.5.8 Вектор, заданный координатами своих начальной и конечной точек

Формат:

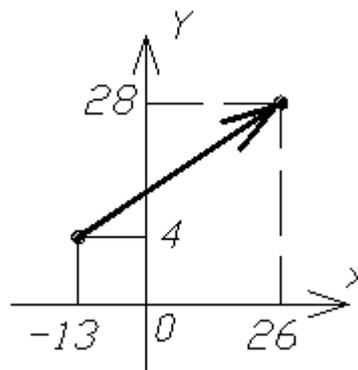
$ИМЯ = x_1, y_1, x_2, y_2$

Параметры:

Параметр	Описание
x_1 y_1	Координаты начальной точки.
x_2 y_2	Координаты конечной точки.

Пример:

ВЕКТ В1 = -13, 4, 26, 28



1.2.5.9 Вектор, параллельный заданной прямой

Формат:

$$\text{имя} = \text{прямая}, \begin{cases} \text{ХБ} \\ \text{УБ} \\ \text{ХМ} \\ \text{УМ} \end{cases}$$

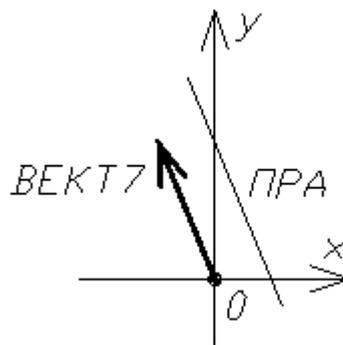
Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
ХБ	Вектор направлен в сторону большей координаты X.
УБ	Вектор направлен в сторону большей координаты Y.
ХМ	Вектор направлен в сторону меньшей координаты X.
УМ	Вектор направлен в сторону меньшей координаты Y.

 Длина вектора равна единице, начало его находится в начале координат.

Пример:

ВЕКТ7 = ПРА, ХМ



1.2.5.10 Вектор, равный произведению двух векторов

Формат:

имя = *вектор1*, **УМНОЖ**, *вектор2*

Параметры:

Параметр	Описание
<i>вектор1</i>	Первый вектор.
<i>вектор2</i>	Второй вектор.

Пример:

ВЕКТ11 = ВА36, УМНОЖ, ВР7

1.2.5.11 Вектор, полученный в результате преобразования координат

Формат:

имя = вектор, матрица

Параметры:

Параметр	Описание
<i>вектор</i>	Вектор.
<i>матрица</i>	Матрица преобразования координат.

Пример:

ВЕКТОР В6 = ВЕКТ, М2

1.2.6 Определение матрицы

Матрица определяет способ преобразования координат. Это преобразование может быть:

- параллельным переносом исходной системы координат;
- поворотом исходной системы координат;
- симметрией;
- комбинацией этих преобразований.

Любое преобразование координатной системы может быть описано с помощью системы трех уравнений:

$$a_{11}x_1 + a_{12}y_1 + a_{13}z_1 + a_{14} = x_2$$

$$a_{21}x_1 + a_{22}y_1 + a_{23}z_1 + a_{24} = y_2$$

$$a_{31}x_1 + a_{32}y_1 + a_{33}z_1 + a_{34} = z_2$$

Всегда можно определить значения таких коэффициентов a_{11} , a_{12} , ..., a_{34} , что при подстановке в эту систему уравнений значений координат произвольной точки x_1 , y_1 , z_1 в локальной системе координат будут получены координаты этой точки x_2 , y_2 , z_2 в абсолютной (исходной) системе координат.

Коэффициенты данной системы уравнений используются для представления матрицы

преобразования в канонической форме и выводятся на печать в следующей последовательности:

$$a_{11} \ a_{12} \ a_{13}$$

$$a_{21} \ a_{22} \ a_{23}$$

$$a_{31} \ a_{32} \ a_{33}$$

$$a_{14} \ a_{24} \ a_{34}$$

Первые три элемента каждого столбца определяют направление осей локальной системы координат по отношению к исходной (абсолютной) системе координат:

$$a_{1p} \ a_{2p} \ a_{3p} \quad \text{-единичный вектор оси } x_p;$$

$$a_{1p} \ a_{2p} \ a_{3p} \quad \text{-единичный вектор оси } y_p;$$

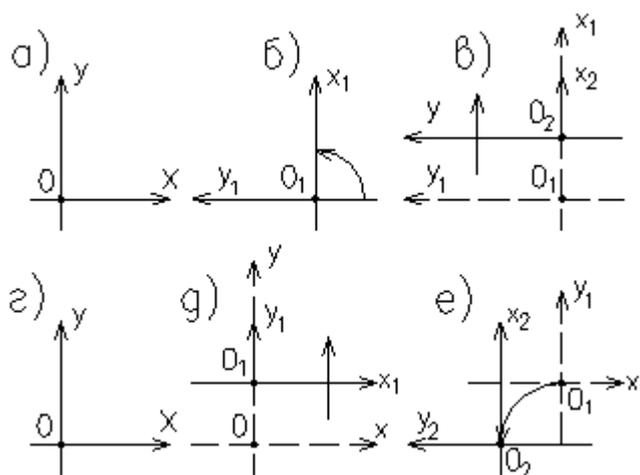
$$a_{1p} \ a_{2p} \ a_{3p} \quad \text{-единичный вектор оси } z_p.$$

Четвертая строка определяет положение начала координат локальной системы: a_{1p} , a_{2p} , a_{3p} - координаты x , y , z начала локальной системы координат в исходной (абсолютной) системе.

Результатом матричного преобразования является геометрический объект, который имеет в локальной системе координат, полученной преобразованием исходной системы, параметры исходного геометрического объекта.

При использовании комбинированных матричных преобразований, задаваемых произведением матриц следует помнить, что матричное умножение не коммутативно, т.е. конечный результат зависит от последовательности элементарных преобразований. Это иллюстрирует следующий рисунок, на котором представлено последовательное преобразование:

- исходной системы координат (а) матрицей поворота (б) и матрицей переноса (в);
- исходной системы координат (г) матрицей переноса (д) и матрицей поворота (е);



В геометрических определениях матрицы используются следующие модификаторы:

ПОВХУ – поворот осей **X** и **Y** вокруг оси **Z**;

ПОВУЗ – поворот осей **Y** и **Z** вокруг оси **X**;

ПОВЗХ – поворот осей **Z** и **X** вокруг оси **Y**.

(Направление отсчета углов поворота – в соответствии с рисунком.)

Темы этого раздела:

- [Матрица, равная ранее определенной матрице](#)^[101]
- [Матрица, определяющая преобразование, обратное заданному преобразованию](#)^[101]
- [Матрица, определяющая систему координат, полученную поворотом на угол вокруг оси](#)^[102]
- [Матрица, определяющая систему координат, полученную параллельным переносом](#)^[103]
- [Матрица, определяющая координаты, полученные параллельным переносом и поворотом](#)^[104]
- [Матрица, являющаяся произведением двух других матриц](#)^[105]
- [Матрица, определяющая перенос начала координат в точку, заданную явно или вектором](#)^[105]
- [Матрица, определяющая поворот с последующим переносом начала координат](#)^[106]
- [Матрица, определяющая симметрию относительно заданной прямой](#)^[107]
- [Матрица, определяющая систему координат, направления осей которой заданы векторами](#)^[107]
- [Матрица, определяющая систему координат с началом в точке и с осями-векторами](#)^[108]
- [Матрица, являющаяся зеркальным отображением исходной относительно плоскости](#)^[109]
- [Матрица, определяющая масштабирование объекта](#)^[110]

1.2.6.1 Матрица, равная ранее определенной матрице

Формат:

имя = матрица

Параметры:

Параметр	Описание
<i>матрица</i>	Матрица преобразования координат.

Пример:

МАТР1 = МТ6

1.2.6.2 Матрица, определяющая преобразование, обратное заданному преобразованию

Формат:

имя = ОБРАТН, матрица

Параметры:

Параметр	Описание
<i>матрица</i>	Матрица преобразования координат.

Пример:

Если **ТЧ2=МАТР2,ТЧ1** и **М12=ОБРАТН,МАТР2**,
то справедливо геометрическое определение: **ТЧ1 = ТЧ2, М12**.

1.2.6.3 Матрица, определяющая систему координат, полученную поворотом на угол вокруг оси

Формат:

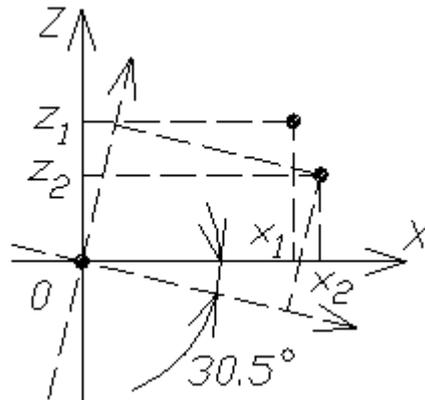
$$\text{имя} = \left\{ \begin{array}{l} \text{ПОВХУ} \\ \text{ПОВУЗ} \\ \text{ПОВЗХ} \end{array} \right\}, \text{угол} [, \text{ЦЕНТР}, \text{точка}]$$

Параметры:

Параметр	Описание
ПОВХУ (ПОВОРОТ)	Поворот системы координат вокруг оси Z .
ПОВУЗ	Поворот системы координат вокруг оси X .
ПОВЗХ	Поворот системы координат вокруг оси Y .
<i>угол</i>	Величина угла поворота в градусах: – по часовой стрелке; + против часовой стрелки.
<i>точка</i>	Точка, определяющая начало новой системы координат.

Пример:

МАТР2 = ПОВЗХ, 30.5



1.2.6.4 Матрица, определяющая систему координат, полученную параллельным переносом

Формат:

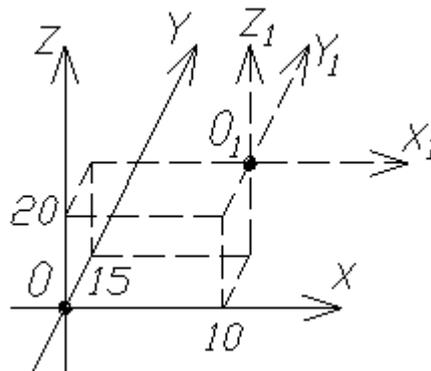
$МММ = \text{ПЕРЕНОС}, x, y [, z]$

Параметры:

Параметр	Описание
x y z	Координаты точки переноса.

Пример:

МАТР3 = ПЕРЕНОС, 10, 15, 20



1.2.6.5 Матрица, определяющая координаты, полученные параллельным переносом и поворотом

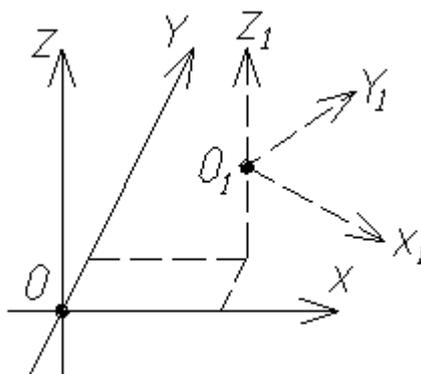
Формат:

$$\text{мля} = \left\{ \begin{array}{l} \text{ПОВХУ} \\ \text{ПОВУЗ} \\ \text{ПОВЗХ} \end{array} \right\}, \text{угол}, \text{ПЕРЕНОС}_{x,y,z}$$

Параметры:

Параметр	Описание
ПОВХУ (ПОВОРОТ)	Поворот системы координат вокруг оси Z .
ПОВУЗ	Поворот системы координат вокруг оси X .
ПОВЗХ	Поворот системы координат вокруг оси Y .
<i>угол</i>	Величина угла поворота в градусах: – по часовой стрелке; + против часовой стрелки.
<i>x</i> <i>y</i> <i>z</i>	Координаты точки переноса.

Пример:

МАТР4 = ПОВЗХ, 30.5, ПЕРЕНОС, 10, 15, 20

1.2.6.6 Матрица, являющаяся произведением двух других матриц

Формат:

```
имя = матрица1, матрица2
```

Параметры:

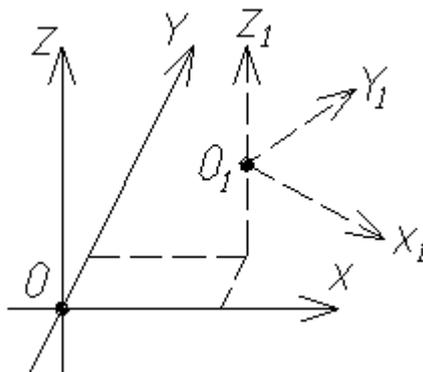
Параметр	Описание
матрица1	Первая матрица.
матрица2	Вторая матрица.

Произведение матриц эквивалентно применению сначала матрицы2, а затем преобразованию полученного результата в соответствии с матрицей1.

Пример:

Матрица, заданная в примере является произведением матрицы переноса (МАТР3) и матрицы поворота (MT2)

- МАТР2 = ПОВЗХ, 30.5
- МАТР3 = ПЕРЕНОС, 10, 15, 20
- МАТР5 = МАТР3, MT2



1.2.6.7 Матрица, определяющая перенос начала координат в точку, заданную явно или вектором

Формат:

```
имя = ПЕРЕНОС, {точка}
                {вектор}
```

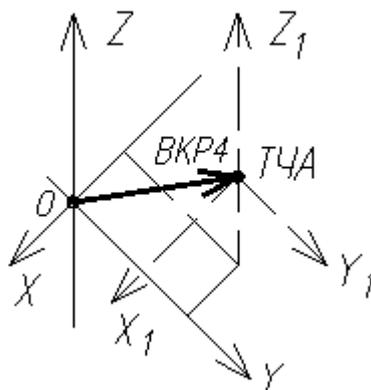
Параметры:

Параметр	Описание
<i>точка</i>	Заданная точка.
<i>вектор</i>	Заданный вектор.

Пример:

МАТР6 = ПЕРЕНОС, ТЧА

МАТР6А = ПЕРЕНОС, ВКР4



1.2.6.8 Матрица, определяющая поворот с последующим переносом начала координат

Формат:

$$\text{имя} = \left\{ \begin{array}{l} \text{точка} \\ \text{вектор} \end{array} \right\}, \left\{ \begin{array}{l} \text{ПОВХУ} \\ \text{ПОВУЗ} \\ \text{ПОВЗХ} \end{array} \right\}, \text{угол}$$

Параметры:

Параметр	Описание
<i>точка</i>	Заданная точка.
<i>вектор</i>	Заданный вектор.
ПОВХУ (ПОВОРОТ)	Поворот системы координат вокруг оси Z .
ПОВУЗ	Поворот системы координат вокруг оси X .
ПОВЗХ	Поворот системы координат вокруг оси Y .

Параметр	Описание
<i>угол</i>	Величина угла поворота в градусах: – по часовой стрелке; + против часовой стрелки.

Пример:

МАТР7 = ТЧА, ПОВЗХ, 30.5

1.2.6.9 Матрица, определяющая симметрию относительно заданной прямой

Формат:

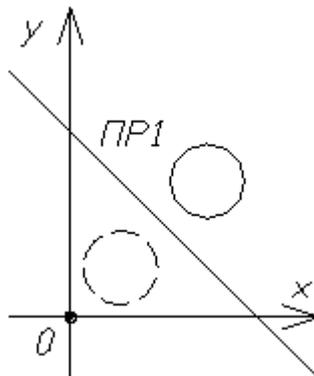
имя = прямая

Параметры:

Параметр	Описание
<i>прямая</i>	Заданная прямая.

Пример:

МАТР8 = ПР1



1.2.6.10 Матрица, определяющая систему координат, направления осей которой заданы векторами

Формат:

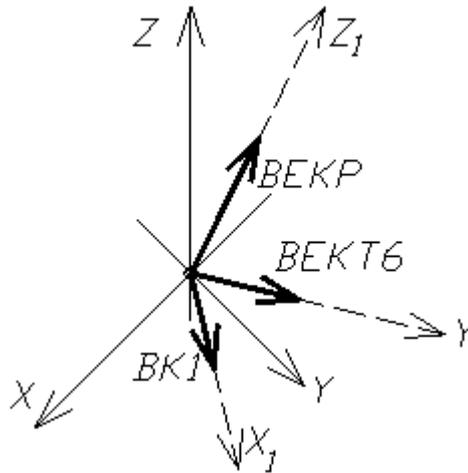
имя = вектор1, вектор2, вектор3

Параметры:

Параметр	Описание
<i>вектор1</i>	Первая ось системы координат.
<i>вектор2</i>	Вторая ось системы координат.
<i>вектор3</i>	Третья ось системы координат.

Пример:

МАТР9 = ВК1, ВЕКТ6, ВЕКР



1.2.6.11 Матрица, определяющая систему координат с началом в точке и с осями-векторами

Формат:

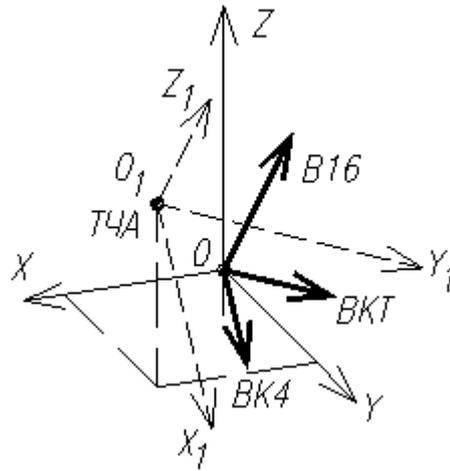
$\text{имя} = \text{вектор1}, \text{вектор2}, \text{вектор3}, \text{ПЕРЕНОС}, \left\{ \begin{array}{l} \text{точка} \\ \text{вектор4} \end{array} \right\}$

Параметры:

Параметр	Описание
<i>вектор1</i>	Первая ось системы координат.
<i>вектор2</i>	Вторая ось системы координат.
<i>вектор3</i>	Третья ось системы координат.
<i>точка</i>	Заданная точка начала координат.
<i>вектор4</i>	Заданный вектор начала координат.

Пример:

МАТР10 = ВК4, ВКТ, В16, ПЕРЕНОС, ТЧА



1.2.6.12 Матрица, являющаяся зеркальным отображением исходной относительно плоскости

Формат:

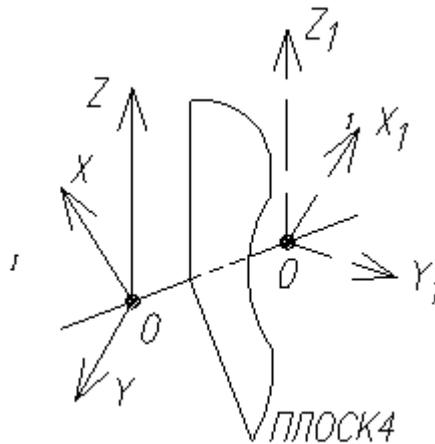
Имя = плоскость

Параметры:

Параметр	Описание
<i>плоскость</i>	Заданная плоскость.

Пример:

МАТР11 = ПЛОСК4



1.2.6.13 Матрица, определяющая масштабирование объекта

Формат:

```
имя = МАСШТАБ, число [, ЦЕНТР, точка]
```

Параметры:

Параметр	Описание
число	Коэффициент масштабирования.
точка	Точка, определяющая центр преобразования

Пример:

```
МАТР12 = МАСШТАБ, 2.5
```

1.2.7 Определение плоскости

В определениях плоскости используются следующие модификаторы.

ХУПАР – определяемая плоскость параллельна плоскости **ХУ**;

УЗПАР – определяемая плоскость параллельна плоскости **УЗ**;

ЗХПАР – определяемая плоскость параллельна плоскости **ЗХ**.

Отсутствие модификатора эквивалентно использованию модификатора **ХУПАР**.

Темы этого раздела:

- [Плоскость, совпадающая с ранее определенной плоскостью](#)^[110]
- [Плоскость, полученная преобразованием заданной плоскости в соответствии с матрицей](#)^[111]
- [Плоскость, параллельная одной из координатных плоскостей и проходящая через точку](#)^[111]
- [Плоскость, параллельная одной из координатных плоскостей на заданном расстоянии от нее](#)^[112]
- [Плоскость, параллельная другой плоскости и проходящая через заданную точку](#)^[112]
- [Плоскость, определенная коэффициентами ее нормального уравнения](#)^[112]
- [Плоскость, проходящая через три точки, не лежащие на одной прямой](#)^[113]
- [Плоскость, проходящая через заданную точку перпендикулярно вектору](#)^[113]
- [Плоскость, параллельная двум векторам и проходящая через заданную точку](#)^[114]

1.2.7.1 Плоскость, совпадающая с ранее определенной плоскостью

Формат:

```
имя = плоскость
```

Параметры:

Параметр	Описание
<i>плоскость</i>	Плоскость.

Пример:

ПЛОСК1 = ПЛОСК8

1.2.7.2 Плоскость, полученная преобразованием заданной плоскости в соответствии с матрицей

Формат:

имя = плоскость, матрица

Параметры:

Параметр	Описание
<i>плоскость</i>	Плоскость.
<i>матрица</i>	Матрица.

Пример:

ПЛОСК2 = ПЛОСК8, МТ6

1.2.7.3 Плоскость, параллельная одной из координатных плоскостей и проходящая через точку

Формат:

имя = $\left\{ \begin{array}{l} \text{ХУПАР} \\ \text{УЗПАР} \\ \text{ЗХПАР} \end{array} \right\}$, точка

Параметры:

Параметр	Описание
ХУПАР	Плоскость параллельная плоскости XY .
УЗПАР	Плоскость параллельная плоскости YZ .
ЗХПАР	Плоскость параллельная плоскости ZX .
<i>точка</i>	Точка.

Пример:

ПЛОСК3 = УЗПАР, ТЧ3

1.2.7.4 Плоскость, параллельная одной из координатных плоскостей на заданном расстоянии от нее

Формат:

$$\text{имя} = \left\{ \begin{array}{l} \text{ХУПАР} \\ \text{УЗПАР} \\ \text{ЗХПАР} \end{array} \right\}, \text{число}$$

Параметры:

Параметр	Описание
ХУПАР	Плоскость параллельная плоскости XY .
УЗПАР	Плоскость параллельная плоскости YZ .
ЗХПАР	Плоскость параллельная плоскости ZX .
число	Координата пересечения определяемой плоскости с осью, перпендикулярной координатной плоскости (расстояние).

Пример:

ПЛОСК4 = ЗХПАР, -29.6

1.2.7.5 Плоскость, параллельная другой плоскости и проходящая через заданную точку

Формат:

$$\text{имя} = \text{точка}, \text{ПАРЛЕЛ}, \text{плоскость}$$

Параметры:

Параметр	Описание
точка	Точка.
плоскость	Плоскость.

Пример:

ПЛОСК5 = ТЧЗ, ПАРЛЕЛ, ПЛОСК8

1.2.7.6 Плоскость, определенная коэффициентами ее нормального уравнения

Формат:

$$\text{имя} = \text{число1}, \text{число2}, \text{число3}, \text{число4}$$

Параметры:

Параметр	Описание
<i>число1</i>	Коэффициент при координате X.
<i>число2</i>	Коэффициент при координате Y.
<i>число3</i>	Коэффициент при координате Z.
<i>число4</i>	Свободный член уравнения.

Пример:

ПЛОСК6 = 1, 0, 0, 50

1.2.7.7 Плоскость, проходящая через три точки, не лежащие на одной прямой

Формат:

имя = точка1, точка2, точка3

Параметры:

Параметр	Описание
<i>точка1</i>	Первая точка.
<i>точка2</i>	Вторая точка.
<i>точка3</i>	Третья точка.

Пример:

ПЛОСК7 = ТЧ1, ТЧ18, ТЧ46

1.2.7.8 Плоскость, проходящая через заданную точку перпендикулярно вектору

Формат:

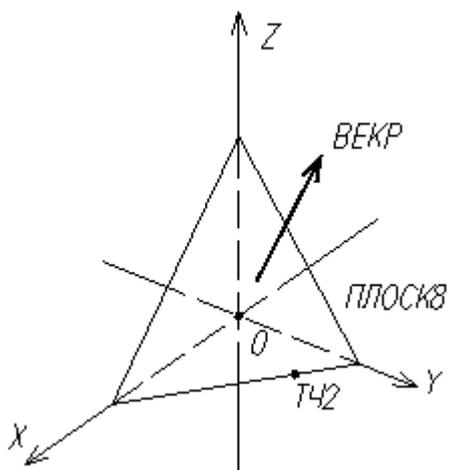
имя = вектор, точка

Параметры:

Параметр	Описание
<i>вектор</i>	Вектор.
<i>точка</i>	Точка.

Пример:

ПЛОСК8 = ВЕКР, ТЧ2



1.2.7.9 Плоскость, параллельная двум векторам и проходящая через заданную точку

Формат:

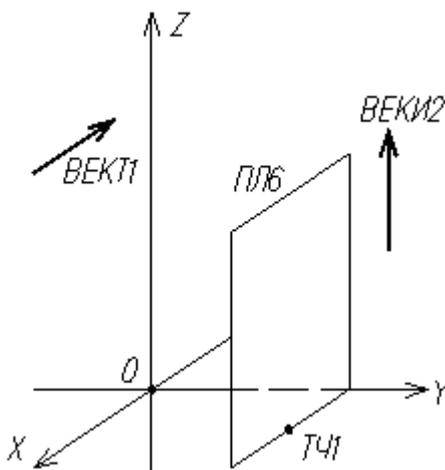
имя = вектор1, вектор2, точка

Параметры:

Параметр	Описание
<i>вектор1</i>	Первый вектор.
<i>вектор2</i>	Второй вектор.
<i>точка</i>	Точка.

Пример:

ПЛОСК9 = ВЕК1, ВЕК2, ТЧ1



1.2.8 Вложенное определение

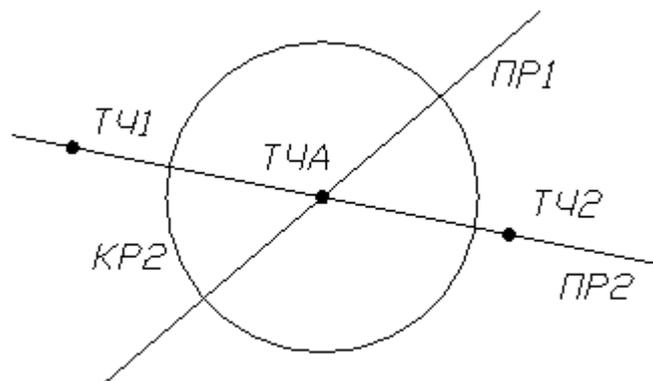
Вместо имени ранее определенного геометрического объекта в правой части геометрического определения или в операторе движения можно задать "вложенное определение". Оно состоит из типа определяемого объекта, необязательного имени объекта и заключенного в скобки варианта геометрического определения соответствующего типа.

Формат вложенного определения:

```
тип [ имя ] (геометрическое определение)
```

Если у такого геометрического объекта отсутствует имя, то значение его геометрических параметров сохраняется временно и используется процессором только для промежуточных вычислений.

На следующем рисунке изображена окружность **КР2**, которая может быть описана с использованием вложенных определений:



КР2 = ТОЧКА (ПРЯМАЯ ПР2 (ТЧ1, ТЧ2), ПР1), РАДИУС, 2.5

Данное геометрическое определение равносильно трем другим определениям:

ПР2=ТЧ1, ТЧ2

ТЧА=ПР2, ПР1

КР2=ЦЕНТР, ТЧА, РАДИУС, 2.5

Различие состоит в том, что значение точки **ТЧА** во втором варианте записи сохранится, а при использовании вложенного определения точка пересечения прямых **ПР1** и **ПР2** не может быть использована в программе повторно. Глубину вложения можно увеличивать практически неограниченно.

1.2.9 Массивы точек

В определениях массива точек используются следующие группы модификаторов.

- Модификаторы направления отсчета угла:
 - **ПОЧС** – задается отсчет угла по часовой стрелке;
 - **ПРЧС** – задается отсчет угла против часовой стрелки.

- Модификаторы направления обхода контура
 - **ПО** – задается обход контура в прямом направлении (в порядке описания сегментов контура);
 - **ОБРАТН** – задается обход контура в обратном направлении (в порядке обратном описанию сегментов контура).
- Модификаторы положения точек относительно контура:
 - **ВНЕ** – выбираются точки вне контура;
 - **ВНУТРИ** – выбираются точки внутри контура;
 - **СЛЕВА** – выбираются точки слева от контура;
 - **СПРАВА** – выбираются точки справа контура;
 - **ПЕРЕСЕЧ** – выбираются точки на контуре.

Темы этого раздела:

- [Точки, равномерно расположенные между двумя точками](#)^[116]
- [Перенос точек заданное число раз](#)^[117]
- [Точки, полученные переносом точки под углом с заданным интервалом](#)^[118]
- [Точки, расположенные по контуру с заданным интервалом](#)^[119]
- [Точки, равномерно расположенные по контуру](#)^[120]
- [Точки, равномерно расположенные по окружности](#)^[120]
- [Точки, равномерно расположенные по сектору окружности](#)^[121]
- [Точки, расположенных по окружности с заданным угловым интервалом](#)^[122]
- [Все опорные точки контура](#)^[123]
- [Массив точек, упорядоченных контуром](#)^[123]
- [Включение точек в массив точек](#)^[124]
- [Исключение точек из массива точек](#)^[125]
- [Преобразование координат точек](#)^[125]
- [Перенос массива точек в каждую точку другого массива](#)^[126]
- [Отсечение точек массива ограниченной областью](#)^[127]
- [Точки пересечения контура и геометрического объекта](#)^[128]
- [Точки в узлах прямоугольной сетки, заполняющие прямоугольную область](#)^[129]
- [Точки в узлах прямоугольной сетки, заполняющие ограниченную область](#)^[131]

1.2.9.1 Точки, равномерно расположенные между двумя точками

Формат:

```
имя = точка1, точка2, число
```

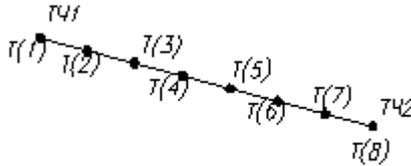
Параметры:

Параметр	Описание
<i>точка₁</i>	Начальная точка отрезка.
<i>точка₂</i>	Конечная точка отрезка.

Параметр	Описание
<i>число</i>	Количество точек.

Пример:

ТОЧКА Т0=ТЧ1,ТЧ2, 8



1.2.9.2 Перенос точек заданное число раз

Формат:

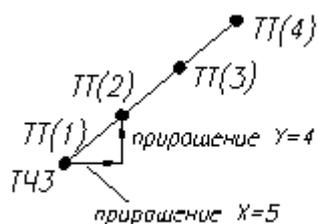
$\text{имя} = \left\{ \begin{array}{l} \text{точка} \\ \text{координаты} \\ \text{массив точек} \end{array} \right\}, \text{ ПЕРЕНОС}, \text{ приращение}_x, \text{ приращение}_y, \text{ число}$

Параметры:

Параметр	Описание
<i>точка</i>	Точка, заданная геометрическим элементом.
<i>координаты</i>	Точка, заданная своими координатами.
<i>массив точек</i>	Массив точек.
<i>приращение_x</i>	Приращение по оси X.
<i>приращение_y</i>	Приращение по оси Y.
<i>число</i>	Число точек.

Пример:

ТТ=ТЧ3, ПЕРЕНОС, 5, 4, 4



1.2.9.3 Точки, полученные переносом точки под определенным углом с заданным интервалом

Формат:

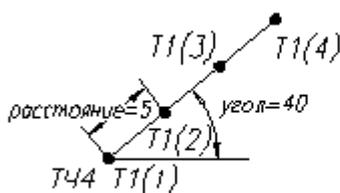
имя = точка, ПОВОРОТ, угол, расстояние, число

Параметры:

Параметр	Описание
<i>точка</i>	Точка.
<i>угол</i>	Угол с осью X.
<i>расстояние</i>	Расстояние между точками.
<i>число</i>	Число точек.

Пример:

T1=T44, ПОВОРОТ, 40, 5, 4



1.2.9.4 Точки, расположенные по контуру с заданным интервалом

Формат:

```
имя = контур, { ПО
                ОБРАТН }, расстояние [ ,СМЕЩЕНИЕ, смещение ]
```

Параметры:

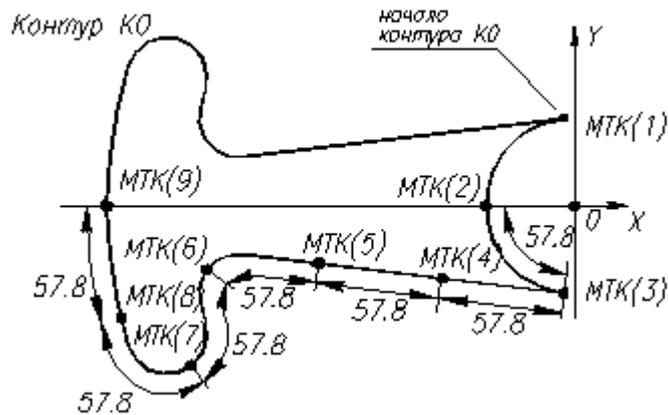
Параметр	Описание
<i>контур</i>	Контур.
ПО	Обход контура в прямом направлении.
ОБРАТН	Обход контура в обратном направлении.
<i>расстояние</i>	Расстояние между точками.
СМЕЩЕНИЕ, смещение	Смещение точек по нормали к контуру. Для рассчитанных точек вводится дополнительное смещение по нормали к контуру. Знак смещения определяет сторону, в которую смещаются точки относительно контура (при обходе по описанию контура): <i>смещение</i> > 0 – вправо <i>смещение</i> < 0 – влево.

☞ Порядок заполнения массива точек определяется направлением построения контура (значение индекса увеличивается по направлению обхода контура от его начала).

Пример:

МТК=K0, ПО, 57.8

МТК1=K0, ПО, 5, СМЕЩЕНИЕ, 1



1.2.9.5 Точки, равномерно расположенные по контуру

Формат:

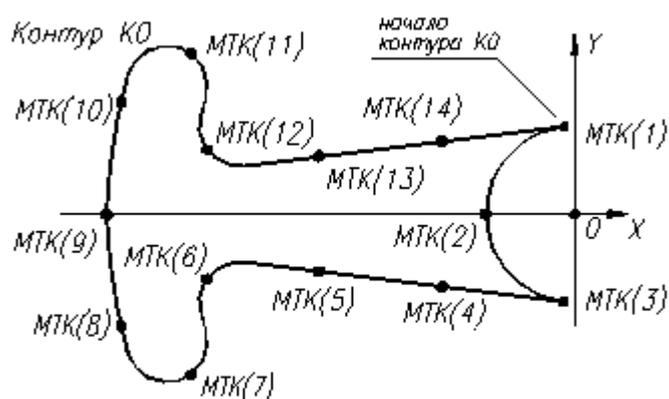
<i>имя = контур, число</i>

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>число</i>	Число точек.

Пример:

МТК=К0, 14



1.2.9.6 Точки, равномерно расположенные по окружности

Формат:

<i>имя = окружность, {</i>	<i>ПОЧС</i>	<i>}, угол, число</i>
	<i>ПРЧС</i>	

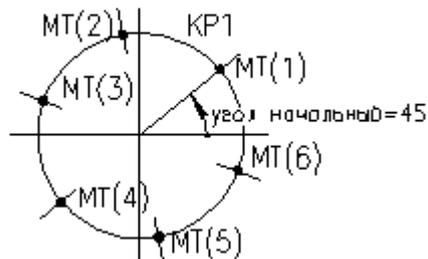
Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
ПОЧС	Движение по окружности по часовой стрелке.
ПРЧС	Движение по окружности против часовой стрелки.

Параметр	Описание
<i>угол</i>	Угол с осью X радиус-вектора начальной точки.
<i>число</i>	Число точек.

Пример:

MT=KP1, ПРЧС, 45, 6



1.2.9.7 Точки, равномерно расположенные по сектору окружности

Формат:

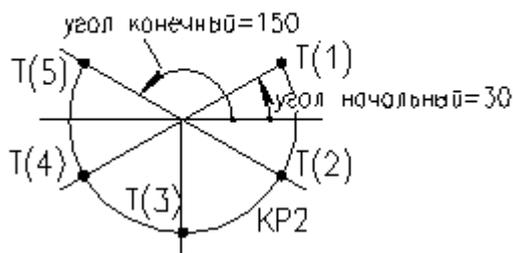
имя = *окружность*, { **ПОЧС** / **ПРЧС** }, *угол₁*, *угол₂*, *число*

Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
ПОЧС	Движение по окружности по часовой стрелке.
ПРЧС	Движение по окружности против часовой стрелки.
<i>угол₁</i>	Угол с осью X радиус-вектора начальной точки.
<i>угол₂</i>	Угол с осью X радиус-вектора конечной точки.
<i>число</i>	Число точек.

Пример:

T=KP2, ПОЧС, 30, 150, 5



1.2.9.8 Точки, расположенные по окружности с заданным угловым интервалом

Формат:

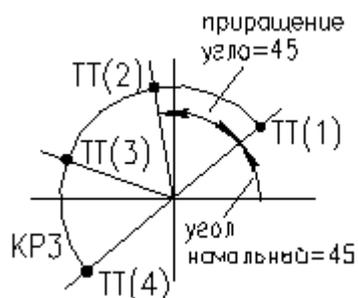
имя = *окружность*, { **ПОЧС** / **ПРЧС** }, *угол₁*, **ПОВОРОТ**, *угол₂*, *число*

Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
ПОЧС	Движение по окружности по часовой стрелке.
ПРЧС	Движение по окружности против часовой стрелки.
<i>угол₁</i>	Угол с осью X радиус-вектора начальной точки.
<i>угол₂</i>	Угол между радиусами, проходящими через соседние точки.
<i>число</i>	Число точек.

Пример:

ТТ=КР3, ПРЧС, 45, ПОВОРОТ, 45, 4



1.2.9.9 Все опорные точки контура

Формат:

$$\text{имя} = \left[\left\{ \begin{array}{l} \text{СЛЕВА} \\ \text{СПРАВА} \end{array} \right\}, \text{контур} [, \text{смещение}] \right]$$

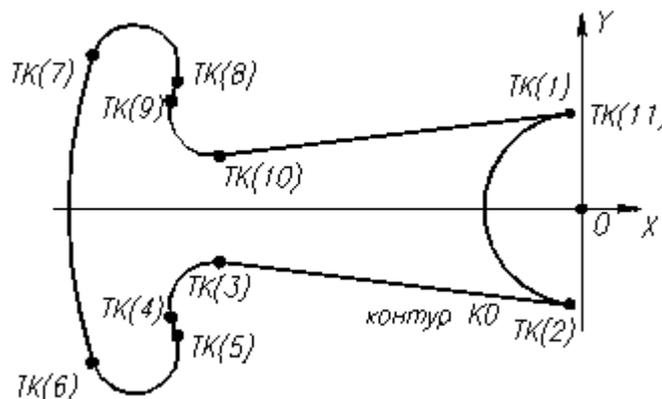
Параметры:

Параметр	Описание
<i>контур</i>	Контур.
СЛЕВА	Эквидистанта строится слева от контура.
СПРАВА	Эквидистанта строится справа от контура.
<i>смещение</i>	Величина смещения для построения эквидистанты к контуру.

 Порядок заполнения массива точек определяется направлением построения контура (значение индекса увеличивается по направлению обхода контура от его начала).

Пример:

ТК=K0
ТОЧКА ТК(12)=СЛЕВА,K1,2



1.2.9.10 Массив точек, упорядоченных контуром

Формат:

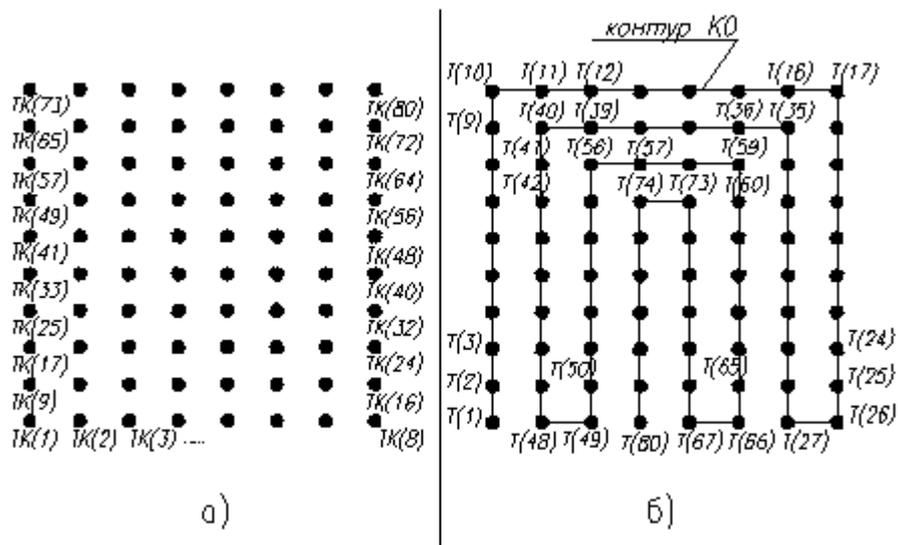
$$\text{имя} = \text{массив точек} \left[\begin{array}{l} \text{ПО} \\ \text{ОБРАТН} \end{array} \right], \text{контур}$$

Параметры:

Параметр	Описание
<i>массив точек</i>	Массив точек.
ПО	Обход контура в прямом направлении.
ОБРАТН	Обход контура в обратном направлении.
<i>контур</i>	Контур.

Пример:

T=TK, ПО, K0



1.2.9.11 Включение точек в массив точек

Формат:

$\text{имя} = \text{массив точек}_1, \text{ПЛЮС}, \left\{ \begin{array}{l} \text{массив точек}_2 \\ \text{точка} \\ \text{координаты} \end{array} \right\}$

Параметры:

Параметр	Описание
<i>массив точек₁</i>	Массив точек, в который происходит включение.
<i>массив точек₂</i>	Добавляемый массив точек.
<i>точка</i>	Добавляемая точка, заданная геометрическим объектом.

Параметр	Описание
<i>координаты</i>	Добавляемая точка, заданная своими координатами.

1.2.9.12 Исключение точек из массива точек

Формат:

$$\text{имя} = \text{массив точек}_1, \text{МИНУС}, \left. \begin{array}{l} \text{массив точек}_2 \\ \text{точка} \\ \text{координаты} \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>массив точек₁</i>	Исходный массив точек.
<i>массив точек₂</i>	Исключаемый массив точек.
<i>точка</i>	Исключаемая точка, заданная геометрическим объектом.
<i>координаты</i>	Исключаемая точка, заданная своими координатами.

1.2.9.13 Преобразование координат точек

Формат:

$$\text{имя} = \left. \begin{array}{l} \text{точка} \\ \text{координаты} \\ \text{массив точек} \end{array} \right\}, \text{матрица}, \text{число}$$

Параметры:

Параметр	Описание
<i>точка</i>	Точка, заданная геометрическим объектом.
<i>координаты</i>	Точка, заданная своими координатами.
<i>массив точек</i>	Массив точек.
<i>матрица</i>	Матрица преобразования.
<i>число</i>	Число повторений.

Пример:

ТЧА=ТЧ1, МАТР(ПЕРЕНОС, 50, -12), 6

1.2.9.14 Перенос массива точек в каждую точку другого массива

Формат:

ИМЯ = *массив точек*₁, ПЕРЕНОС, *массив точек*₂ [, [ЦЕНТР,] *точка*] [, ПОВОРОТ, *угол*]

Параметры:

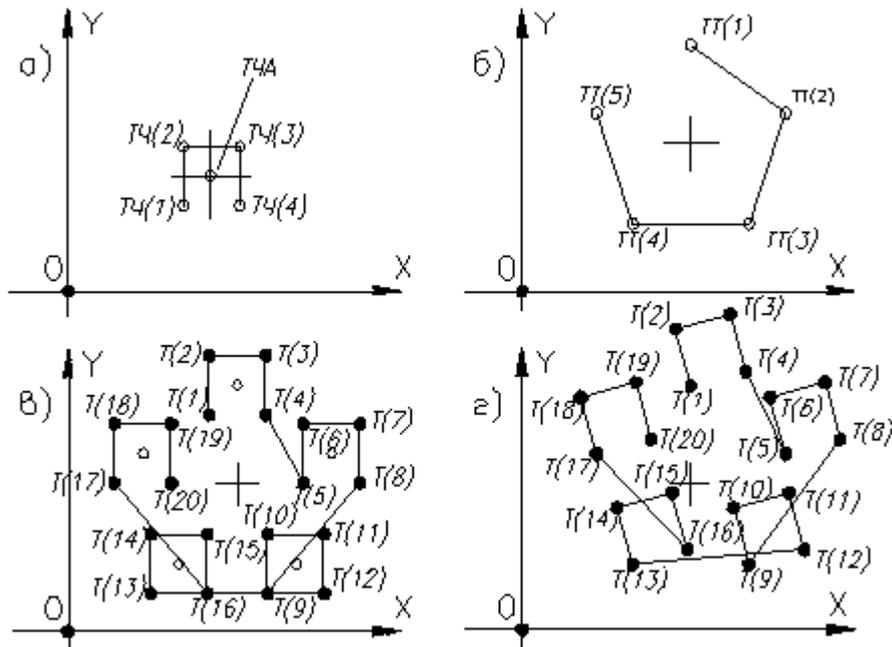
Параметр	Описание
<i>массив точек</i> ₁	Переносимый массив точек.
<i>массив точек</i> ₂	Массив точек, в который производится перенос.
<i>точка</i>	Базовая точка переноса.
<i>угол</i>	Угол поворота.

Результатом такого построения является массив, в котором на месте каждой точки массива *точек*₁, находятся точки из массива *точек*₂. По умолчанию массив₂ совмещается с точками массива₁ своей начальной точкой. Если же задана точка привязки, то при размещении точек, она совмещается с текущей точкой массива₁. Если задан угол, то после привязки к точке происходит разворот массива₂.

Пример:

Т=ТЧ, ПЕРЕНОС, ТТ, ЦЕНТР, ТЧА в)

Т=ТЧ, ПЕРЕНОС, ТТ, ПОВОРОТ, 20 г)



1.2.9.15 Отсечение точек массива ограниченной областью

Формат:

$имя = массив\ точек, \left\{ \begin{array}{l} \text{ВНЕ} \\ \text{ВНУТРИ} \\ \text{ПЕРЕСЕЧ} \end{array} \right\}, \left\{ \begin{array}{l} \text{контур} \\ \text{массив\ контуров} \end{array} \right\}$

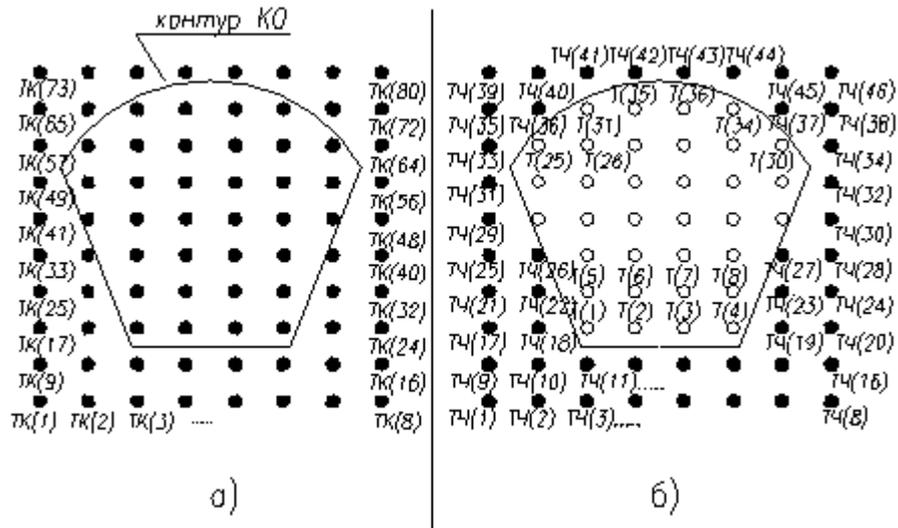
Параметры:

Параметр	Описание
<i>массив точек</i>	Массив точек.
<i>контур</i>	Замкнутый контур.
<i>массив контуров</i>	Массив контуров.

Значение индекса массива точек определяется направлением построения контура (увеличивается по направлению обхода контура от его начала).

Пример:

- T=TK, ВНУТРИ, K0** a)
- TЧ=TK, ВНЕ, K0** б)



1.2.9.16 Точки пересечения контура и геометрического объекта

Формат:

$$\text{имя} = \left\{ \begin{array}{l} \text{ПЕРЕСЕЧ} \\ \text{СЛЕВА} \\ \text{СПРАВА} \end{array} \right\} \text{контур, объект} [, \text{смещение}]$$

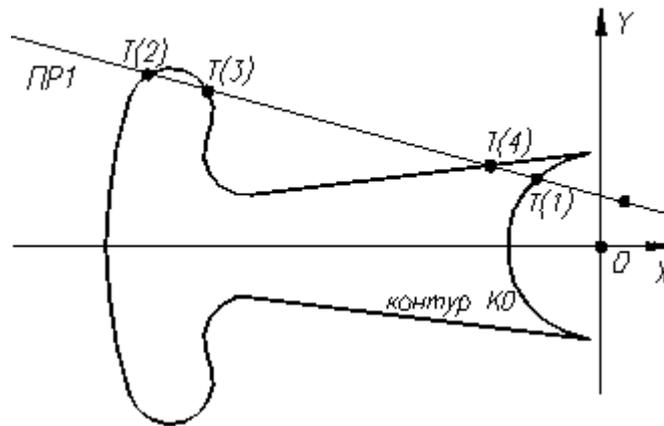
Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>объект</i>	Геометрический объект
<i>смещение</i>	Величина смещения для построения эквидистанты к контуру.

Пример:

ТОЧКА ТЧ19()=ПЕРЕСЕЧ,К0,ПРЯМАЯ(К1,4)

ТОЧКА ТЧ1()=СЛЕВА,К0,К1,4



1.2.9.17 Точки пересечения геометрических объектов

Формат:

имя = ПЕРЕСЕЧ,	}	прямая	}	прямая
		окружность		окружность
		контур		контур
		массив прямых		массив прямых
		массив окружностей		массив окружностей
		массив контуров		массив контуров

Параметры:

Параметр	Описание
контур	пересекаемые объекты
прямая	
окружность	
массив прямых	
массив окружностей	
массив контуров	

1.2.9.18 Точки в узлах прямоугольной сетки, заполняющие прямоугольную область

Формат:

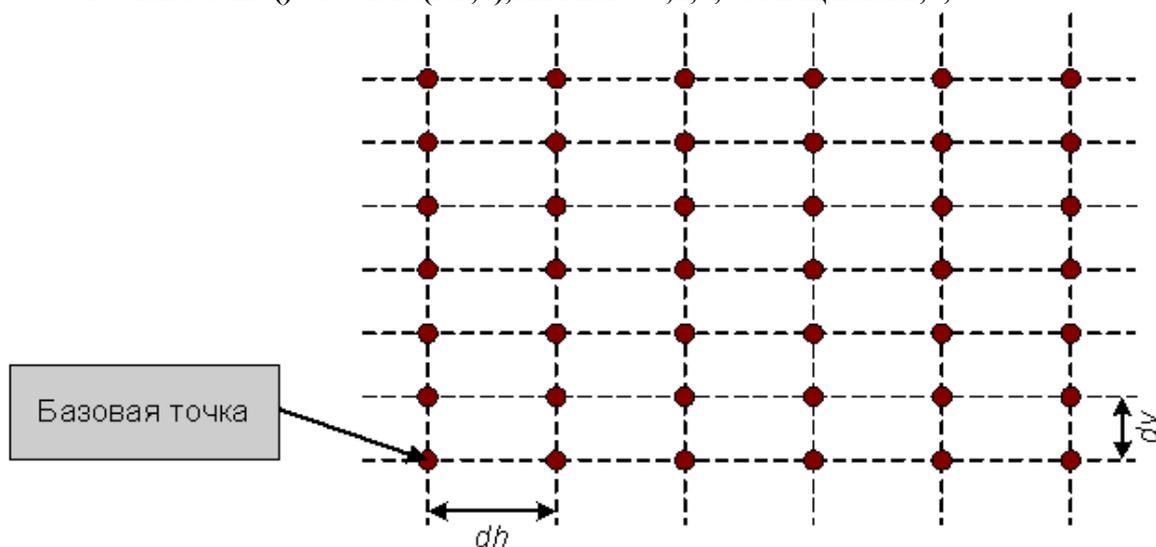
имя = точка, ПЕРЕНОС, шагX, шагY [, СМЕЩЕНИЕ], числоX, числоY

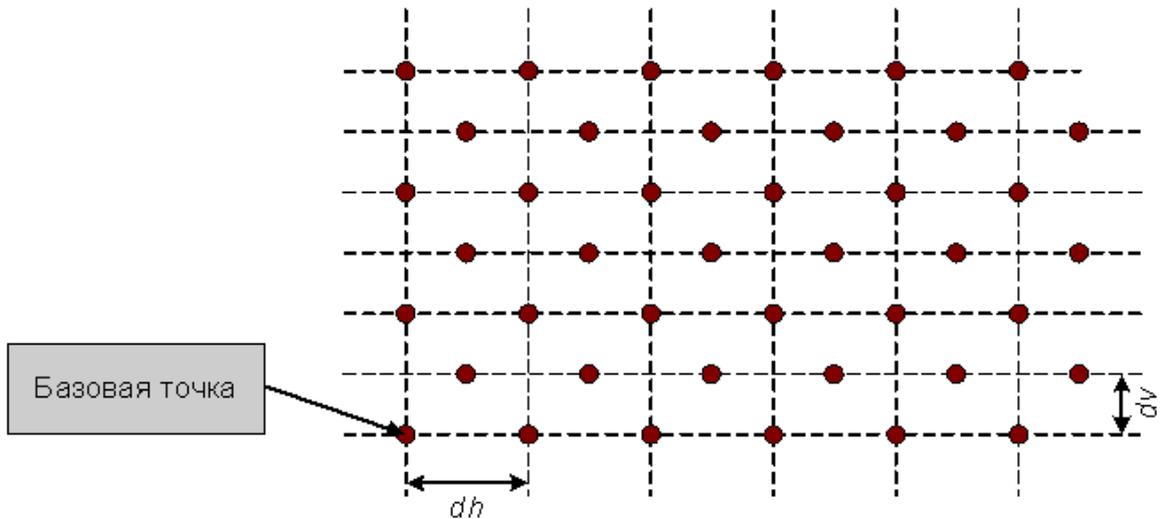
Параметры:

Параметр	Описание
<i>точка</i>	Базовая точка сетки
<i>шагX</i>	Шаг сетки по горизонтали
<i>шагY</i>	Шаг сетки по вертикали
<i>числоX</i>	Число столбцов
<i>числоY</i>	Число строк
СМЕЩЕНИЕ	Смещение четных рядов по горизонтали на половину шага.

Пример:

ТОЧКА ТЧ20)=ТОЧКА(К1,0),ПЕРЕНОС,2,4,СМЕЩЕНИЕ,3,5





1.2.9.19 Точки в узлах прямоугольной сетки, заполняющие ограниченную область

Формат:

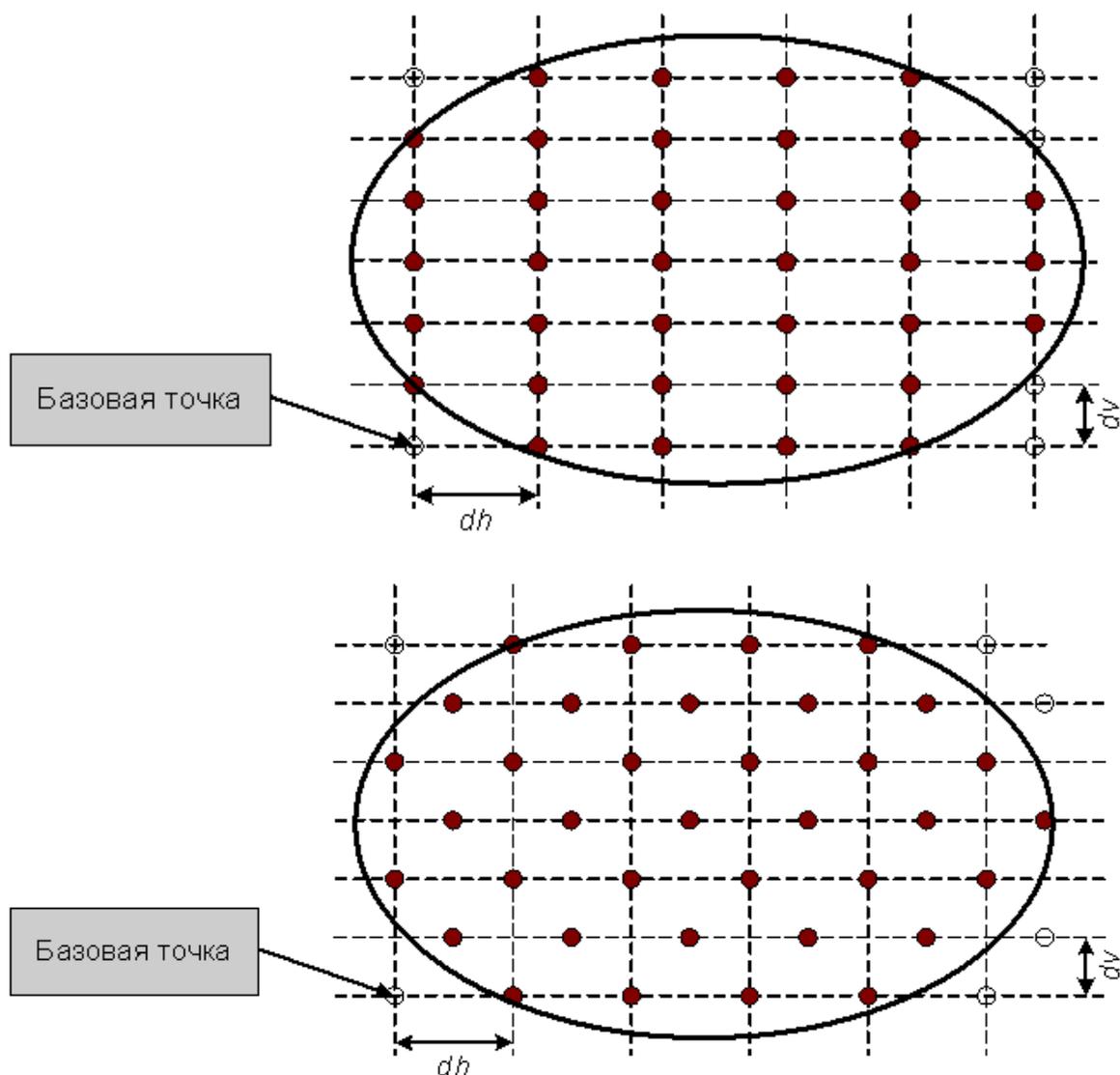
имя = *точка*, ПЕРЕНОС, *шагХ*, *шагУ* [, СМЕЩЕНИЕ], { *контур*
массив контуров }

Параметры:

Параметр	Описание
<i>точка</i>	Базовая точка
<i>шагХ</i>	Шаг по горизонтали
<i>шагУ</i>	Шаг по вертикали
<i>контур</i>	Контур, определяющий область.
<i>массив контуров</i>	Массив контуров, определяющий область.
СМЕЩЕНИЕ	Смещение четных рядов по горизонтали на половину шага.

Пример:

ТОЧКА ТЧ20(0)=ТОЧКА(К1,0),ПЕРЕНОС,2,4,СМЕЩЕНИЕ,3,5



1.2.10 Массивы прямых

В определениях массива прямых используются следующие группы модификаторов.

- Модификаторы выбора направления:
 - **ХБ** – смещение в сторону большей координаты **X**;
 - **УБ** – смещение в сторону большей координаты **Y**;
 - **ХМ** – смещение в сторону меньшей координаты **X**;
 - **УМ** – смещение в сторону меньшей координаты **Y**.
- Модификаторы выбора оси:
 - **ХПАР** – ось **X**;
 - **УПАР** – ось **Y**.

Темы этого раздела:

- [Заданное число прямых, параллельных данной прямой](#)^[133]
- [Прямые, проходящие через все точки массива под заданным углом](#)^[134]
- [Прямые, соединяющие все точки массива с заданной точкой](#)^[135]
- [Прямые, определяющие все отрезки контура](#)^[135]
- [Прямые, ограничивающие контур и параллельные прямой](#)^[136]
- [Прямые, ограничивающие контур и параллельные координатной оси](#)^[137]

1.2.10.1 Заданное число прямых, параллельных данной прямой

Формат:

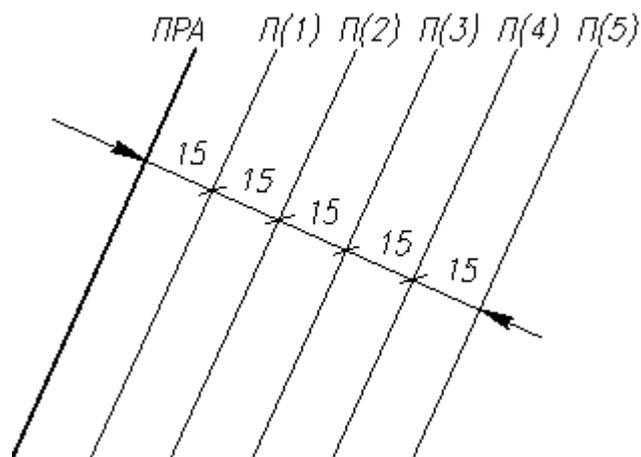
$$\text{имя} = \text{ПАРЛЕЛ}, \text{прямая}, \text{расстояние}, \left\{ \begin{array}{l} \text{ХБ} \\ \text{ХМ} \\ \text{УБ} \\ \text{УМ} \end{array} \right\}, \text{число}$$

Параметры:

Параметр	Описание
<i>прямая</i>	Прямая.
<i>расстояние</i>	Расстояние между прямыми.
ХБ	Смещение прямых в сторону большей координаты X.
УБ	Смещение прямых в сторону большей координаты Y.
ХМ	Смещение прямых в сторону меньшей координаты X.
УМ	Смещение прямых в сторону меньшей координаты Y.
<i>число</i>	Число прямых.

Пример:

ПРЯМАЯ П(5)
П=ПАРЛЕЛ, ПРА, 15, ХБ, 5



1.2.10.2 Прямые, проходящие через все точки массива под заданным углом

Формат:

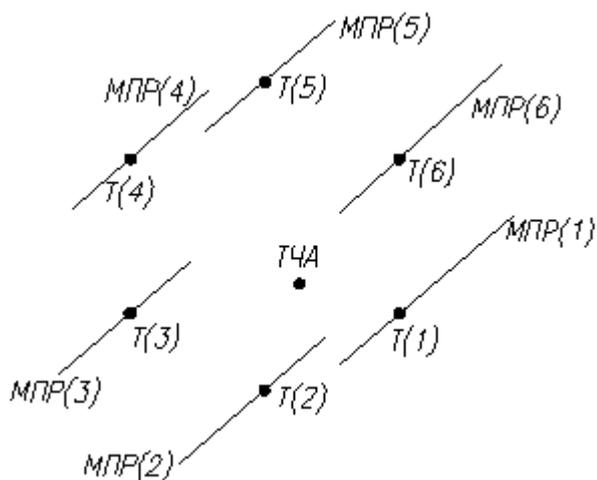
имя = массив точек, угол

Параметры:

Параметр	Описание
<i>массив точек</i>	Массив точек.
<i>угол</i>	Угол между прямой и осью X.

Пример:

МПР=Т, 40



1.2.10.3 Прямые, соединяющие все точки массива с заданной точкой

Формат:

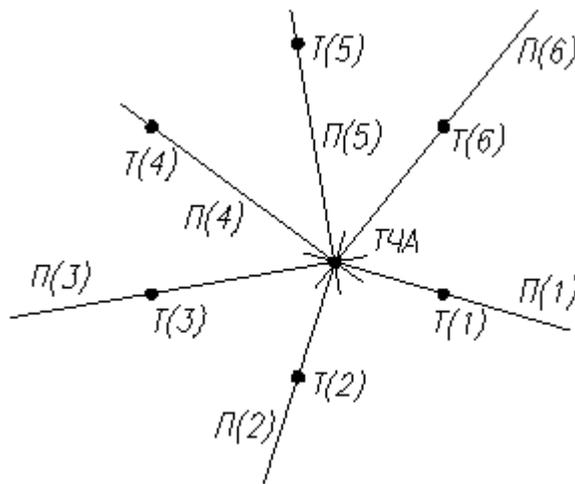
```
имя = массив точек, точка
```

Параметры:

Параметр	Описание
<i>массив точек</i>	Массив точек.
<i>точка</i>	Точка.

Пример:

П=Т, ТЧА



1.2.10.4 Прямые, определяющие все отрезки контура

Формат:

```
имя = контур
```

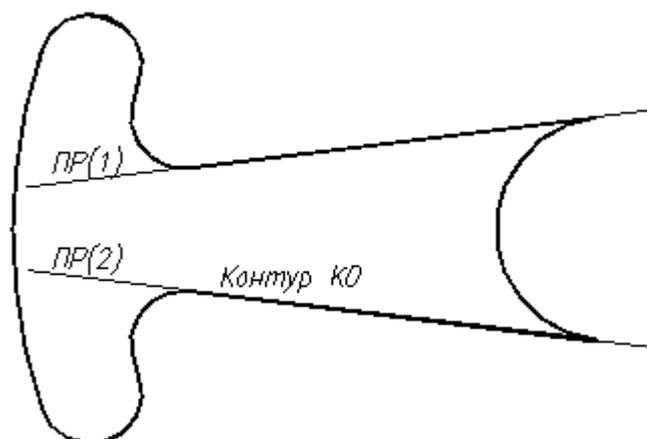
Параметры:

Параметр	Описание
<i>контур</i>	Контур.

 Порядок заполнения массива прямых определяется направлением построения контура (значение индекса увеличивается по направлению обхода контура от его начала).

Пример:

ПР=К0



1.2.10.5 Прямые, ограничивающие контур и параллельные прямой

Формат:

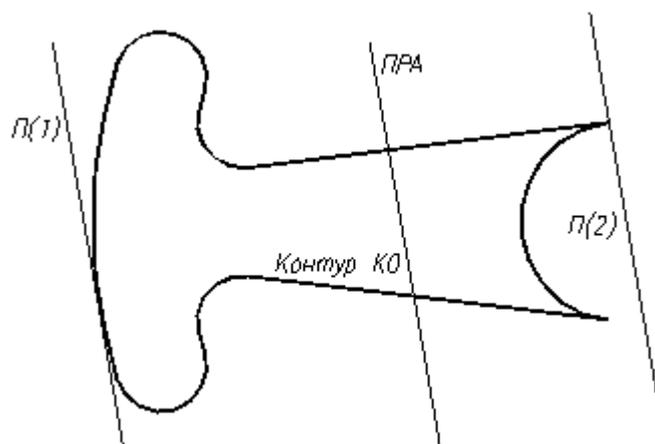
имя = контур, прямая

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>прямая</i>	Прямая.

Пример:

П=КО, ПРА



1.2.10.6 Прямые, ограничивающие контур и параллельные координатной оси

Формат:

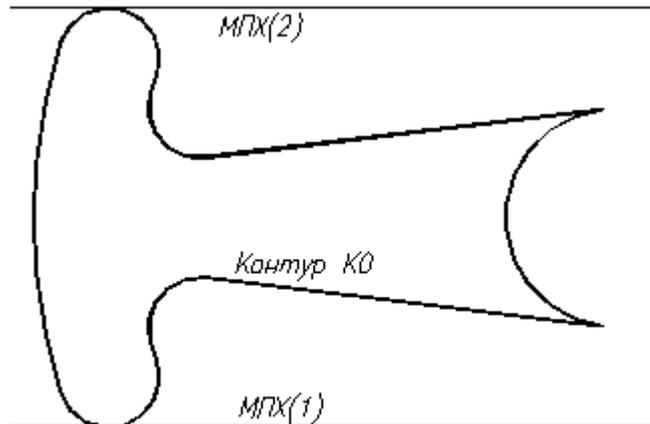
$$\text{имя} = \left\{ \begin{array}{l} \text{ХПАР} \\ \text{УПАР} \end{array} \right\}, \text{контур}$$

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
ХПАР	Ось координат X.
УПАР	Ось координат Y.

Пример:

МПХ=ХПАР, К0



1.2.10.7 Прямые, равномерно расположенные по всем направлениям

Формат:

$$\text{имя} = \text{точка}, \left\{ \begin{array}{l} \text{ПОЧС} \\ \text{ПРЧС} \end{array} \right\}, \left\{ \begin{array}{l} \text{точка} \\ \text{угол} \end{array} \right\}, \text{число}$$

Параметры:

Параметр	Описание
<i>точка</i>	точка, через которую проходят все прямые

Параметр	Описание
ПОЧС ПРЧС	направление следования углов, задающих положение прямых
<i>точка_н</i>	точка, задающая положение первой прямой
<i>угол</i>	положение первой прямой
<i>число</i>	число прямых

1.2.10.8 Прямые, равномерно расположенные в диапазоне углов

Формат:

$$\text{имя} = \text{точка}, \left\{ \begin{array}{l} \text{ПОЧС} \\ \text{ПРЧС} \end{array} \right\}, \left\{ \begin{array}{l} \text{точка}_n \\ \text{угол}_n \end{array} \right\}, \left\{ \begin{array}{l} \text{точка}_k \\ \text{угол}_k \end{array} \right\}, \text{число} [\text{,ПО}]$$

Параметры:

Параметр	Описание
<i>точка</i>	точка, через которую проходят все прямые
ПОЧС ПРЧС	направление следования углов, задающих положение прямых
<i>точка_н</i>	точка, задающая положение первой прямой
<i>угол_н</i>	положение первой прямой
<i>точка_к</i>	точка, задающая положение последней прямой
<i>угол_к</i>	положение последней прямой
<i>число</i>	число прямых
ПО	используется, если необходимо задать диапазон от угла до точки (т.е. <i>угол_н</i> - <i>точка_к</i>). В случае задания диапазона от точки до угла параметр не нужен.

1.2.10.9 Прямые, расположенные с заданным угловым интервалом

Формат:

$$\text{имя} = \text{точка}, \left\{ \begin{array}{l} \text{ПОЧС} \\ \text{ПРЧС} \end{array} \right\}, \left\{ \begin{array}{l} \text{точка}_n \\ \text{угол}_n \end{array} \right\}, \text{ПОВОРОТ}, \text{угол}, \text{число}$$

Параметры:

Параметр	Описание
<i>точка</i>	точка, через которую проходят все прямые
ПОЧС ПРЧС	направление следования углов, задающих положение прямых
<i>точка_n</i>	точка, задающая положение первой прямой
<i>угол_n</i>	положение первой прямой
<i>угол</i>	угловой интервал между соседними прямыми
<i>число</i>	число прямых

1.2.11 Массивы окружностей

В определениях массива окружностей используются следующие модификаторы:

- **МЕНШ** – выбор окружностей меньшего радиуса;
- **БОЛШ** – выбор окружностей большего радиуса.

Темы этого раздела:

- [Окружности, концентричные данной с указанным интервалом](#)¹³⁹
- [Окружности заданного радиуса с центрами в точках массива](#)¹⁴⁰
- [Окружности, задающие все дуги контура](#)¹⁴¹

1.2.11.1 Заданное число окружностей, концентричных данной с указанным интервалом

Формат:

$$\text{имя} = \text{ПАРЛЕЛ}, \left\{ \begin{array}{l} \text{БОЛШ} \\ \text{МЕНШ} \end{array} \right\}, \text{окружность}, \text{расстояние}, \text{число}$$

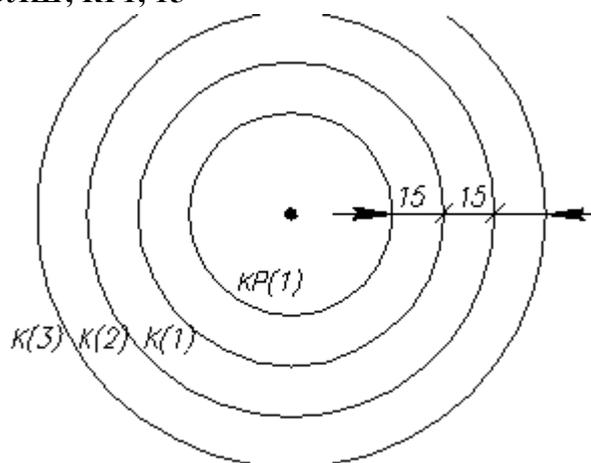
Параметры:

Параметр	Описание
<i>окружность</i>	Окружность.
<i>расстояние</i>	Расстояние между окружностями.
<i>число</i>	Число окружностей
МЕНШ	Выбор окружностей меньшего радиуса.
БОЛШ	Выбор окружностей большего радиуса.

Пример:

ОКРУЖН К()

К=ПАРЛЕЛ, БОЛШ, КР1, 15



1.2.11.2 Окружности заданного радиуса с центрами в точках массива

Формат:

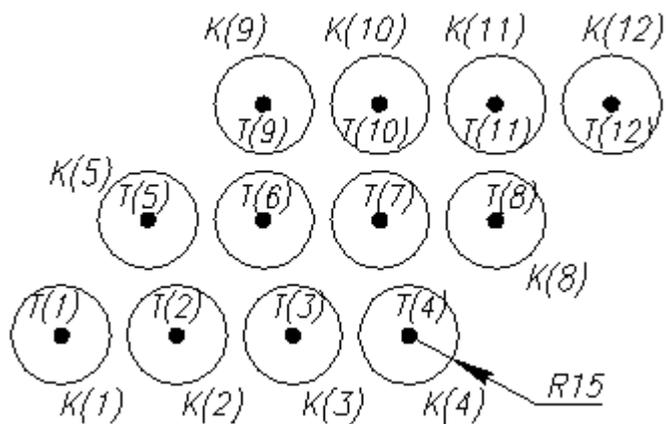
имя = массив точек, радиус

Параметры:

Параметр	Описание
<i>массив точек</i>	Массив точек (центров окружностей).
<i>радиус</i>	Радиус.

Пример:

К=Т, 15



1.2.11.3 Окружности, задающие все дуги контура

Формат:

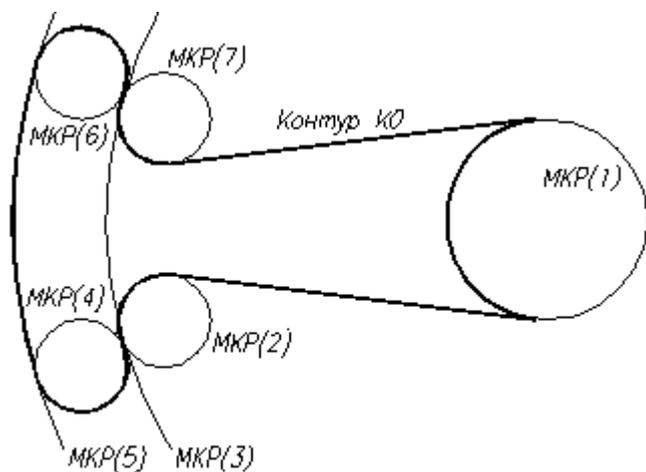
имя = контур

Параметры:

Параметр	Описание
<i>контур</i>	Контур.

Пример:

МКР=K0



1.2.12 Массивы контуров

В определениях массива контуров используются следующие модификаторы:

- **ВНЕ** – выбор контуров вне опорного контура;
- **ВНУТРИ** – выбор контуров внутри опорного контура;
- **СЛЕВА** – выбор контуров слева от опорного контура;
- **СПРАВА** – выбор контуров справа от опорного контура.

Темы этого раздела:

- Семейство внутренних эквидистант к замкнутому контуру^[142]
- Копирование контура с преобразованием координат^[143]
- Контур, полученные переносом контура заданное число раз^[144]
- Контур, полученные переносом контура во все точки массива с поворотом на заданный угол^[146]
- Заданное число последовательно построенных эквидистант к контуру^[147]
- Массив контуров, заданный текстом^[148]
- Текст, расположенный по дуге^[149]
- Объединение областей^[153]
- Исключение области из области^[153]
- Построение границ следа контура^[154]

1.2.12.1 Семейство внутренних эквидистант к замкнутому контуру

Формат:

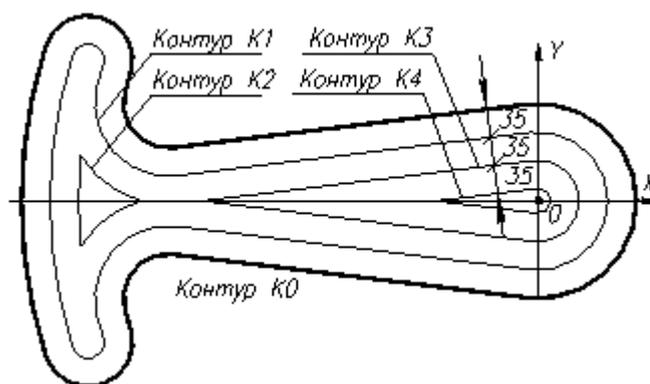
```
имя = контур, расстояние
```

Параметры:

Параметр	Описание
<i>контур</i>	Замкнутый контур.
<i>расстояние</i>	Расстояние между эквидистантами.

Пример:

КОНТУР К()=К0, 35



1.2.12.2 Контуры, полученные преобразованием контура заданное число раз

Формат:

$$\text{имя} = \left\{ \begin{array}{l} \text{контур} \\ \text{массив контуров} \end{array} \right\}, \text{ матрица, число}$$

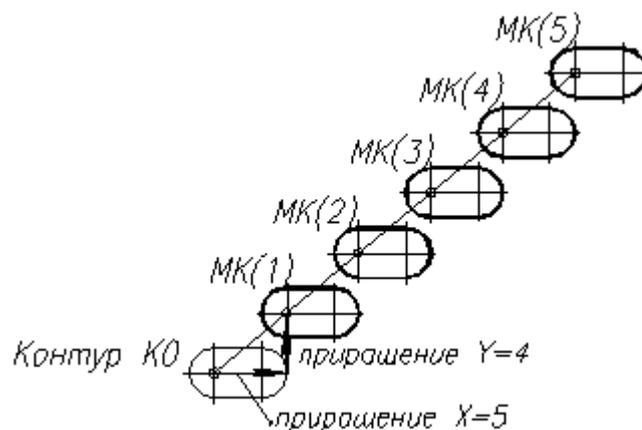
Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>массив контуров</i>	Массив контуров.
<i>матрица</i>	Матрица преобразования.
<i>число</i>	число преобразований.

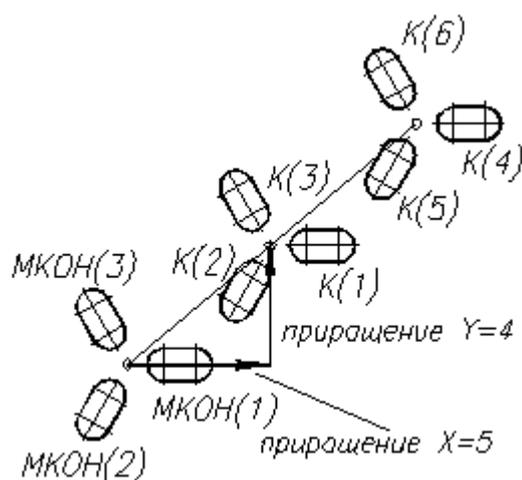
Пример:

МАТР МП=ПЕРЕНОС, 5, 4

МК=K0, МП, 5



МАТР МП=ПЕРЕНОС, 5, 4
К=МКОН, МП, 2



1.2.12.3 Контур, полученные переносом контура заданное число раз

Формат:

$$\text{имя} = \left\{ \begin{array}{l} \text{контур} \\ \text{массив контуров} \end{array} \right\}, \text{ПЕРЕНОС}, \text{приращение}_x, \text{приращение}_y, \text{число}$$

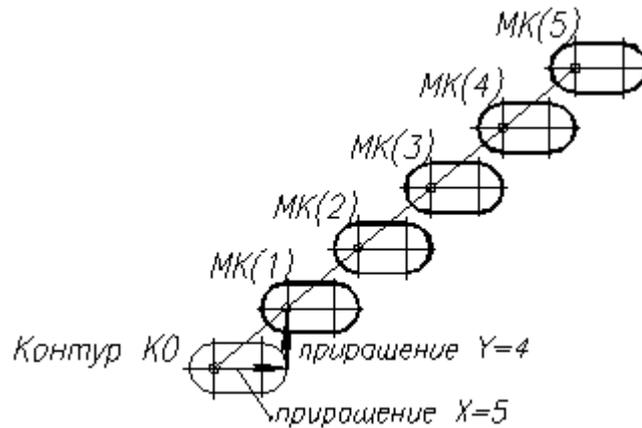
Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>массив контуров</i>	Массив контуров.
<i>приращение_x</i>	Приращение по оси X.

Параметр	Описание
<i>приращение_y</i>	Приращение по оси Y.
<i>число</i>	Число повторений.

Пример:

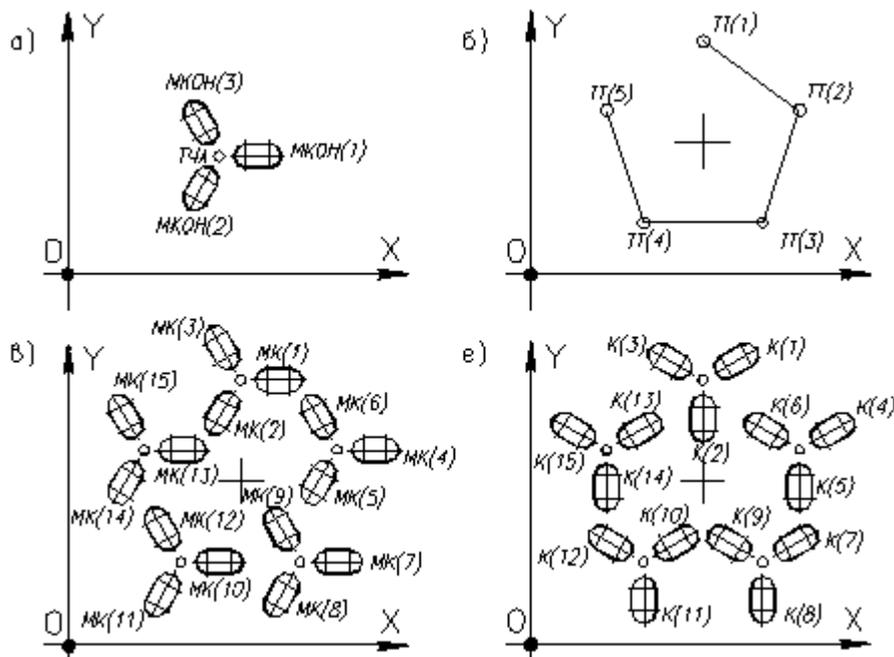
МК=K0, ПЕРЕНОС, 5, 4, 5



В данном примере массив ТЧ (рис. а) размещается в каждой точке массива ТТ (рис б) двумя способами (рис в и г).

МК=МКОН, ПЕРЕНОС, ТТ, ЦЕНТР, ТЧА

К=МКОН, ПЕРЕНОС, ТТ, ЦЕНТР, ТЧА, ПОВОРОТ, 30g



1.2.12.4 Контур, полученные переносом контура во все точки массива с поворотом на заданный угол

Формат:

$$\text{имя} = \left\{ \begin{array}{l} \text{контур} \\ \text{массив контуров} \end{array} \right\}, \text{ПЕРЕНОС, массив точек, ЦЕНТР, точка [, ПОВОРОТ, угол]}$$

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>массив контуров</i>	Массив контуров.
<i>массив точек</i>	Массив точек.
<i>точка</i>	Базовая точка переноса. Перенос контура происходит таким образом, что базовая точка оказывается в точках массива.
<i>угол</i>	Угол поворота.

По умолчанию, исходный контур включается в определяемый массив контуров без поворота, а точкой привязки является начало контура – т.е. исходный контур копируется в каждой точке массива точек и совмещается с ней своим началом.

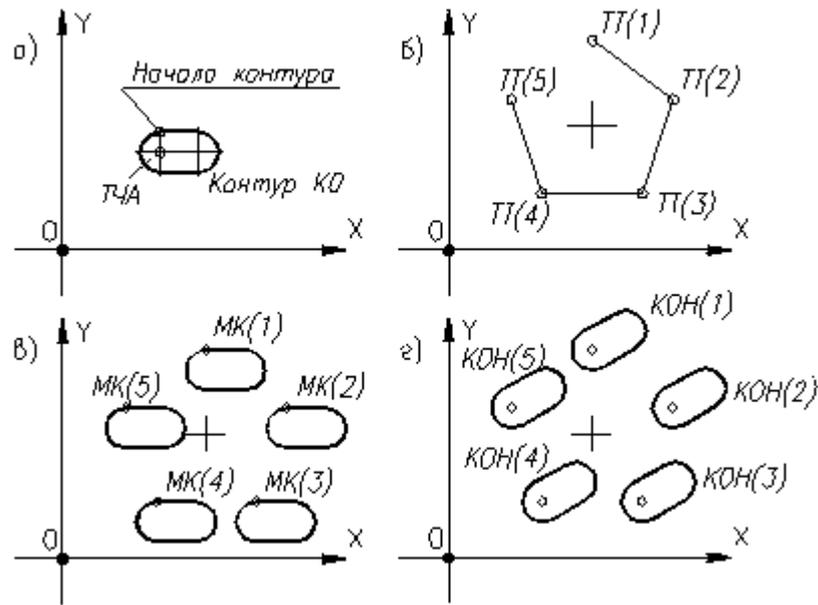
Пример:

В данном примере массив **ТЧ** (рис. а) размещается в каждой точке массива **ТТ** (рис б) двумя способами (рис в и г).

МК=К0, ПЕРЕНОС, ТТ

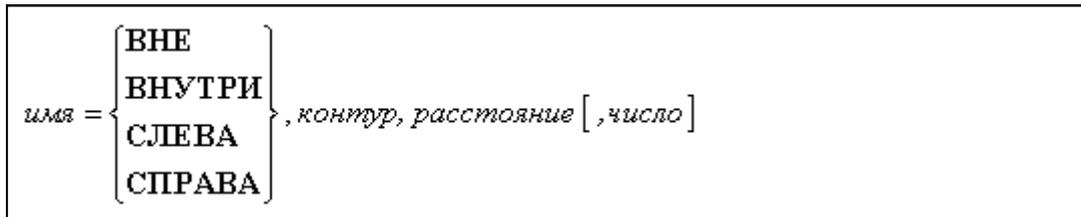
в)

КОН=К0, ПЕРЕНОС, ТТ, ЦЕНТР, ТЧА, ПОВОРОТ, 40 г)



1.2.12.5 Заданное число последовательно построенных эквидистант к контуру

Формат:



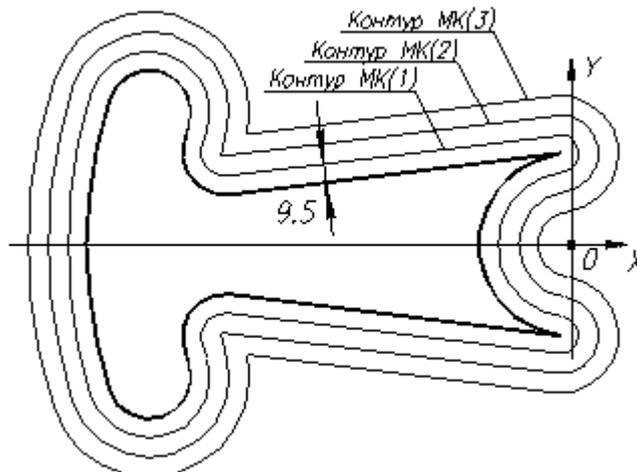
Параметры:

Параметр	Описание
ВНЕ	Эквидистанта расположена вне контура.
ВНУТРИ	Эквидистанта расположена внутри контура.
СЛЕВА	Эквидистанта расположена слева от контура.
СПРАВА	Эквидистанта расположена справа контура.
<i>контур</i>	Замкнутый контур.
<i>расстояние</i>	Расстояние между эквидистантами.
<i>число</i>	Заданное число эквидистант.

Если число эквидистант не задано, то строится только одна эквидистанта.

Пример:

МК=ВНЕ, К0, 9.5, 3



1.2.12.6 Массив контуров, заданный текстом

Формат:

$$\text{имя} = \text{текст}, \text{точка} [, \text{ПОВОРОТ}, \text{угол}], \left\{ \begin{array}{l} \text{СЛЕВА} \\ \text{СПРАВА} \\ \text{ЦЕНТР} \end{array} \right\}, \left[\begin{array}{l} \text{ВЕРХН} \\ \text{НИЖН} \end{array} \right]$$

Параметры:

Параметр	Описание
<i>текст</i>	Строка исходного текста.
<i>точка</i>	Базовая точка, относительно которой производится выравнивание
ПОВОРОТ , <i>угол</i>	Угол поворота текста.
ЦЕНТР	Выравнивание: по середине
СЛЕВА	Горизонтальное выравнивание: по левому краю
СПРАВА	Горизонтальное выравнивание: по правому краю
ВЕРХН	Выравнивание по горизонтали: по верху
НИЖН	Выравнивание по горизонтали: по низу

Пример:

КОНТУР НАДПИСЬ() = 'Техтран', ТЧНАЧ

Параметры шрифта должны быть предварительно заданы оператором **ШРИФТ**.

Формат:

ШРИФТ *название* [, **КУРСИВ**] [, **ПОЛУЖИРН**], **РАЗМЕР**, *размер*,
 [, **СЖАТИЕ**, *сжатие*] [, **НАКЛОН**, *угол*]

Параметры:

Параметр	Описание
<i>название</i>	Название шрифта.
КУРСИВ	Курсив.
ПОЛУЖИРН	Полужирный.
РАЗМЕР , <i>размер</i>	Размер шрифта в миллиметрах.
СЖАТИЕ , <i>сжатие</i>	Коэффициент сжатия букв по горизонтали (<i>сжатие</i> = 2 – в два раза 'стройнее'). По умолчанию <i>сжатие</i> = 1.
НАКЛОН , <i>угол</i>	Угол наклона букв. По умолчанию <i>угол</i> = 0.

Пример:

ШРИФТ 'Arial Cyr', РАЗМЕР, 30

1.2.12.7 Текст, расположенный по дуге

Формат:

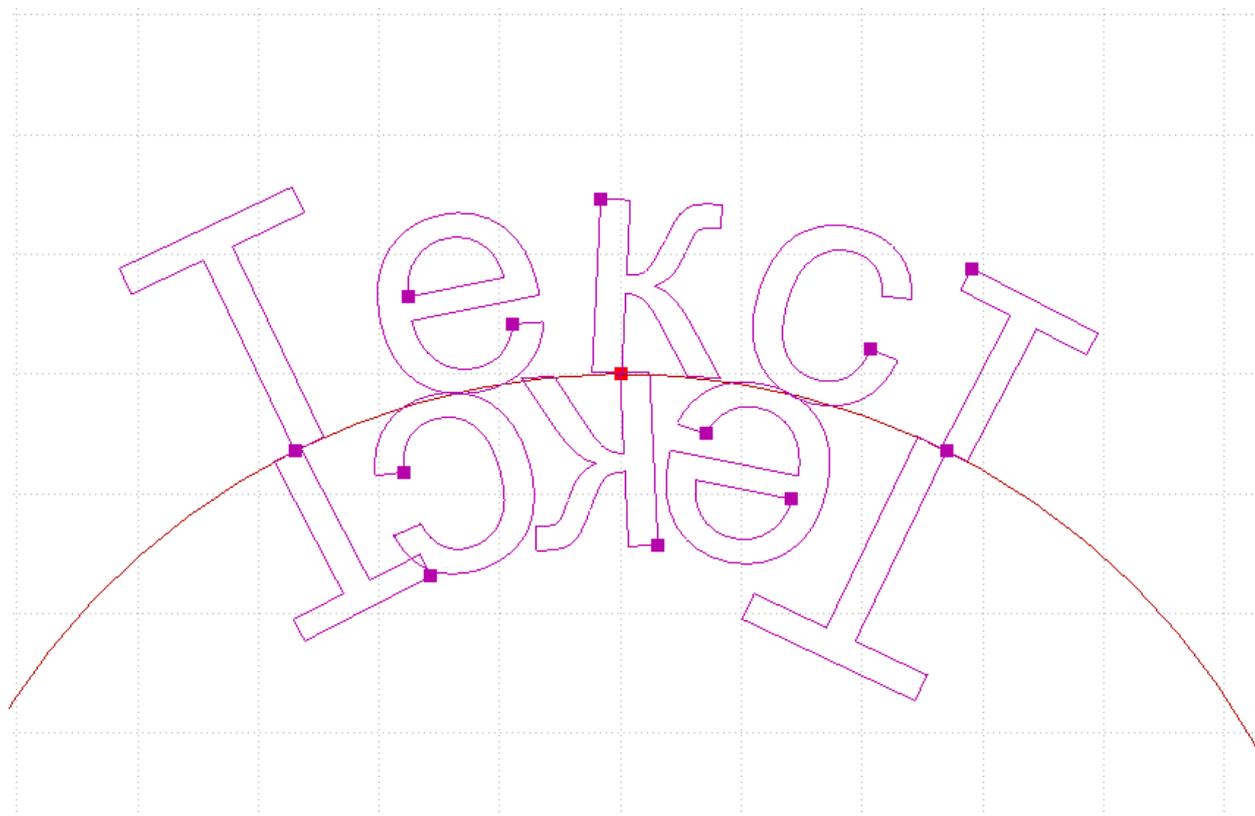
имя = *текст*, *точка*, *центр*, { **ПОЧС** }, { **СЛЕВА** }, { **СПРАВА** }, { **ВЕРХН** },
 { **ПРЧС** }, { **ЦЕНТР** }, { **НИЖН** }

Параметры:

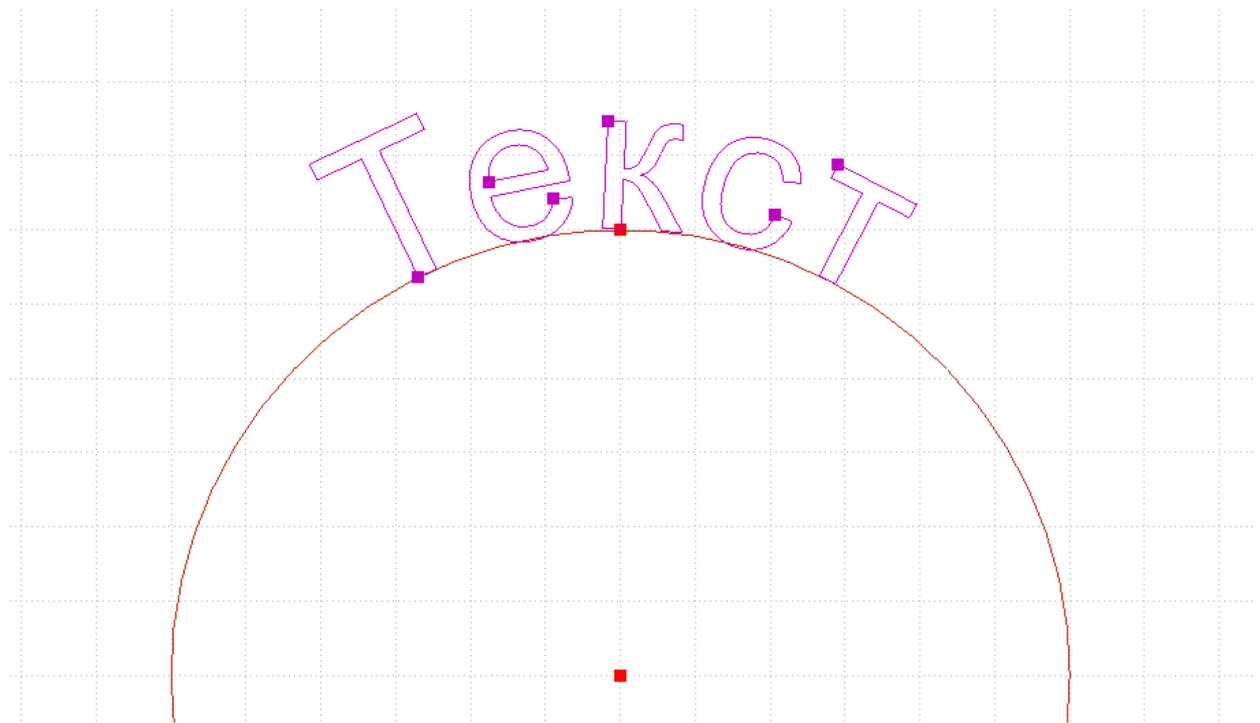
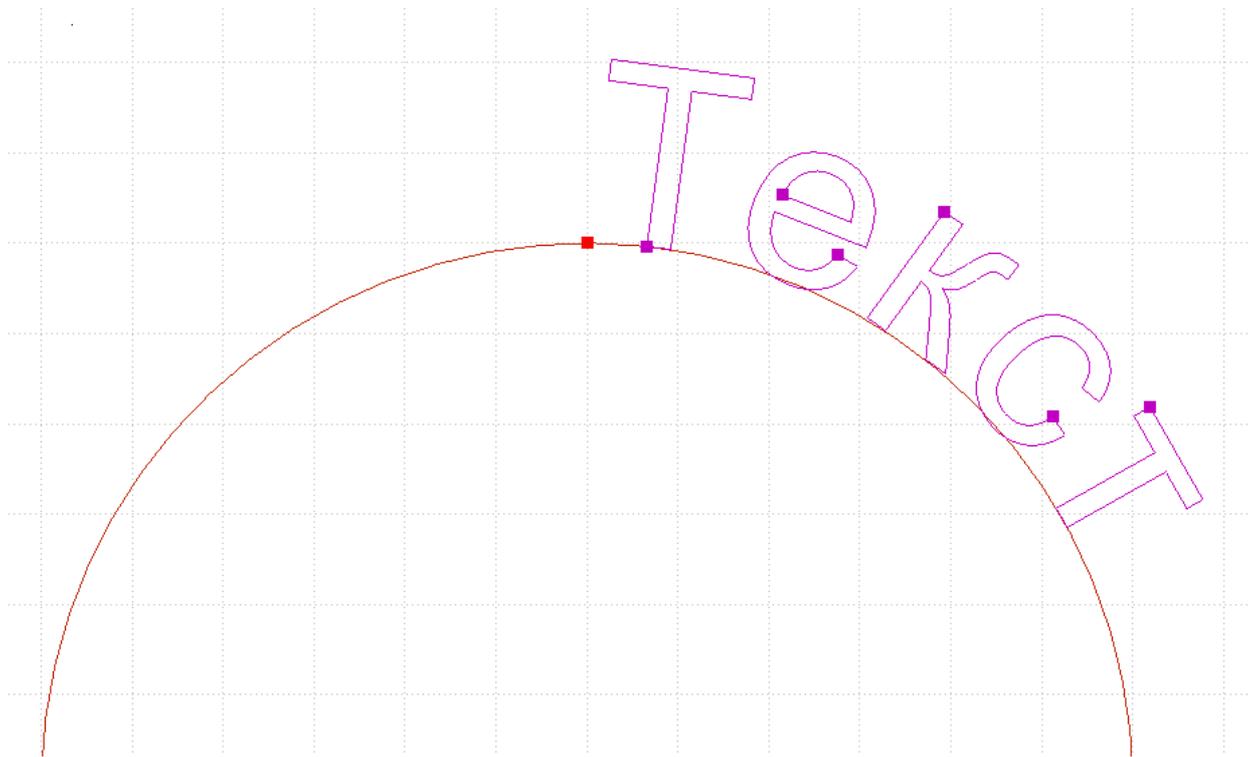
Параметр	Описание
<i>текст</i>	Строка исходного текста.
<i>точка</i>	Базовая точка, относительно которой производится выравнивание
<i>центр</i>	Точка центра дуги, по которой располагается текст.
ПОЧС	Расположение текста на окружности по часовой стрелке

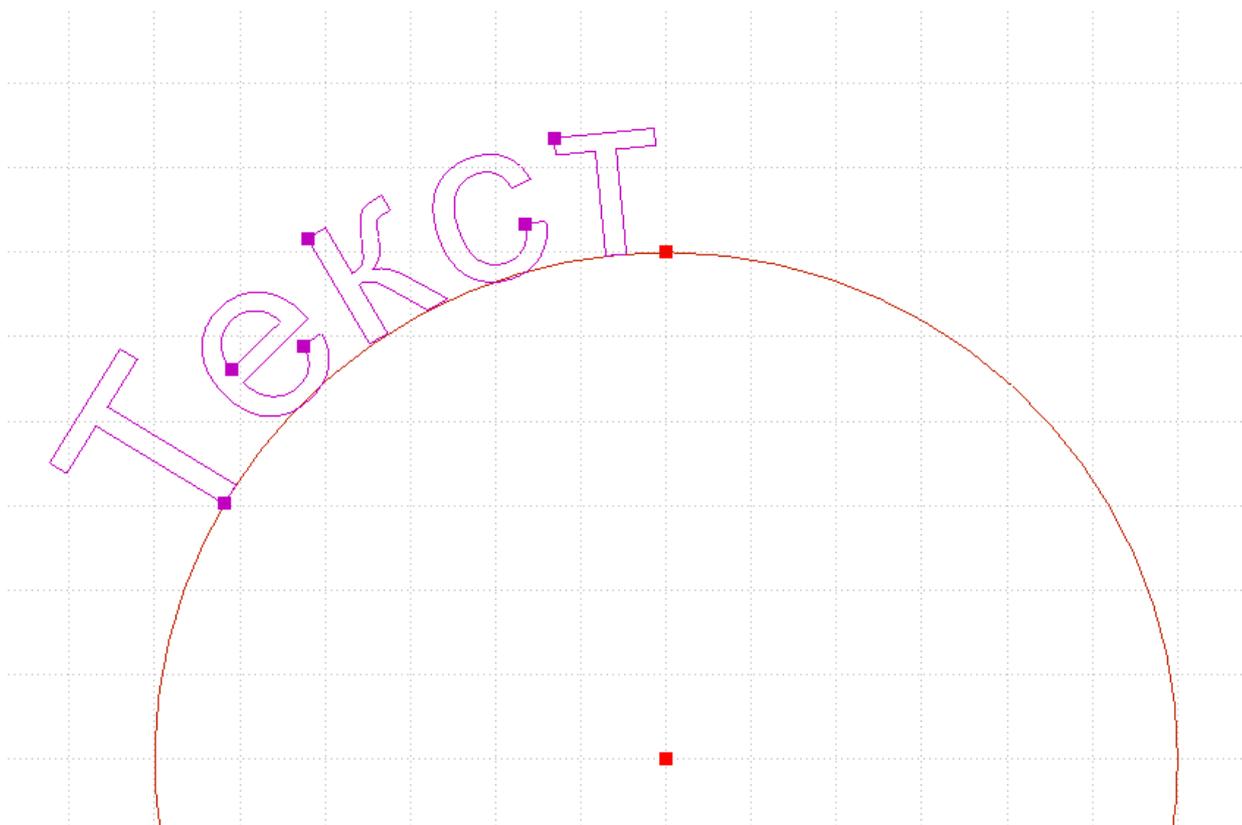
Параметр	Описание
ПРЧС	Расположение текста на окружности против часовой стрелки
ЦЕНТР	Выравнивание: по середине
СЛЕВА	Горизонтальное выравнивание: по левому краю
СПРАВА	Горизонтальное выравнивание: по правому краю
ВЕРХН	Выравнивание по горизонтали: по верху
НИЖН	Выравнивание по горизонтали: по низу

Направление надписи



Выравнивание по горизонтали





Пример:

КОНТУР K8()='Техтран',ТЧ2,ТЧ3,ПОЧС,СЛЕВА,ВЕРХН

Параметры шрифта должны быть предварительно заданы оператором **ШРИФТ**.

Формат:

ШРИФТ *название* [, **КУРСИВ**] [, **ПОЛУЖИРН**] , **РАЗМЕР** , *размер* ,
 [, **СЖАТИЕ** , *сжатие*] [, **НАКЛОН** , *угол*]

Параметры:

Параметр	Описание
<i>название</i>	Название шрифта.
КУРСИВ	Курсив.
ПОЛУЖИРН	Полужирный.
РАЗМЕР , <i>размер</i>	Размер шрифта в миллиметрах.
СЖАТИЕ , <i>сжатие</i>	Коэффициент сжатия букв по горизонтали (<i>сжатие</i> = 2 – в два раза 'стройнее'). По умолчанию <i>сжатие</i> = 1.
НАКЛОН , <i>угол</i>	Угол наклона букв. По умолчанию <i>угол</i> = 0.

Пример:

ШРИФТ 'Arial Cyr', РАЗМЕР, 30

1.2.12.8 Объединение областей

Формат:

$$\text{имя} = \text{ПЛЮС}, \left\{ \begin{array}{l} \text{контур} \\ \text{массив контуров} \end{array} \right\}, \left\{ \begin{array}{l} \text{контур} \\ \text{массив контуров} \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>массив контуров</i>	Массив контуров.

Пример:

КОНТУР К2()=ПЛЮС,К0,К1

1.2.12.9 Исключение области из области

Формат:

$$\text{имя} = \text{МИНУС}, \left\{ \begin{array}{l} \text{контур} \\ \text{массив контуров} \end{array} \right\}, \left\{ \begin{array}{l} \text{контур} \\ \text{массив контуров} \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>массив контуров</i>	Массив контуров.

Пример:

КОНТУР К2()=МИНУС,К0,К1

1.2.12.10 Пересечение областей

Формат:

$$\text{имя} = \text{ПЕРЕСЕЧ}, \left\{ \begin{array}{l} \text{контур} \\ \text{массив контуров} \end{array} \right\}, \left\{ \begin{array}{l} \text{контур} \\ \text{массив контуров} \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>контур</i>	Контур.
<i>массив контуров</i>	Массив контуров.

Пример:

КОНТУР К2()=ПЕРЕСЕЧ,К0,К1

1.2.12.11 Построение границ следа контура

Формат:

имя = СЛЕД, контур, контур, точка

Параметры:

Параметр	Описание
<i>контур</i>	Траектория, по которому производится перемещение
<i>контур</i>	Перемещаемый контур.
<i>точка</i>	Базовая точка перемещения, которой контур перемещается по траектории (точка может не принадлежать перемещаемому контуру)

Пример:

КОНТУР К2()=СЛЕД,К0,К1,ТОЧКА(0,0)

1.3 Описание инструмента

Для описания инструмента предназначен оператор ИНСТР, который позволяет задавать:

- геометрические параметры инструмента, необходимые для расчета траектории его движения относительно контура и поверхности детали;
- технологические данные, необходимые при формировании УП (вылеты, номера корректоров, позиции в инструментальном магазине и т.д.).

Оператор состоит из нескольких компонент, содержащих перечисленную информацию.

Формат:

ИНСТР *имя = описание инструмента [,примечание]*
[,аппаратная головка] [,описание корректоров]

Параметры:

Параметр	Описание
<i>имя</i>	Идентификатор имени инструмента.
<i>описание инструмента</i>	Описание параметров инструмента.
<i>примечание</i>	Примечание к инструменту.
<i>аппаратная головка</i>	Описание устройства крепления инструмента.
<i>описание корректоров</i>	Описание корректоров, связанных с инструментом.

 Оператор **ИНСТР** имеет описательный характер: геометрические и технологические параметры инструмента становятся доступны системе только после применения операторов выбора и загрузки инструмента;

Для совместимости с предыдущими версиями системы сохраняется формат оператора **ИНСТР**, задающий только геометрические параметры инструмента:

ИНСТР диаметр $\begin{bmatrix} 0 \\ \text{радиус} \end{bmatrix}$

Параметры:

Параметр	Описание
<i>диаметр</i>	Диаметр инструмента.
<i>радиус</i>	Радиус скругления режущей кромки.

 1. В случае использования этого формата не требуется применения операторов выбора и загрузки инструмента.

2. Отсутствие оператора **ИНСТР** эквивалентно заданию инструмента нулевого диаметра.

При использовании оператора любого из двух форматов следует помнить:

- Радиус скругления не может быть больше радиуса цилиндрической части фрезы.
- Изменение диаметра инструмента запрещено, если после оператора **ИНСТР** следует оператор продолжения непрерывного движения.

Для каждого вида обработки описание инструмента имеет свою специфику.

1.4 Программирование перемещений инструмента

Траекторией движения инструмента является линия, по которой перемещается в пространстве геометрический центр инструмента. Траектория состоит из отдельных участков, обход которых производится в определенном порядке. В программе участки траектории представляются одномерным массивом **ХОД**, элементы которого расположены в порядке обработки.

В Техтроне существуют два типа движения:

- **поточечное движение**, при котором перемещение инструмента программируется прямым заданием либо конечной точки, либо величин приращений координат;
- **непрерывное движение**, при котором перемещение программируется перечислением элементов траектории инструмента (т.е. участков прямых, окружностей, частей контуров) с указанием направления движения.

Возможность задания негоризонтальной **поверхности детали** позволяет программировать обработку контуров и областей, имеющих наклонное дно, благодаря чему на Техтроне можно (с определенными ограничениями) описывать обработку деталей достаточно сложной пространственной формы.

Во всех операторах движения вместо имен геометрических объектов можно использовать вложенные определения объектов тех же типов. Если отсутствует координата **Z**, она считается равной нулю.

Темы этого раздела:

- [Участок траектории, построенный из элементов](#)^[156]
- [Траектория, полученная объединением участков траектории](#)^[192]
- [Копирование участка траектории с преобразованием координат](#)^[193]
- [Перенос участка траектории заданное число раз](#)^[193]
- [Перенос участка траектории во все точки массива точек](#)^[194]
- [Преобразование траектории движения инструмента](#)^[178]
- [Управление формированием эквидистанты и траектории](#)^[183]

1.4.1 Участок траектории, построенный из элементов

Участок траектории, построенный из элементов, состоит из последовательности операторов, начинающейся оператором **ИЗ**.

Формат:

```
[ХОД(индекс) =] ИЗ {точка}
                    {x,y[,z]}

оператор1
оператор2
.....
операторn-1
операторn
```

Параметры:

Параметр	Описание
<i>индекс</i>	Номер участка траектории в массиве ХОД . Может быть либо назначен явно, либо присваивается по умолчанию, если левая часть отсутствует.
<i>точка</i>	Начальная точка участка траектории.
<i>x, y, z</i>	Координаты точки.
<i>оператор_p, ... оператор_n</i>	Сегменты участка траектории, заданные операторами движения, фрагментами других контуров, специальные операторы.

Каждый *оператор_i* в описании участка траектории имеет формат:

```
{ оператор поточечного движения }
{ оператор непрерывного движения }
{ оператор постпроцессора }
{ вспомогательный оператор }
```

Границей участка траектории является другой оператор **ИЗ** или оператор **КОНЕЦ**.

Пример:

В примере приведены программы, содержащие по три участка траектории, каждый из которых начинается оператором **ИЗ**.

```
ДЕТАЛЬ                ДЕТАЛЬ
.
ИЗ ТЧА } ХОД(1)        .
.                ХОД(6)=ИЗ ТЧА } ХОД
.                .                (6)
.
```

ИЗ 0,12,200 . . ИЗ ТЧ122 . . КОНЕЦ	} } } }	ХОД(2) ХОД(3) КОНЕЦ	} } } }	ХОД(4)=ИЗ 0,12,200 . . ИЗ ТЧ122 . . КОНЕЦ	} } } }	ХОД (4) ХОД (7)
---------------------------------------------------------	------------------------------	-------------------------------------------	------------------------------	----------------------------------------------------------------	------------------------------	--------------------------------------

В программе слева показано назначение индексов по умолчанию – в порядке возрастания, с шагом, равным единице. В программе справа сочетается явное и неявное назначение индексов. Последний участок траектории имеет индекс, следующий за максимальным из уже назначенных. Элементы массива **ХОД(1)**, **ХОД(2)**, **ХОД(3)** и т.д. являются в этом примере неопределенными.

Следует помнить, что обработка постпроцессором элементов массива **ХОД** происходит в порядке возрастания индекса. Поэтому в программе справа фактическая последовательность обработки не соответствует последовательности операторов в правом столбце.

При построении траектории инструмента действуют следующие правила:

- Инструмент должен быть загружен (оператор **ЗАГРУЗ**) в начале программы
- после смены инструмента (оператор **РАЗГРУЗ**)
- после разбиения программы (оператор **РАЗБПР**)
- После каждой загрузки инструмента должно быть определено исходное положение оператором **ИЗ**
- при отсутствии явного задания исходного положения инструмента, оператор **ИЗ** формируется неявно исходя из того, что после загрузки инструмент находится в точке смены (оператор **ТЧКМЕН**)
- В качестве начала участка перемещения берется конец предыдущего.

Темы этого раздела:

- [Исходное положение инструмента](#)^[159]
- [Поточечное движение](#)^[159]
- [Непрерывное движение](#)^[162]
- [Сочетание поточечного и непрерывного движения](#)^[174]
- [Фиктивное движение](#)^[175]
- [Управление поверхностью детали](#)^[176]
- [Отводы и возвраты инструмента](#)^[177]

1.4.1.1 Исходное положение инструмента

Прежде чем программировать движение инструмента, необходимо задать его исходное положение. Для этого служит оператор **ИЗ**, который начинает построение следующего участка траектории.

Формат:

$$[\text{ХОД}(\text{индекс}) =] \text{ ИЗ } \left\{ \begin{array}{l} \text{точка} \\ x, y [, z] \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>индекс</i>	Номер участка траектории; может быть либо назначен явно, либо присваивается по умолчанию, если левая часть отсутствует.
<i>точка</i>	Точка.
<i>x, y, z</i>	Координаты точки.

Значение индекса по умолчанию назначается равным единице – для первого применения оператора **ИЗ** и возрастает с шагом **1** – для каждого последующего оператора **ИЗ**. При сочетании явного и неявного способов нумерации участков, на единицу наращивается максимальный имеющийся индекс.

Пример:

- ИЗ 10, 20**
- ИЗ А, Б, -250.35**
- ИЗ ТЧ1**
- ИЗ ТОЧКА (ПР1, ПР2)**
- ИЗ (X+Y)/2, (2*X-КВКОР(Y))/2**
- ХОД(2) = ИЗ ТЧ5**

Положение инструмента является неопределенным в начале программы, а также после операторов **РАЗГРУЗ** и **АБЗАЦ**. В этих случаях последующее построение траектории необходимо начать с оператора **ИЗ**. Если положение инструмента не определено, оператор **ИЗ** может быть сформирован неявно по точке смены инструмента. Если определена точка смены инструмента (оператор **ТЧКСМЕН**), считается, что исходное положение инструмента совпадает с точкой смены.

1.4.1.2 Поточечное движение

При программировании поточечного движения явно задается каждое перемещение инструмента либо приращениями координат относительно предыдущего положения, либо точкой результирующего положения центра (вершины) инструмента. Этот тип движения реализуется операторами **ВТОЧКУ** и **ПРИРАЩ**.

Темы этого раздела:

- [Движение в точку](#)^[160]
- [Движение, заданное приращением координат](#)^[161]

1.4.1.2.1 Движение в точку

Движение в точку задает оператор **ВТОЧКУ**. Он вызывает перемещение:

- по прямой;
- по дуге окружности;
- по всем точкам массива.

Формат:

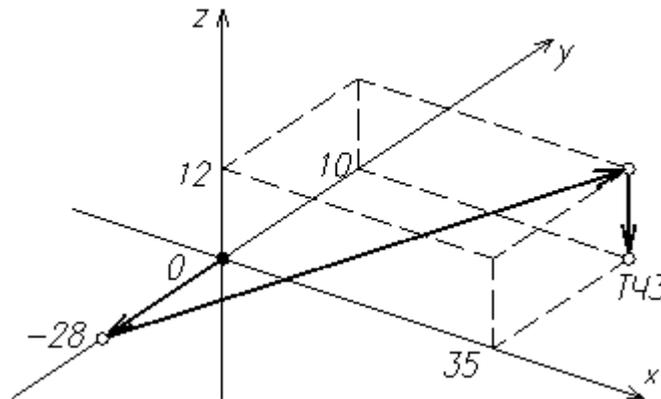
$$\text{ВТОЧКУ} \left\{ \begin{array}{l} \left\{ \begin{array}{l} \text{точка} \\ x, y [, z] \end{array} \right\} \left[\begin{array}{l} \left\{ \begin{array}{l} \text{ПОЧС} \\ \text{ПРЧС} \end{array} \right\}, \left\{ \begin{array}{l} \text{окружность} \\ X_c, Y_c, R \end{array} \right\} \end{array} \right] \right\} \\ \text{массив точек} \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>точка</i>	Точка.
<i>x, y, z</i>	Координаты точки.
ПОЧС	Движение по окружности по часовой стрелке.
ПРЧС	Движение по окружности против часовой стрелки.
<i>окружность</i>	Окружность.
<i>X_c, Y_c</i>	Координаты центра окружности.
<i>R</i>	Радиус окружности.
<i>массив точек</i>	Массив точек.

 Если оператор задает движение по дуге, то конечная точка должна лежать на окружности.

Пример:



ТЧЗ = 35,10
ИЗ 0, 0
ВТОЧКУ 0,-28
ВТОЧКУ 35, 10, 12
ВТОЧКУ ТЧЗ

1.4.1.2.2 Движение, заданное приращением координат

Движение, заданное приращением координат, вызывает оператор **ПРИРАЩ**. Он формирует перемещение относительно текущего положения в точку, заданную:

- вектором;
- приращениями по осям координат;

Формат:

$$\text{ПРИРАЩ} \left\{ \begin{array}{l} \text{вектор} \\ x, y [, z] \\ z \end{array} \right\}$$

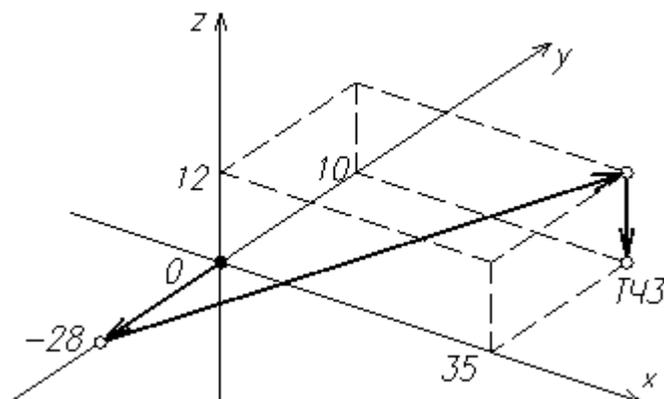
Параметры:

Параметр	Описание
<i>вектор</i>	Вектор.
<i>x, y, z</i>	Величины смещений относительно текущего положения.

Если в операторе **ПРИРАЩ** координаты **x**, **y** или **z** равны нулю, то перемещений по соответствующим осям не происходит. Поэтому оператор **ПРИРАЩ 0, 0, 50** эквивалентен оператору **ПРИРАЩ 50**.

При последовательном многократном использовании оператора **ПРИРАЩ** следует соблюдать осторожность, так как ошибка в одном из них влияет на все последующие положения инструмента.

Движения, изображенные на рисунке, можно описать при помощи оператора **ПРИРАЩ** следующим образом:



ИЗ 0, 0
ПРИРАЩ 0, -28, 0
ПРИРАЩ 35,(28+10), 12
ПРИРАЩ -12

При описании движения допускается сочетание операторов **ВТОЧКУ** и **ПРИРАЩ**:

ИЗ 0, 0
ПРИРАЩ 0, -28, 0
ВТОЧКУ 35, 10, 12
ПРИРАЩ -12

Допускается сокращенная запись операторов поточечного движения - замена одним оператором нескольких записанных подряд операторов поточечного движения. Операнды этих операторов разделяются в этом случае двоеточием. Можно чередовать различные формы задания точек (приращения), например:

ВТОЧКУ ТЧ1: 6.2, А + В, 0 : ТОЧКА (ПР1, ПР2) : ТЧ3
ПРИРАЩ 5, 1 : 0, 3*КОС (45) : ВЕКТ1

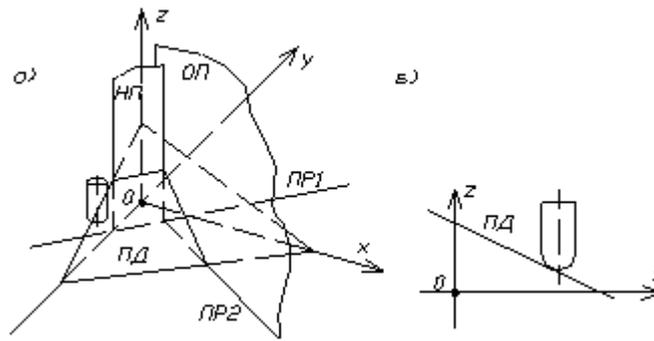
Следует иметь в виду, что сокращенная запись операторов движения затрудняет понимание диагностических сообщений, имеющих ссылку на номер строки.

1.4.1.3 Непрерывное движение

При программировании непрерывного движения на Техтроне описывается движение инструмента относительно трех поверхностей:

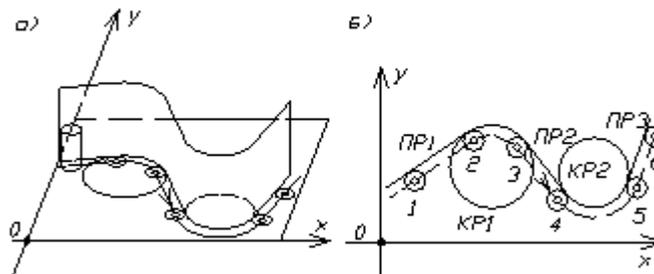
- - поверхности детали (*ПД*);
- - направляющей поверхности (*НП*);
- - ограничивающей поверхности (*ОП*).

Следующий рисунок поясняет эти понятия:



При непрерывном движении инструмент находится в постоянном контакте с поверхностью детали и направляющей поверхностью и своим торцом касается поверхности детали (см. рисунок б), а боковой поверхностью или осью касается направляющей поверхности. Поверхность детали представляет собой плоскость - горизонтальную или наклонную. По умолчанию ПД совпадает с плоскостью xOy . Конечное положение инструмента после выполнения очередного оператора непрерывного движения определяется ограничивающей поверхностью.

Следующий рисунок иллюстрирует применение направляющей и ограничивающей поверхностей:



Прямые и окружности, составляющие контур детали, представляют собой проекции НП и ОП на горизонтальную плоскость. В дальнейшем, для простоты изложения, НП и ОП отождествляются со своими проекциями на плоскость xOy . Так, для участков траектории, изображенной на рисунке б направляющей и ограничивающей поверхностями являются:

Участок траектории	НП	ОП
1-2	ПР1	КР1
2-3	КР1	ПР2

Участок траектории	НП	ОП
3-4	ПР2	КР2
4-5	КР2	ПР3
5-6	ПР3

Темы этого раздела:

- [Начало непрерывного движения](#)^[164]
- [Задание направления подхода](#)^[166]
- [Продолжение непрерывного движения](#)^[169]
- [Неявное определение ограничивающей поверхности](#)^[173]

1.4.1.3.1 Начало непрерывного движения

Начало непрерывного движения задается оператором **ИДИ**, который выводит инструмент в рабочее положение относительно управляющих поверхностей: *НП*, *ПД* и *ОП*. Положение инструмента относительно каждой из них указывается модификаторами **ДО**, **НА**, **ЗА**.

Формат:

$$\text{ИДИ} \left\{ \begin{array}{l} \text{ДО} \\ \text{НА} \\ \text{ЗА} \end{array} \right\} \text{НП} \left[\left\{ \begin{array}{l} \text{ДО} \\ \text{НА} \end{array} \right\} \text{ПД} \right] \left[\left\{ \begin{array}{l} \text{ДО} \\ \text{НА} \\ \text{ЗА} \end{array} \right\} \text{ОП} \right]$$

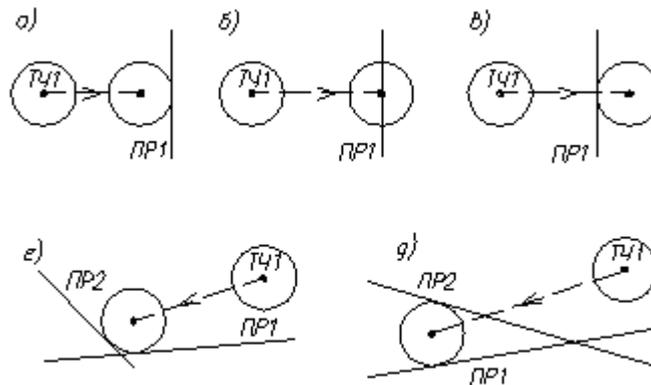
Параметры:

Параметр	Описание
<i>НП</i>	Направляющая поверхность.
<i>ПД</i>	Поверхность детали.
<i>ОП</i>	Ограничивающая поверхность.
ДО НА ЗА	Модификаторы, определяющие положение инструмента относительно управляющих поверхностей.

По оператору **ИДИ** инструмент выводится в требуемую точку по кратчайшему пути. Если *НП* и *ОП* имеют две точки пересечения, то выбирается ближайшая из них. В операторе **ИДИ** с двумя или тремя поверхностями в качестве *ОП* должна быть указана поверхность, которая будет направляющей в следующем операторе движения.

На следующий пример показывает применение модификаторов **ДО**, **НА**, **ЗА** при движении с одной (**а-в**), и с двумя поверхностями (**г-д**) соответственно.

ИЗ ТЧ1; ИДИ ДО ПР1	а)
ИЗ ТЧ1; ИДИ НА ПР1	б)
ИЗ ТЧ1; ИДИ ЗА ПР1	в)
ИЗ ТЧ1; ИДИ ДО ПР1 ДО ПР2	г)
ИЗ ТЧ1; ИДИ ДО ПР1 ЗА ПР2	д)

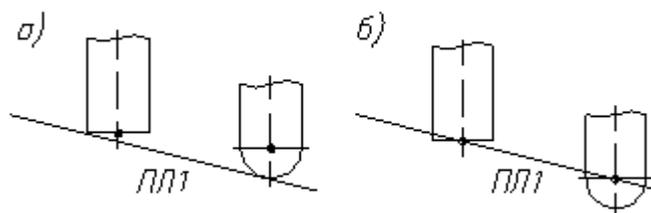


При необходимости направление подхода может указываться операторами **НАПРВ** - в направлении вектора или **НАПРТ** - в направлении точки.

Если в операторе **ИДИ** задана поверхность детали, то при непрерывном движении инструмент будет находиться в контакте с ней в соответствии с заданным модификатором. Команды поточечного движения (**ВТОЧКУ** и **ПРИРАЩ**), вызывающие отход от поверхности детали по координате **Z** будут игнорироваться. Отвести инструмент от **ПД** можно только после оператора **ПОВДЕТ ОТМЕН**.

Следующий пример показывает положение инструмента относительно **ПД**, заданное операторами:

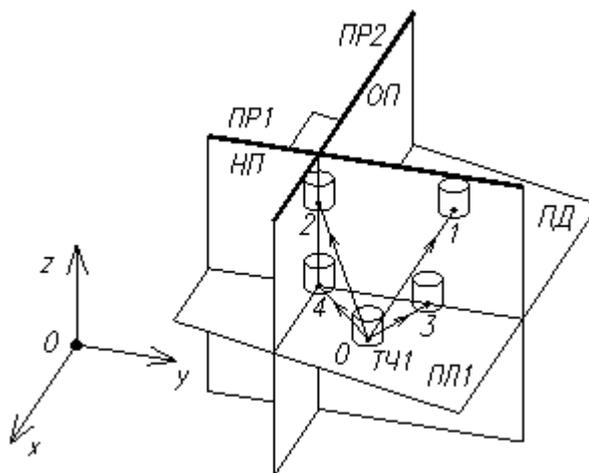
ИДИ НА ПРХ ДО ПЛ1 ЗА ПР3	а)
ИДИ НА ПРХ НА ПЛ1 ЗА ПР3	б)



Если же **ПД** в операторе **ИДИ** опущена, то поверхностью детали становится плоскость, параллельная плоскости **xOy** и отстоящая от нее на значение текущей координаты **Z**. При

этом разрешено перемещать инструмент по вертикальной оси.

Различие в использовании оператора **ИДИ** с одной, двумя и тремя поверхностями иллюстрирует следующий пример:



ИЗ ТЧ1; ИДИ ДО ПР1	<i>перемещение 0-1</i>
ИЗ ТЧ1; ИДИ ДО ПР1 ДО ПР2	<i>перемещение 0-2</i>
ИЗ ТЧ1; ИДИ ДО ПР1 ДО ПЛ1	<i>перемещение 0-3</i>
ИЗ ТЧ1; ИДИ ДО ПР1 ДО ПЛ1 ДО ПР2	<i>перемещение 0-4</i>

См. также:

- [Задание направления подхода](#)¹⁶⁶

1.4.1.3.2 Задание направления подвода

Направление подхода задается с помощью операторов **НАПРТ** и **НАПРВ**. Эти операторы являются вспомогательными и используются совместно с оператором **ИДИ** в следующих случаях:

- Если при использовании оператора **ИДИ** с одной поверхностью (кроме поверхности детали) надо осуществить подход не к ближайшей точке поверхности;
- Если при использовании оператора **ИДИ** с двумя поверхностями (не считая поверхности детали) необходимо выбрать одну из двух возможных точек пересечения **НП** и **ОП**;
- Если инструмент уже находится в требуемом положении относительно **НП**;
- Если в операторе **ИДИ** в качестве **НП** указан контур.

В первых двух случаях отсутствие оператора, задающего направление подхода, вызовет подход к ближайшей точке, а в остальных двух случаях приведет к ошибке.

Операторы **НАПРТ** и **НАПРВ** не управляют перемещением по третьей координате, т.е. направление подхода к **НП** определяется проекцией вектора подхода на горизонтальную плоскость.

Подход в направлении точки задает оператор **НАПРТ**.

Формат:

$$\text{НАПРТ} \left\{ \begin{array}{l} \text{точка} \\ x, y \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>точка</i>	Точка.
<i>x</i>	Координата X точки.
<i>y</i>	Координата Y точки.

Подход в направлении вектора задает оператор **НАПРВ**.

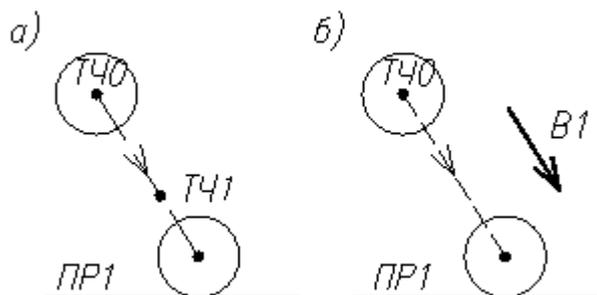
Формат:

$$\text{НАПРВ} \left\{ \begin{array}{l} \text{вектор} \\ x, y \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>вектор</i>	Вектор.
<i>x</i>	Проекция вектора на координатную ось X .
<i>y</i>	Проекция вектора на координатную ось Y .

Следующий пример иллюстрирует методы подхода к *НП* в направлении точки и вектора соответственно:



ИЗ ТЧ0

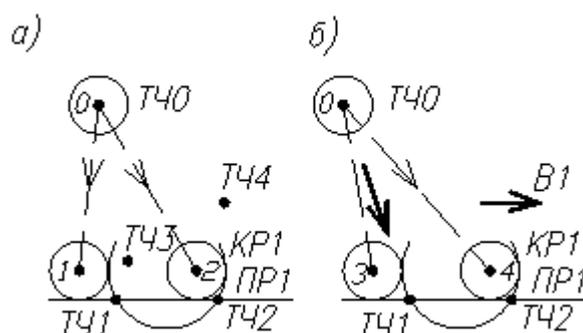
ИЗ ТЧ0

**НАПРТ ТЧ1
ИДИ ДО ПР1**

**НАПРВ В1
ИДИ ДО ПР1**

В случае, когда *НП* и *ОП* пересекаются в двух точках, точка подхода выбирается с помощью операторов **НАПРТ** и **НАПРВ**. Если используется **НАПРТ**, то из двух возможных положений выбирается ближайшее к точке, заданной в этом операторе. Если используется **НАПРВ**, то заданный в нем вектор используется для указания требуемого положения инструмента, так же как и модификаторы типа **ХБ-УМ** в геометрических определениях.

Пример:



Перемещение 0-1 (а). Оператор **ИДИ** задает переход в позицию 1, так как она ближе к **ТЧ3**, чем позиция 2.

**ИЗ ТЧ0
НАПРТ ТЧ3
ИДИ ДО ПР1 ДО КР1**

Перемещение 0-2 (а). Инструмент перемещается в позицию 2, так как она ближе к **ТЧ4**, чем позиция 1.

**ИЗ ТЧ0
НАПРТ ТЧ4
ИДИ ДО ПР1 ДО КР1**

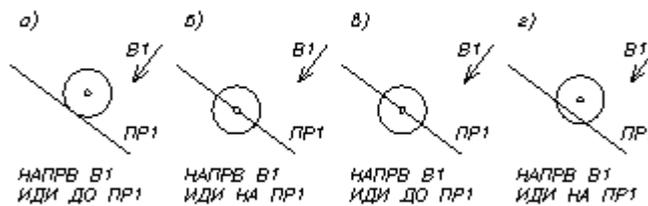
Перемещение 0-3 (б).

**ИЗ ТЧ0
НАПРВ ВЕКТОР (ТЧ0, ТЧ1)
ИДИ ДО ПР1 ДО КР1**

Перемещение 0-4 (б).

**ИЗ ТЧ0
НАПРВ В1
ИДИ ДО ПР1 ДО КР1**

Если инструмент уже подведен к *НП* (касается или пересекает ее), то для изменения его положения относительно *НП* или для задания направления, в соответствии с которым будет выбран оператор непрерывного движения, следует также использовать операторы **НАПРВ** или **НАПРТ**.



При этом запрещен вывод инструмента в направлении, встречном указанному в операторах **НАПРВ** или **НАПРТ** (е).

Если в операторе **ИДИ** в качестве *НП* указан контур или ход, то указание направления подхода является обязательным:

ИЗ ТЧ1
НАПРВ В1
ИДИ ДО КОНТ1

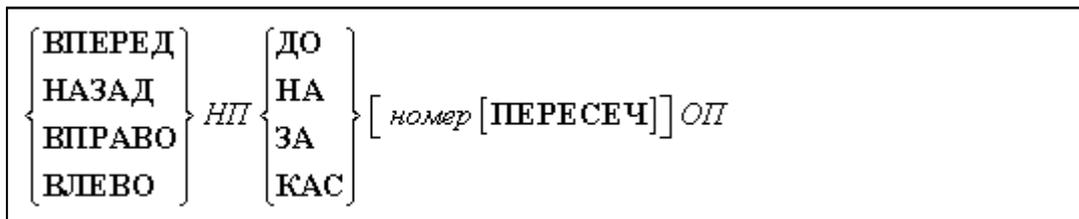
См. также:

- [Подход к контуру](#)^[229]

1.4.1.3.3 Продолжение непрерывного движения

После выхода на направляющую поверхность движение по каждому из геометрических элементов контура задается операторами **ВПЕРЕД**, **НАЗАД**, **ВЛЕВО**, **ВПРАВО**.

Формат:

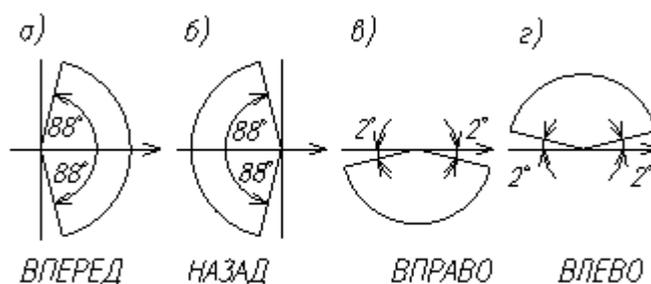


Параметры:

Параметр	Описание
ВПЕРЕД НАЗАД ВЛЕВО ВПРАВО	Направление перемещения инструмента по направляющей поверхности относительно предыдущего перемещения.

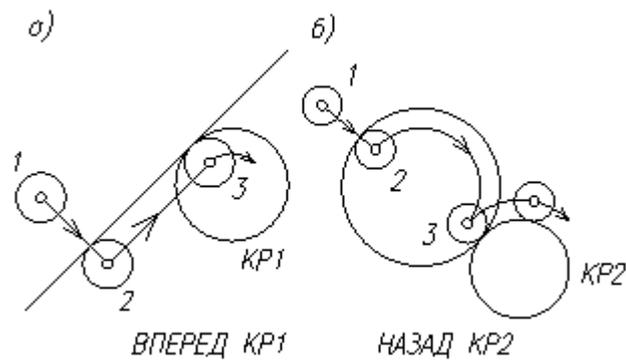
Параметр	Описание
<i>НП</i>	Направляющая поверхность.
<i>ОП</i>	Ограничивающая поверхность.
<i>номер</i>	Номер пересечения НП с ОП, до которого задано движение по НП.
ДО НА ЗА КАС	Положение инструмента относительно ОП (аналогично оператору ИДИ).

Направление движения выбирается относительно предыдущего перемещения как показано на рисунке:

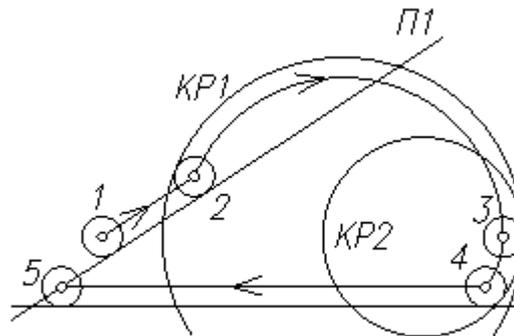


Секторами ограничены области действия служебных слов, указывающих направление перемещения.

При движении по окружности направлением движения считается направление касательной к окружности в точке пересечения *НП* и *ОП*. Поэтому в случае касания *НП* и *ОП* следует применять только операторы **ВПЕРЕД** и **НАЗАД**:



Пример программирования непрерывного движения.



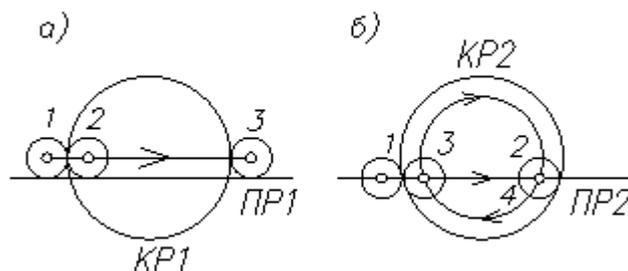
Перемещение инструмента из положения 1 в положение 5 задается следующими операторами.

ВПЕРЕД П1 ЗА KP1	1-2
ВЛЕВО KP1 КАС К2	2-3
<i>или</i> ВПЕРЕД KP1 КАС К2	
ВПЕРЕД KP2 ДО П2	3-4
ВПРАВО П2 НА П1	4-5

Поверхности, указанные в операторе непрерывного движения могут пересекаться и не один раз. Если не указывать номер пересечения, то инструмент будет выведен на (до, за) первое пересечение *НП* и *ОП*. Поэтому, когда требуется остановиться на последующих пересечениях, следует указывать номер пересечения *НП* с *ОП*.

Номер пересечения отсчитывается для каждого оператора движения в отдельности, вне связи с предыдущими операторами непрерывного движения. Причем, если перед выполнением оператора движения инструмент уже касался *НП* и *ОП*, это пересечение считается первым.

Далее приводятся примеры программирования непрерывного движения для контуров, изображенных на следующем рисунке:



Вариант а):

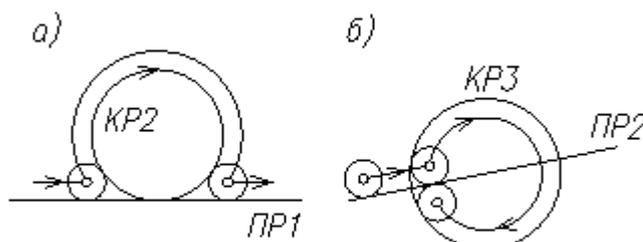
ВПЕРЕД ПР1 ЗА КР1 или **ВПЕРЕД ПР1 ЗА 1 КР1** 1-2
ВПЕРЕД ПР1 ЗА 2 КР1 2-3

Вариант б):

ВПЕРЕД ПР2 ДО 2 КР2 1-2
ВЛЕВО КР2 НА 2 ПР2 2-3
ВПЕРЕД КР2 НА 2 ПР2 3-4

Номер пересечения, указанный в операторе движения, не должен превышать общего числа пересечений данных *НП* и *ОП*, в противном случае ситуация считается ошибочной (например, задание второго пересечения двух прямых). Исключение составляет третье пересечение окружности с прямой, под которым подразумевается полный обход окружности.

Ниже приводятся примеры программирования с "третьим пересечением" для траекторий, изображенных на рисунке.



Вариант а):

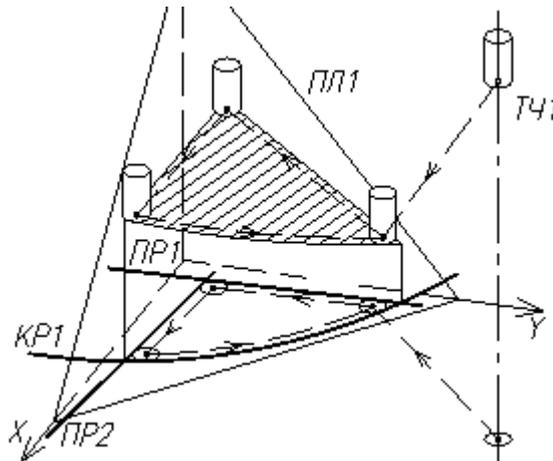
ВПЕРЕД ПР1 ДО КР2
НАЗАД КР2 ДО 3 ПЕРЕСЕЧ ПР1

Вариант б):

**ВПЕРЕД ПР2 ЗА КР3
ВЛЕВО КР3 ДО 3 ПР2**

Если поверхность детали является наклонной плоскостью, то команды движения следует задавать так же, как и в случае расположения обрабатываемого контура в плоскости xOy . Истинное перемещение инструмента представляет собой проекцию на $ПД$ в направлении оси Z эквидистанты к этому контуру (с учетом формы инструмента и его положения относительно $ПД$).

Следующий пример показывает программируемое и истинное перемещения инструмента при заданной поверхности детали:



**МЕТОД ЛИНЕЙН
ИЗ ТЧ1
ИДИ ДО ПР1 ДО ПЛ1 ЗА КР1
ВЛЕВО ПР1 ДО ПР2
ВЛЕВО ПР2 ДО КР1
ВЛЕВО КР1 ДО ПР1
ПОВДЕТ ОТМЕН
ВТОЧКУ ТЧ1**

1.4.1.3.4 Неявное определение ограничивающей поверхности

Неявное определение ограничивающей поверхности имеет место в том случае, когда ограничивающая поверхность и относящийся к ней модификатор в операторе движения опущены. Они выбираются системой автоматически - по направляющей поверхности следующего оператора движения.

При этом необходимо задать положение инструмента относительно контура при помощи одного из операторов:

```

{
ИНСТНА
ИНСТЛВ
ИНСТПР
}
    
```

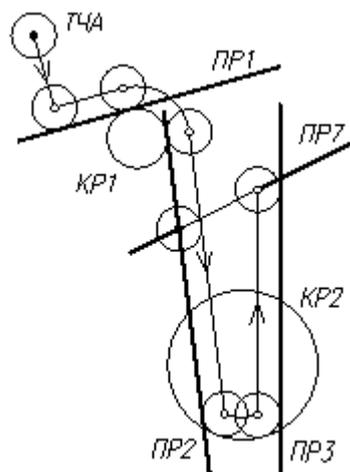
Параметры:

Параметр	Описание
ИНСТНА	Инструмент на контуре детали.
ИНСТЛВ	Инструмент слева от контура (по направлению движения).
ИНСТПР	Инструмент слева от контура (по направлению движения).

Несколько следующих подряд операторов движения одного типа могут заменяться одним, при этом операнды перечисляются через двоеточие. Число пересечений и ограничивающая поверхность с соответствующим модификатором задаются как обычно. При необходимости можно переходить от неявной ограничивающей поверхности к явной и наоборот.

Данный прием программирования позволяет уменьшить объем вводимого текста, однако при этом затрудняется понимание диагностических сообщений, связанных с номером строки.

На следующем рисунке изображен контур при программировании которого целесообразно использовать операторы непрерывного движения с неявной ограничивающей поверхностью:



ИЗ ТЧА
ИДИ ДО ПР1
ИНСТЛВ
ВЛЕВО ПР1
ВПЕРЕД КР1:ПР2 ДО 2 ПЕРЕСЕЧ
ВЛЕВО КР2:ПР3 НА ПР7
ВЛЕВО ПР7 НА ПР2

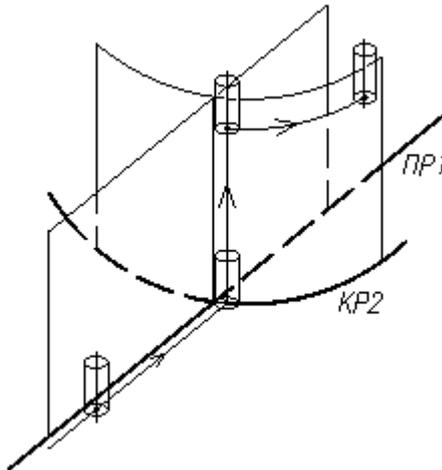
Последний оператор движения всегда должен содержать явно заданную ограничивающую поверхность. После операторов движения, не содержащих ограничивающей поверхности, запрещено использовать операторы постпроцессора и управления интерполяцией.

1.4.1.4 Сочетание поточечного и непрерывного движения

Совместное применение операторов поточечного и непрерывного движений имеет некоторые особенности. Поточечное движение прерывает непрерывное движение, поэтому после операторов **ПРИРАЩ** и **ВТОЧКУ** невозможно продолжить непрерывное движение,

используя операторы **ВПРАВО**, **ВЛЕВО**, **ВПЕРЕД**, **НАЗАД**. Для возобновления непрерывного движения следует задать оператор **ИДИ**.

Исключение составляет перемещение по третьей координате при незаданной поверхности детали.



ВПЕРЕД ПР1 ДО КР2
ПРИРАЩ 50
ВПРАВО КР2

1.4.1.5 Фиктивное движение

Расчет перемещений инструмента без передачи их в управляющую программу можно запрограммировать с помощью операторов фиктивного движения. Этот прием используют для того, чтобы переместить инструмент в положение, которое описать явным образом неудобно или невозможно.

Отключение вывода траектории в управляющую программу осуществляется оператором **НЕОБР**.

Формат:

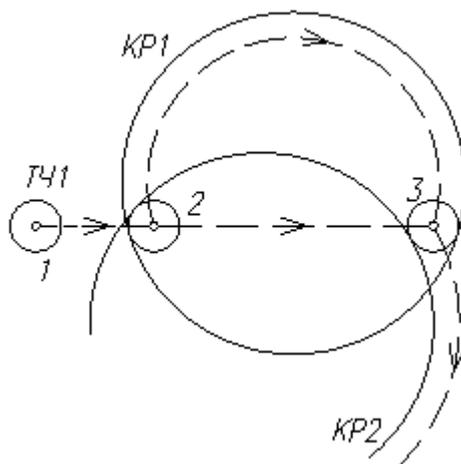
НЕОБР

Возобновление вывода траектории в управляющую программу осуществляется оператором **ОБРАБ**.

Формат:

ОБРАБ

Следующий пример иллюстрирует применение операторов фиктивного движения для непосредственного перехода из положения 1 в положение 3.



ИЗ ТЧ1
НЕОБР
ИДИ ЗА КР2 ЗА КР1
ВЛЕВО КР1 ДО 2 КР2
ОБРАБ
ВПЕРЕД КР2

При этом исключается движение по окружности **КР1** из положения 2 в положение 3.

1.4.1.6 Управление поверхностью детали

Для управления поверхностью детали при поточечном и непрерывном движении предназначен оператор **ПОВДЕТ**.

Формат:

ПОВДЕТ	<i>число</i>
	<i>имя</i>
	ОТМЕН

Параметры:

Параметр	Описание
<i>число</i>	Число или арифметическое выражение, задающее <i>ПД</i> как плоскость, параллельную плоскости хОу и отстоящую от нее на указанном расстоянии.
<i>имя</i>	Символическое имя плоскости, которая будет использоваться как <i>ПД</i> в последующих командах движения.
ОТМЕН	Отмена ранее введенной <i>ПД</i> и возврат к плоскости хОу .

В отличие от оператора **ПЛЗ**, который задает координату **Z** при построении точек,

оператор **ПОВДЕТ** используется с командами движения.

При поточечном движении координата **Z** каждой точки будет вычисляться в соответствии с заданной *ПД*, независимо от величин координат **Z**, заданных в геометрических определениях точек.

При непрерывном движении применение оператора **ПОВДЕТ** равносильно заданию в операторе **ИДИ** поверхности детали с модификатором **НА**. Вводя новые операторы **ПОВДЕТ**, можно переходить от одной поверхности к другой, не прерывая непрерывного движения.

Независимо от того, как в программе определена поверхность детали (в операторе **ПОВДЕТ** или в операторе **ИДИ**), инструмент будет находиться в контакте с ней до тех пор, пока *ПД* не будет отменена командой **ПОВДЕТ ОТМЕН**.

 1. Запрещено применение оператора вида **ПРИРАЩ Z**, приводящего к сходу инструмента с *ПД*.

2. Применять одновременно оператор **ПОВДЕТ** и поверхность детали в операторе **ИДИ** нельзя.

1.4.1.7 Отводы и возвраты инструмента

В зависимости от паспортных данных отвод инструмента в фиксированную позицию или в плоскость безопасности осуществляется либо процессором системы (формируется участок траектории), либо постпроцессором (формируется специальная команда, либо стандартные кадры).

Для программирования этих перемещений используются операторы описания точки смены инструмента и плоскости безопасности (**ТЧКСМЕН** и **БЕЗОПЛ**) и операторы движения **ВОЗВРАТ** и **ОТВОД**.

Для задания точки смены инструмента предназначен оператор **ТЧКСМЕН**.

Формат:

$$\text{ТЧКСМЕН} \left\{ \begin{array}{l} \text{имя} \\ x, y, z \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>имя</i>	Точка смены инструмента.
<i>x, y, z</i>	Координаты точки смены инструмента.

Для задания плоскости безопасности предназначен оператор **БЕЗОПЛ**.

Формат:

$$\text{БЕЗОПЛ} \left\{ \begin{array}{l} \text{ХУПЛ} \\ \text{УЗПЛ} \\ \text{ЗХПЛ} \end{array} \right\} \text{координата}$$

Параметры:

Параметр	Описание
ХУПЛ УЗПЛ ЗХПЛ	Модификаторы выбора координатной плоскости (XY, YZ, ZX соответственно), которой параллельна плоскость безопасности.
<i>координата</i>	Координата безопасного уровня по оси, перпендикулярной указанной координатной плоскости.

Для возврата инструмента в исходное положение или в точку смены инструмента предназначен оператор **ВОЗВРАТ**.

Формат:

ВОЗВРАТ

Для отвода инструмента в плоскость безопасности предназначен оператор **ОТВОД**.

Формат:

ОТВОД

1.4.2 Преобразование траектории движения инструмента

Достаточно часто бывает удобно кодировать элементы контура и траекторию не в базовой системе координат, а в некоторой местной системе. Для приведения местных координат объектов или траектории к базовой системе координат необходимо задать положение этой местной системы в базовой, для чего используются матричные преобразования.

Возможны следующие способы использования матриц:

- в геометрических определениях;
- в командах движения;
- в операторе преобразования участков траектории.

В первом случае новый геометрический объект определяется посредством матричного преобразования, производимого над ранее определенным геометрическим объектом. Во втором случае преобразованию подвергается отмеченный участок траектории, для чего используются операторы **ТРАНС** и **ТРАНСРЕЗ**. И, наконец, третий способ наиболее универсальный. Он позволяет объединять контуры и фрагменты траектории, преобразуя их по заданной матрице.

Темы этого раздела:

- [Преобразование управляющих поверхностей](#)^[179]
- [Преобразование точек траектории](#)^[182]

1.4.2.1 Преобразование управляющих поверхностей

Для преобразования направляющих и ограничивающих поверхностей, относительно которых описывается движение, предназначен оператор **ТРАНС**.

Формат:

$\text{ТРАНС} \left\{ \begin{array}{l} \text{матрица} \\ \text{ОТМЕН} \end{array} \right\}$

Параметры:

Параметр	Описание
<i>матрица</i>	Матрица преобразования.
ОТМЕН	Отмена преобразования.

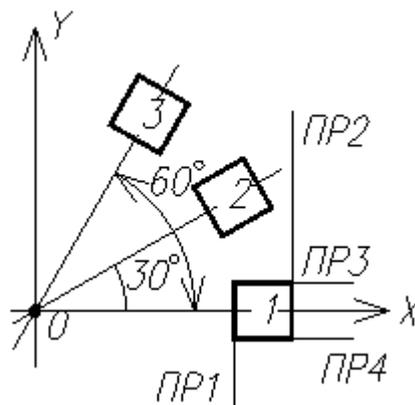
Оператор **ТРАНС** выполняет преобразование *НП* и *ОП* в соответствии с заданной матрицей до применения нового оператора **ТРАНС**. Оператор **ТРАНС ОТМЕН** задает возврат к исходной системе координат.

Если в программе последовательно применяются несколько операторов **ТРАНС**, то следует учесть, что:

- каждый следующий оператор **ТРАНС** определяет преобразование исходной системы координат, а не текущей;
- оператор **ТРАНС ОТМЕН** задает возврат к исходной системе координат, а не к действовавшей ранее.

Матрица, заданная в этом операторе, должна быть плоской, т.е. может быть либо матрицей поворота в плоскости *xOy*, либо матрицей переноса по координатам *X* и *Y*, либо матрицей симметрии относительно прямой, либо произведением матриц такого вида.

Например, необходимо обработать три одинаковых элемента (*I-3*), изображенных на следующем рисунке.



Фрагмент программы имеет вид:

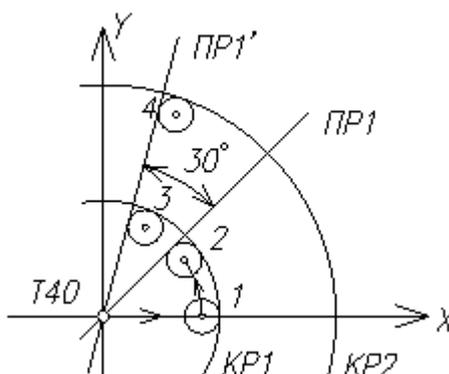
```

.....
команды обработки элемента 1
МАТР М1=ПОВОРОТ,30
ТРАНС М1
команды обработки элемента 1
МАТР М2=ПОВОРОТ,60
ТРАНС М2
команды обработки элемента 1
ТРАНС ОТМЕН
.....

```

Оператор **ТРАНС** действует на поверхности движения (прямые, окружности и др.), а не на получаемые точки траектории. Поэтому следует помнить, что после преобразования поверхностей инструмент может потерять контакт с *НП* или *ОП*, т.к. они занимают новое положение. Во избежание разрыва траектории необходимо продолжать непрерывное движение относительно новых *НП* и *ОП*, или подвести инструмент к ним, используя оператор **ИДИ**.

Например, предполагается вывести инструмент последовательно из **T40** в позиции 1-4.



ИЗ T40

Операторы программы	Участок траектории	
	по программе	фактически
ВЛЕВО ПР3	2-3	2-3
ВПРАВО ПР4 НА ПР1	3-4	3-4
МАТР М1=ПР1		
ТРАНС М1		
ВПЕРЕД ПР4	4-3	4-5
ВЛЕВО ПР3	3-2	5-6
ВПРАВО ПР2	2-1	6-7

См. также:

- ▣ [Преобразование точек траектории](#)^[182]

1.4.2.2 Преобразование точек траектории

Для преобразования точек рассчитанной траектории предназначен оператор **ТРАНСРЕЗ**.

Формат:

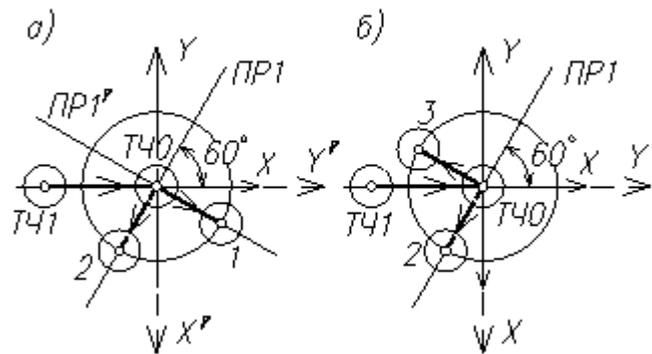
ТРАНСРЕЗ $\left\{ \begin{array}{l} \text{матрица} \\ \text{ОТМЕН} \end{array} \right\}$

Параметры:

Параметр	Описание
<i>матрица</i>	Матрица преобразования.
ОТМЕН	Отмена преобразования.

Оператор **ТРАНСРЕЗ**, в отличие от оператора **ТРАНС**, осуществляет преобразование по заданной матрице рассчитанных точек траектории. Матрица, заданная в этом операторе, может быть любой (в том числе и трехмерной). Действие его распространяется до нового оператора **ТРАНСРЕЗ** или до оператора **ТРАНСРЕЗ ОТМЕН**.

Различие в результатах, полученных при использовании операторов **ТРАНС** и **ТРАНСРЕЗ** иллюстрируется следующим примером.



ПР1=ТЧ0,60
МАТР М1=ПОВХУ,-90
ИЗ ТЧ1
НАПРТ ТЧ0
ИДИ НА ПР1
ТРАНС М1
ВПРАВО ПР1 НА ОКРУЖН (ТЧ0,50)

Пунктиром изображено программируемая траектория движения инструмента, сплошной линией - фактическая траектория.

Без преобразования траектории (при отсутствии оператора **ТРАНС**) инструмент выводится в положение 2.

При наличии оператора **ТРАНС** прямая **ПР1** примет положение **ПР1'**, а оператор **ВПРАВО** вызовет перемещение инструмента из **ТЧ0** в положение 1 (вариант *а*).

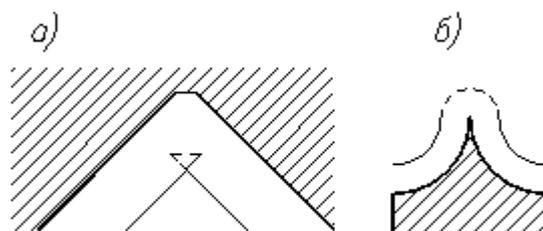
Если использовать оператор **ТРАНСРЕЗ М1** вместо оператора **ТРАНС М1**, то система сначала определит положение инструмента в соответствии с заданными направлением движения и *ОП* (положение 2). Затем эта точка траектории будет преобразована по заданной матрице - т.е. будет повернута по часовой стрелке на 90 град. Она займет положение 3, куда и переместится инструмент из **ТЧ0** (вариант *б*).

См. также:

- [Преобразование управляющих поверхностей](#)^[179]

1.4.3 Управление формированием эквидистанты и траектории

При построении эквидистанты к контуру в ряде случаев оказывается невозможным формирование некоторого ее участка - либо из-за самопересечения эквидистанты (рис. *а*), либо из-за невозможности сопряжения двух ее соседних участков (рис. *б*). Такие участки эквидистанты называются вырожденными и отмечены на рисунке пунктиром.



Специфика алгоритмов работы некоторых устройств ЧПУ требует сглаживания внешних и (или) внутренних углов траектории (как правило, при вводе коррекции).

Для установки режимов формирования эквидистанты и траектории служат операторы **ПЕТЛИ** и **СГЛАЖ**.

При формировании траектории необходимо учитывать наличие круговой интерполяции у системы ЧПУ и параметры круговой интерполяции. Эти данные могут быть заданы с помощью оператора **МЕТОД**. Точность аппроксимации окружностей при линейной интерполяции задается операторами **НАРДОП** и **ВНДОП**.

Темы этого раздела:

- [Анализ наличия петель](#)^[185]
- [Сглаживание эквидистанты](#)^[186]
- [Сглаживание траектории](#)^[187]
- [Задание метода интерполяции и режима аппроксимации](#)^[187]
- [Точность аппроксимации](#)^[191]

1.4.3.1 Сопряжение соседних сегментов при построении эквидистанты

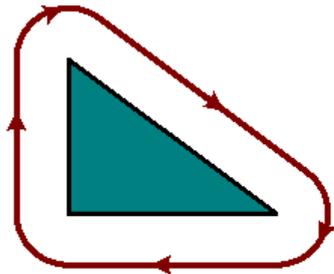
Для управления способом сопряжением соседних сегментов при построении эквидистанты предназначен оператор **СТЫК** следующего формата:

СТЫК	$\left. \begin{array}{l} \text{ПРОДОЛЖ} \\ \text{СГЛАЖ} \\ \text{СРЕЗ} \\ \text{ПРЯМ} \end{array} \right\}$
-------------	-------------------------------------------------------------------------------------------------------------

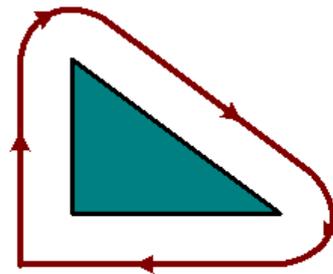
Параметры:

Параметр	Описание
ПРОДОЛЖ	Сопряжение продолжением сегментов до их пересечения

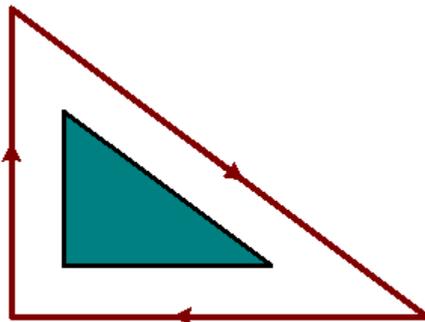
Параметр	Описание
СГЛАЖ	Встраивание между сегментами сопрягающего скругления
ПРЯМ	Сопряжение продолжением касательных к сегментам до их пересечения
СРЕЗ	Отличается от ПРЯМ тем, что острые углы усекаются дополнительным отрезком.



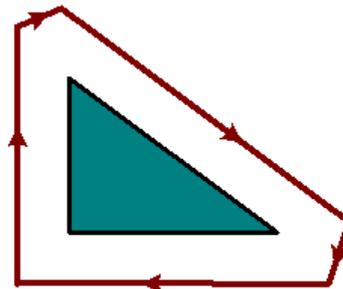
Скругление



Сглаженный угол



Угол



Скошенный угол

1.4.3.2 Анализ наличия петель

Для включения анализа наличия петель и их отсечения предназначен оператор **ПЕТЛИ**.

Формат:

ПЕТЛИ	{	ВЫКЛ	}
		ВКЛ	
		ВСЕ	

Параметры:

Параметр	Описание
ВКЛ	Включение режима анализа наличия петель. При наличии вырождений петли отсекаются.
ВЫКЛ	Выключение режима анализа наличия петель. Наличие вырождений в этом режиме приводит к ошибке при построении эквидистанты.
ВСЕ	Сохранение петель при построении траектории.

 Оператор **ПЕТЛИ** действует только при построении эквидистанты к контуру - как в геометрических определениях, так и при использовании контура в качестве *НП* непрерывного движения.

1.4.3.3 Сглаживание эквидистанты

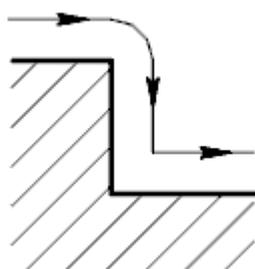
Для сглаживания эквидистанты радиусом инструмента предназначен оператор **СГЛАЖ** следующего формата:

$\text{СГЛАЖ} \left\{ \begin{array}{l} \text{ВКЛ} \\ \text{ВЫКЛ} \end{array} \right\}$

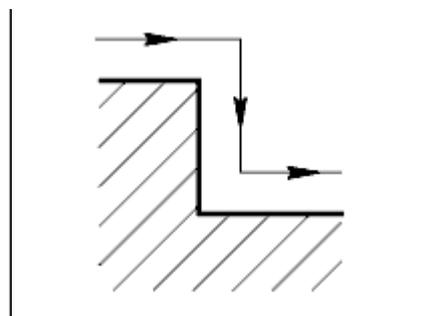
Параметры:

Параметр	Описание
ВКЛ	Включение режима сглаживания. Вызывает скругление эквидистанты радиусом инструмента при внешнем обходе углов контура.
ВЫКЛ	Отмена режима сглаживания.

Оператор действует только при построении эквидистанты к контуру - как в геометрических определениях, так и при использовании контура в качестве *НП* непрерывного движения. Это показано на следующих рисунках.



СГЛАЖ ВКЛ



СГЛАЖ ВЫКЛ

1.4.3.4 Сглаживание траектории

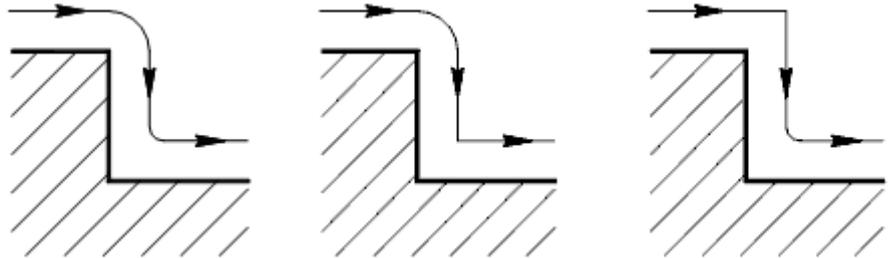
Для скругления внешних и внутренних углов траектории предназначен оператор СГЛАЖ следующего формата:

```
СГЛАЖ { ВНЕ [РАДИУС], число }
        { ВНУТРИ [ОТМЕН] }
```

Параметры:

Параметр	Описание
ВНЕ	Скругление внешних углов траектории.
ВНУТРИ	Скругление внутренних углов траектории.
<i>число</i>	Величина радиуса скругления в мм.
ОТМЕН	Отмена скругления траектории на внешних или внутренних углах в зависимости от модификаторов ВНЕ / ВНУТРИ .

Оператор действует на траекторию любого движения - как непрерывного, так и поточечного, и только при включенной коррекции на радиус инструмента. Это показано на следующих рисунках.



КОРРЕК СЛЕВА
СГЛАЖ ВНЕ,РАДИУС,10
ОТМЕН
СГЛАЖ ВНУТРИ,РАДИУС,2
РАДИУС,2

КОРРЕК СЛЕВА **КОРРЕК СЛЕВА**
СГЛАЖ ВНЕ,РАДИУС,10 **СГЛАЖ ВНЕ,**
СГЛАЖ ВНУТРИ,ОТМЕН **СГЛАЖ ВНУТРИ,**

1.4.3.5 Задание метода интерполяции и режима аппроксимации

Для задания метода интерполяции и режима аппроксимации окружности отрезками прямой используется оператор МЕТОД.

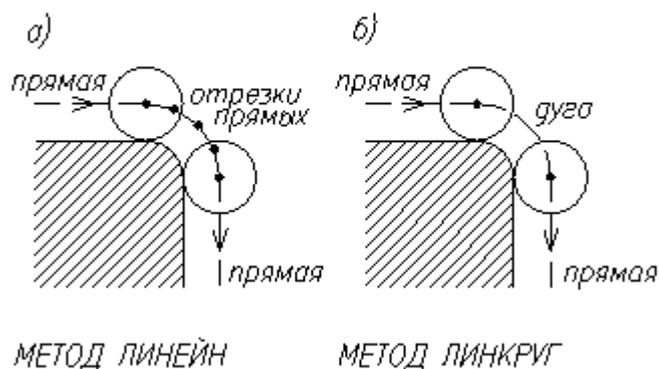
Формат:

$$\text{МЕТОД} \left\{ \begin{array}{l} \text{ЛИНЕЙН} \left[\text{, МАКСЛИН, } \textit{длина} \right] \\ \text{ЛИНКРУГ} \left[\begin{array}{l} \text{НЕКВАДР} \\ \text{, КВАДР} \end{array} \right] \left[\text{, МАКСРАД, } \textit{радиус} \right] \left[\text{, МИНДЛИН, } \textit{дуга} \right] \end{array} \right\}$$

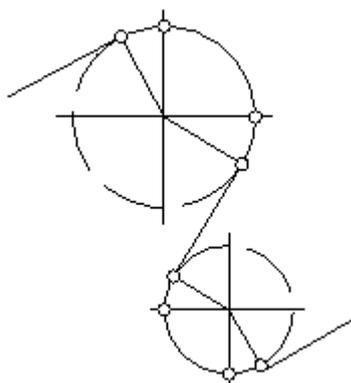
Параметры:

Параметр	Описание
ЛИНЕЙН	Линейная интерполяция (аппроксимация окружностей отрезками прямых).
ЛИНКРУГ	Линейно-круговая интерполяция (аппроксимация кривых дугами окружностей).
МАКСЛИН	Признак ограничения длины аппроксимирующих отрезков.
НЕКВАДР	Линейно-круговая интерполяция без деления окружностей на квадранты.
КВАДР	Линейно-круговая интерполяция с делением окружностей на квадранты.
МАКСРАД	Признак замены круговой интерполяции линейной в случае превышения заданного значения радиуса дуги.
МИНДЛИН	Признак ограничения длины дуги.
<i>длина</i>	Максимальная длина отрезков, аппроксимирующих дугу окружности.
<i>радиус</i>	Максимальный радиус дуги, при котором автоматически происходит переход к линейной интерполяции.
<i>дуга</i>	Минимальная длина дуги, при которой производится автоматическая замена ее отрезком прямой.

На рисунке представлены: линейная (*a*) и линейно-круговая (*б*) интерполяции.

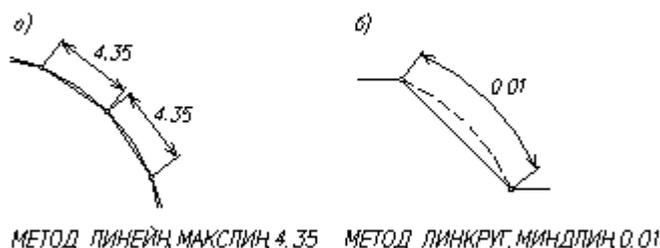


При использовании модификатора **КВАДР** система разбивает дуги окружностей, составляющих траекторию, на участки, лежащие в отдельных квадрантах :



Отсутствие в программе оператора **МЕТОД** эквивалентно применению оператора **МЕТОД ЛИНКРУГ, НЕКВАДР**.

Примеры использования параметров **МАКСЛИН** и **МИНДЛИН** приведены на следующем рисунке.



Применение параметра **МАКСРАД** в операторе **МЕТОД** приведен на следующем рисунке.



Во всех приводимых примерах точками ограничены участки траектории, заданные в УП отдельными кадрами.

Круговая интерполяция разрешена только в одной из координатных плоскостей или в плоскостях, параллельных им. Круговая интерполяция с делением на квадранты возможна только в плоскостях, параллельных плоскости xOy .

Если поверхность детали не параллельна какой-либо координатной плоскости, следует задавать оператор **МЕТОД ЛИНЕЙН**, даже если обрабатываемый контур не содержит дуг окружностей, т.к. система в ряде случаев сглаживает траекторию сопрягающими дугами.

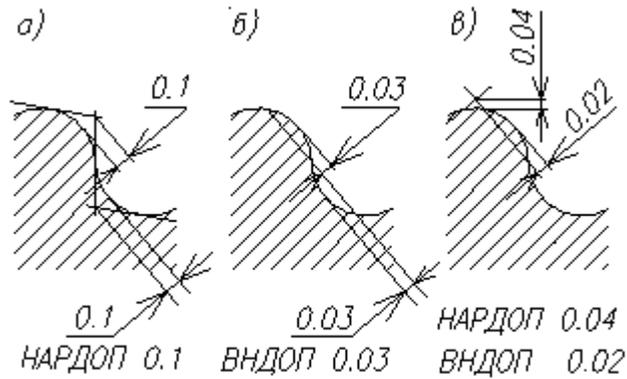
 Применение оператора **МЕТОД** при программировании поточечного движения не имеет смысла, т.к. этим способом задаются только прямолинейные перемещения центра инструмента.

См. также:

- [Точность аппроксимации](#)^[191]

1.4.3.6 Точность аппроксимации

Для задания точности линейной аппроксимации дуг окружностей, которые заменяются отрезками прямых, используются операторы **НАРДОП** и **ВНДОП** (см. рисунок):



Точность аппроксимации касательными (наружный допуск) задается оператором **НАРДОП** (рис. а).

Формат:

```
НАРДОП число
```

Точность аппроксимации хордами (внутренний допуск) задается оператором **ВНДОП** (рис. б).

Формат:

```
ВНДОП число
```

Параметры:

Параметр	Описание
число	Точность аппроксимации.

При одновременном использовании этих операторов производится аппроксимация секущими (рис в).

☛ Если не заданы ни **НАРДОП**, ни **ВНДОП**, то система устанавливает следующие их значения:

НАРДОП 0.05
ВНДОП 0

1.4.4 Траектория, полученная объединением участков траектории

Участок траектории может быть получен объединением участков траектории и контуров.

Формат:

$$\text{ХОД}(\text{индекс}_1) = \left\{ \begin{array}{l} \text{ПО} \\ \text{ОБРАТН} \end{array} \right\} [\text{матрица},] \left[\text{ОТДО} \left\{ \begin{array}{l} \text{номер}_1 \\ \text{точка}_1 \end{array} \right\} \left\{ \begin{array}{l} \text{номер}_2 \\ \text{точка}_2 \end{array} \right\} \right] \left\{ \begin{array}{l} \text{ХОД}(\text{индекс}_2) \\ \text{контур} \end{array} \right\} [, \dots]$$

Параметры:

Параметр	Описание
<i>индекс₁</i>	Номер нового участка траектории.
ПО	Присоединяемый контур или его часть не изменяет направления и присоединяется к последнему сегменту составного контура своим началом.
ОБРАТН	Присоединяемый контур или его часть изменяет направление на противоположное и присоединяется к последнему сегменту составного контура своим концом.
<i>матрица</i>	Матрица, задающая преобразование присоединяемого контура или фрагмента.
ОТДО	Присоединение фрагмента контура, ограниченного двумя точками.
<i>номер₁</i>	Номер начальной опорной точки присоединяемого фрагмента контура.
<i>номер₂</i>	Номер конечной опорной точки присоединяемого фрагмента контура.
<i>точка₁</i>	Имя начальной точки присоединяемого фрагмента контура.
<i>точка₂</i>	Имя конечной точки присоединяемого фрагмента контура.
<i>индекс₂</i>	Номер исходного участка траектории.
<i>контур</i>	Имя присоединяемого контура.

Несколько фрагментов могут быть объединены в одной строке через запятую.

Участок траектории с заданным индексом строится в результате комбинации контуров и участков траектории с использованием матриц и модификаторов, задающих направление и границы встраиваемого участка.

При использовании оператора данного формата возможны ошибки движения, связанные с отсутствием оператора **ИЗ**, например:

```

ДЕТАЛЬ
.
.
ХОД(3)=ИЗ 0,0,0
.
.
КОНТУР К1=ИЗ ТЧ1,ХБ
.
.
КОНКОН
ХОД(1)=-1,К1, ХОД(3)
КОНЕЦ

```

В данном примере объект **ХОД(1)**, определен как составной и, в соответствии с номером своего индекса, будет обрабатываться первым. Поскольку первым элементом этого составного объекта является контур, а не участок траектории, то в объекте **ХОД(1)** будет отсутствовать начало непрерывного движения - оператор **ИЗ**, что приведет к ошибке.

См. также:

- [Оператор включения контура](#)^[212]

1.4.5 Копирование участка траектории с преобразованием координат

Для включения в траекторию участка, полученного матричным преобразованием заданного участка, предназначен оператор **+=** следующего формата.

```
ХОД += ХОД(индекс), матрица, число
```

Параметры:

Параметр	Описание
<i>индекс</i>	Номер участка траектории.
<i>матрица</i>	Матрица, задающая преобразование участка траектории.
<i>число</i>	Число повторений операции преобразования траектории.

1.4.6 Перенос участка траектории заданное число раз

Для включения в траекторию участка, полученного переносом заданного участка, предназначен оператор **+=** следующего формата.

```
ХОД += ХОД(индекс), ПЕРЕНОС, x, y, число
```

Параметры:

Параметр	Описание
<i>индекс</i>	Номер участка траектории.
<i>x, y</i>	Смещения по осям X и Y.
<i>число</i>	Число повторений операции преобразования траектории.

1.4.7 Перенос участка траектории во все точки массива точек

Для включения в траекторию участка, полученного переносом заданного участка во все точки массива точек, предназначен оператор += следующего формата.

ХОД += ХОД(*индекс*), ПЕРЕНОС, *массив точек*, ЦЕНТР, *точка*, ПОВОРОТ, *угол*

Параметры:

Параметр	Описание
<i>индекс</i>	Номер участка траектории.
<i>массив точек</i>	Массив точек.
<i>точка</i>	Точка привязки участка траектории. Участок траектории помещается в новую позицию так, что точка привязки совмещается с точкой массива.
<i>угол</i>	Угол поворота.

1.5 Управляющие операторы

Управляющие операторы позволяют изменять последовательность выполнения других операторов программы, в т.ч. при определенных условиях пропускать или многократно выполнять некоторую последовательность операторов. Такие возможности позволяют составлять на Техтроне программы, имеющие сложную логическую структуру, но значительно более компактные, чем программы с линейной структурой.

К управляющим операторам относятся операторы **НАМЕТКУ**, **ЕСЛИ** и **ПОВТОР**.

Темы этого раздела:

- [Оператор безусловного перехода](#)^[195]
- [Условный оператор](#)^[195]
- [Условный оператор с альтернативой](#)^[196]
- [Условный оператор с несколькими альтернативами](#)^[198]
- [Оператор цикла со счетчиком](#)^[199]
- [Оператор цикла с условием](#)^[200]
- [Вложенные циклы](#)^[202]

- [Операторы прерывания выполнения](#)^[203]

1.5.1 Оператор безусловного перехода

Оператор **НАМЕТКУ** вызывает безусловную передачу управления на помеченный оператор.

Формат:

```
НАМЕТКУ имя
```

Параметры:

Параметр	Описание
<i>имя</i>	Имя использованной в программе метки.

После этого оператора выполняется не следующий по порядку оператор, а тот, который помечен указанной меткой.

Пример:

```
ИЗ ТЧ8
НАМЕТКУ К2
ИДИ НА ПРХ
К2: ВТОЧКУ ТЧА
ПРИРАЩ 20
```

В приведенном фрагменте программы после оператора **ИЗ** будет выполнен не оператор **ИДИ**, а помеченный оператор **ВТОЧКУ**.

1.5.2 Условный оператор

Условный оператор предназначен для выполнения одного или нескольких операторов при определенном условии, заданном логическим выражением.

Формат:

```
ЕСЛИ (логическое выражение) ТО
последовательность операторов
КОНЕСЛИ
```

Параметры:

Параметр	Описание
<i>логическое выражение</i>	Условие выполнения оператора.
<i>последовательность операторов</i>	Операторы Техтрана.

Операторы из *последовательности операторов* выполняются в том случае, если логическое выражение имеет значение **ИСТИНА**. В противном случае никаких действий не производится.

Пример:

В следующем примере, если значение переменной **К** меньше **10**, то будут выполнены операторы присваивания для **ТЧ1** и **ТЧ2**:

```
ЕСЛИ (К < 10) ТО  
  ТЧ1 = 0, 0, 5  
  ТЧ2 = ПРЗ, ПР4  
КОНЕСЛИ  
  ТЧ6 = ЦЕНТР, КР2
```

При других значениях переменной **К** логическое выражение принимает значение **ЛОЖЬ**, операторы между **ЕСЛИ** и **КОНЕСЛИ** опускаются и выполняется оператор **ТЧ6 =**

Сокращенная запись условного оператора имеет вид:

```
ЕСЛИ (логическое выражение) оператор
```

В сокращенной записи условного оператора *оператор* может быть только простым оператором. Применение составных операторов типа **ЕСЛИ**, **ПОВТОР**, **КОНТУР**, **МАКРО** -недопустимо.

Пример:

```
ЕСЛИ (А > 0.5 ИЛИ А < 0) НАМЕТКУ М24
```

Переход на метку **М24** выполняется, если значение переменной **А** больше **0.5** или меньше **0**.

1.5.3 Условный оператор с альтернативой

Условный оператор с альтернативой позволяет выполнить ту или другую последовательность операторов на основании истинности условия, заданного логическим выражением.

Формат:

```
ЕСЛИ (логическое выражение) ТО  
  [последовательность операторов1]  
  [ИНАЧЕ  
  [последовательность операторов2]]  
КОНЕСЛИ
```

Параметры:

Параметр	Описание
<i>логическое выражение</i>	Условие выполнения оператора.
<i>последовательность операторов₁</i> <i>последовательность операторов₂</i>	Операторы Техтрана.

Если логическое выражение имеет значение **ИСТИНА**, то выполняются операторы *последовательности₁*, расположенные после служебного слова **ТО**. Если же значение логического выражения - **ЛОЖЬ**, то выполняются операторы *последовательности₂*, стоящие за словом **ИНАЧЕ**. Любая из этих последовательностей может быть пустой - в этом случае в зависимости от значения логического выражения будет выполняться либо единственная присутствующая последовательность, либо операторы, стоящие за ограничителем **КОНЕСЛИ**. При отсутствии слова **КОНЕСЛИ** выдается сообщение об ошибке и требуется корректировка текста программы.

Примеры:

```
ЕСЛИ (A >= 0) ТО
  A=A
ИНАЧЕ
  A=-A
КОНЕСЛИ
```

В результате будет вычислено абсолютное значение А.

В следующем примере отсутствует вторая последовательность операторов:

```
ЕСЛИ (X >= 0) ТО
  Y=КВКОР(X)
ИНАЧЕ
КОНЕСЛИ
```

Т.к. извлечение квадратного корня из отрицательного числа запрещено, вариант при $X < 0$ не должен рассматриваться. Оператор **ЕСЛИ-ТО-ИНАЧЕ-КОНЕСЛИ** в данном случае аналогичен оператору: **ЕСЛИ(X БРВ 0)Y=КВКОР (X)**.

В следующем примере показана возможность модификации траектории в зависимости от положения ограничивающей поверхности:

```
ВПЕРЕД ПРХ НА ПРУ
ПР1 = ПАРЛЕЛ, ПРХ, А
ЕСЛИ (A > 0) ТО
  ВЛЕВО ПРУ НА ПР1
ИНАЧЕ
  ВПРАВО ПРУ НА ПР1
КОНЕСЛИ
```

Данный прием позволяет избежать ошибок движения, когда заранее известны варианты положения прямой **ПР1** относительно оси **X**.

1.5.4 Условный оператор с несколькими альтернативами

Условный оператор с несколькими альтернативами проверяет истинность одного из условий, заданных логическими выражениями, и позволяет выполнить соответствующую этому условию последовательность операторов.

Формат:

```

ЕСЛИ (логическое выражение1) ТО
  [последовательность операторов1]
ИНАЧЕ ЕСЛИ (логическое выражение2) ТО
  [последовательность операторов2]
  ...
  [ИНАЧЕ
  [последовательность операторов по умолчанию]]
КОНЕСЛИ
  
```

Параметры:

Параметр	Описание
<i>логическое выражение₁</i> , <i>логическое выражение₂</i>	Условия выполнения операторов.
<i>последовательность операторов₁</i> , <i>последовательность операторов₂</i> , <i>последовательность операторов по умолчанию</i>	Операторы Техтрана.

При выполнении этого оператора последовательно проверяется каждое из условий и, если условие имеет значение **ИСТИНА**, то выполняется соответствующая ему последовательность операторов. Если все условия имеют значение **ЛОЖЬ**, то выполняется последовательность операторов по умолчанию (если она задана).

Пример:

```

ВПЕРЕД ПРХ НА ПРУ
ПР1 = ПАРЛЕЛ, ПРХ, А
ЕСЛИ (А<0) ТО
  ВЛЕВО ПРУ НА ПР1
ИНАЧЕ ЕСЛИ (А>0) ТО
  ВПРАВО ПРУ НА ПР1
ИНАЧЕ
  ВПЕРЕД ПРХ НА ПР2
  
```

КОНЕСЛИ

1.5.5 Оператор цикла со счетчиком

Оператор **ПОВТОР** предназначен для многократного повторения участка программы, называемого телом цикла. Оператор **ПОВТОР** со счетчиком применяется в тех случаях, когда известно количество повторений цикла.

Формат:

```
ПОВТОР переменная = нач, кон [, шаг ]
```

```
последовательность операторов
```

```
КОНЦИКЛ [переменная ]
```

Параметры:

Параметр	Описание
<i>переменная</i>	Идентификатор вещественной переменной, называемой управляющей переменной цикла .
<i>нач</i>	Число, переменная или арифметическое выражение, задающее начальное значение управляющей переменной цикла.
<i>кон</i>	Число, переменная или арифметическое выражение, задающее конечное значение управляющей переменной цикла.
<i>шаг</i>	Число, переменная или арифметическое выражение, называемое шагом цикла и задающее приращение управляющей переменной цикла (по умолчанию шаг=1).
<i>последовательность операторов</i>	Операторы Техтрона (тело цикла).

При выполнении цикла значение *переменной* изменяется от *нач* до *кон* с заданным шагом. Шаг может быть положительным, отрицательным, но не может быть равным нулю.

Цикл выполняется так:

1. Управляющей *переменной* цикла присваивается начальное значение *нач*.
2. Выполняется проверка завершения цикла. Цикл завершается при выполнении одного из следующих условий:
 - при положительном значении шага цикла текущее значение переменной цикла превышает конечное значение ($шаг > 0$, $переменная > кон$);
 - при отрицательном значении шага цикла текущее значение переменной цикла меньше конечного значения ($шаг < 0$, $переменная < кон$).

При завершении цикла управление передается на оператор, следующий за оператором **КОНЦИКЛ**.

3. Выполняются операторы, составляющие тело цикла. Среди операторов могут быть такие, которые изменяют текущее значение переменной цикла, и такие, которые производят досрочный выход из цикла.

4. Переменной цикла присваивается значение $\text{переменная} = \text{переменная} + \text{шаг}$ и производится переход на пункт 2.

При нормальном завершении цикла управляющая переменная принимает первое значение не удовлетворяющее условию пункта 2, при досрочном выходе из цикла – сохраняет свое текущее значение. Значение управляющей переменной может быть использовано при вычислениях так же, как значение любой другой вещественной переменной - как внутри тела цикла, так и вне его.

Пример:

```
ВЕЩ А(10)
ПОВТОР И=1,10
  А(И)=И*2
КОНЦИКЛ И
```

В данном примере первым десяти элементам массива **А** присваиваются значения четных чисел в пределах от **2** до **20**.

Следует помнить, что наращивание значения счетчика цикла на величину шага происходит только при выполнении оператора **КОНЦИКЛ**, но не всегда после последнего оператора тела цикла, т.к. среди них может находиться оператор перехода на метку.

Например:

```
Т=6.5
ПОВТОР И=1,7
  Т=Т-И # 5.5; 5.5; 4.5; 2.5; -0.5
  ЕСЛИ (Т МНШ 0) НАМЕТКУ М1
  Т=Т+2 # 7.5; 7.5; 6.5; 4.5
КОНЦИКЛ И
М1:Т=125
```

В процессе выполнения цикла переменная **Т**, первоначально равная **6.5**, последовательно принимает следующие значения: **5.5; 7.5; 5.5; 7.5; 4.5; 6.5; 2.5; 4.5; -0.5**. Затем оператор **ЕСЛИ** передает управление на оператор с меткой **М1**, при этом: **Т=125**, а управляющая переменная сохраняет свое последнее значение **И=5**. После знаков комментария приведены значения переменной **Т**, приобретаемые ею на каждом шаге.

1.5.6 Оператор цикла с условием

Оператор **ПОВТОР** предназначен для многократного повторения участка программы, называемого телом цикла. Оператор **ПОВТОР** с условием применяется в тех случаях, когда легко формулируется условие выполнения цикла.

Формат:

ПОВТОР ПОКА (*логическое выражение*)
последовательность операторов
КОНЦИКЛ

Параметры:

Параметр	Описание
<i>логическое выражение</i>	Условие выполнения тела цикла.
<i>последовательность операторов</i>	Операторы Техтрана (тело цикла).

Такая синтаксическая конструкция позволяет повторять выполнение операторов, составляющих тело цикла, до тех пор пока логическое выражение, находящееся в заголовке цикла, сохраняет значение **ИСТИНА**.

Следует обратить внимание, что до оператора **ПОВТОР ПОКА** должны быть заданы значения переменных, использованных в логическом выражении - для того, чтобы мог быть вычислен его результат в самом начале выполнения оператора. Для нормального завершения такого цикла необходимо, чтобы в теле цикла находились операторы присваивания, изменяющие значения переменных, которые входят в логическое выражение. Если перед очередным повторением цикла обнаруживается, что логическое выражение имеет значение **ЛОЖЬ**, то управление передается оператору, следующему за оператором **КОНЦИКЛ**. Если значение выражения **ЛОЖЬ** с самого начала, операторы, составляющие цикл, не выполняются.

В качестве примера приводится программа вычисления кубического корня с использованием метода Ньютона. Он заключается в следующем:

если **A** - число, из которого извлекается кубический корень, то первое приближенное значение результата вычисляется по формуле, при $x_0=1$ (см. формулу):

$$x_1 = \frac{2x_0 + \frac{A}{(x_0)^2}}{3}$$

Далее полагаем $x_0=x_1$ и вычисляем новое, уточненное значение кубического корня (x_2).

Процесс итерации продолжается до тех пор, пока разница между двумя последовательными значениями x не окажется в пределах допустимой ошибки. Например, процесс вычисления корня кубического из числа **4** можно представить следующим образом:

$$x_0=1$$

$$x_1=(2+4/1)/3=2$$

$$x_2=(4+4/4)/3=1.667$$

$$x_3=(3.333+4/2.778)/3= 1.591$$

Если продолжать процесс вычислений, то значения x будут приближаться к **1.5874** -корню кубическому из **4**.

Программа на Техтране будет иметь вид:

```

ХСТАР=0
ХНОВ=1
ПОВТОР ПОКА (АБС(ХНОВ-ХСТАР) > 0.0001)
  ХСТАР=ХНОВ
  ХНОВ = (2*ХСТАР+4/ХСТАР**2)/3
КОНЦИКЛ

```

1.5.7 Вложенные циклы

Разрешается использовать вложенные циклы. В этом случае тело внутреннего цикла должно полностью находиться в теле внешнего цикла, т. е. перекрытие циклов не допускается. Глубина вложений циклов может быть не более 10.

В качестве примера приводится программа возведения в степень 1, 2, 3 и так до 5-ой первых 10-ти чисел натурального ряда:

```

ВЕЩ А(10,5)
ПОВТОР СТЕП=1,5
  ПОВТОР ЧИСЛО=1,10
    А (ЧИСЛО,СТЕП)=ЧИСЛО**СТЕП
  КОНЦИКЛ ЧИСЛО
КОНЦИКЛ СТЕП

```

При входе в цикл по переменной **СТЕП**, ее первоначальное значение приравнивается единице. Затем происходит вход в цикл с управляющей переменной **ЧИСЛО**. При изменении ее значений от **1** до **10** вычисляется 1-ая степень 10-ти чисел. Результаты последовательно заносятся в элементы массива: **А(1,1)**, **А(2,1) ... А(10,1)**. Затем последовательно происходит:

- выход из вложенного цикла;
- наращивание переменной **СТЕП** на 1;
- передача управления счетчику цикла и проверка условия выполнения цикла по переменной **СТЕП**;
- вход во вложенный цикл;
- вычисление 2-ой степени 10-ти чисел (**СТЕП=2**).

При втором выполнении вложенного цикла показатель степени равен 2-м, переменная **ЧИСЛО** снова изменяется от 1 до 10, а результаты возведения в степень последовательно заносятся в ячейки массива: **А(1,2)**, **А(2,2) ... А(10,2)**.

Всего вложенный цикл выполняется 5 раз -при изменении переменной **СТЕП** (показателя степени) от **1** до **5**; единственный оператор, составляющий тело вложенного цикла выполняется 10 раз при каждом значении переменной **СТЕП**, в результате чего заполняются 50 ячеек массива **А**.

Не допускается применять операторы, находящиеся вне тела цикла и передающие управление внутрь цикла. Использование циклов должно быть согласовано с использованием макросов. Это значит, что цикл, начинающийся (кончающийся) внутри макроса, должен там же и кончатся (начинаться).

Варианты допустимых передач управления показаны на следующем рисунке:



1.5.8 Операторы прерывания выполнения

Оператор **ПРЕРВАТЬ** прерывает выполнение тела цикла или секции.

В теле цикла оператор **ПРЕРВАТЬ** прерывает выполнение ближайшего содержащего его оператора **ПОВТОР** и передает управление на оператор, следующий за оператором **КОНЦИКЛ**.

В секции оператор **ПРЕРВАТЬ** прерывает выполнение операторов секции.

Формат:

ПРЕРВАТЬ

Выполнение оператора **АВОСТ** приводит к аварийному завершению программы с выдачей сообщения.

Формат:

АВОСТ <строка сообщения>

1.6 Специальные операторы

Термин **специальные операторы** объединяет средства языка, которые предназначены для:

- изменения стандартного набора служебных слов;
- вставки текста из внешнего файла.
- управления версиями

Темы этого раздела:

- [Синонимы](#)^[204]
- [Вставка текста](#)^[204]

- [Управление версиями](#)^[205]

1.6.1 Синонимы

Оператор **СИНОНИМ** служит для замены стандартных служебных слов синонимами и имеет следующий формат:

$\text{СИНОНИМ}(\text{служебное слово}_1, \text{синоним}_1)$ $\left[\begin{array}{l} , (\text{служебное слово}_2, \text{синоним}_2), \\ \dots\dots\dots \\ (\text{служебное слово}_n, \text{синоним}_n) \end{array} \right]$

В качестве синонимов могут употребляться только имена. Оператор должен быть применен до первого использования синонима.

Пример:

Задав оператор:

СИНОНИМ (ВПЕРЕД, В),(ПЕРЕСЕЧ, П)

можно вместо оператора

ВПЕРЕД ПР1 ДО 2 ПЕРЕСЕЧ КР1

употребить в программе оператор

В ПР1 ДО 2 П КР1

1.6.2 Вставка текста

Оператор вставки текста (*) позволяет вносить в программу текст, хранящийся во внешнем файле и имеет два формата.

Формат 1:

<i>*имя</i>

Формат 2:

<i>*литерал</i>

Параметры:

Параметр	Описание
<i>имя</i>	Имя файла без расширения, содержащего вставляемый текст.
<i>литерал</i>	Спецификация файла.

Пример:

***PROG1**
***'D:\TEXT\PROG2.TXT'**

В первом случае вместо оператора ***PROG1** при обработке программы вставляется и обрабатывается текст, находящийся в файле **PROG1** текущей директории. Во втором случае вставляется содержимое файла **PROG2.TXT** из директории **D:\TEXT**.

1.6.3 Управление версиями

Для обеспечения совместимости с программами, разработанными в предыдущих версиях, используется оператор **ВЕРСИЯ**.

Формат:

ВЕРСИЯ <i>версия₁</i> , <i>версия₂</i> , <i>версия₃</i>

Параметры:

Параметр	Описание
<i>версия_i</i>	номер версии

Данный оператор записывается первой строкой перед оператором **ДЕТАЛЬ**.

Например:

ВЕРСИЯ 4, 5
ДЕТАЛЬ 'ШТУЦЕР'

Отсутствие оператора версия в программе означает то, что действуют соглашения предшествующие версии 4.5.

При создании новой программы при помощи команды *Создать* текущая версия (оператор **ВЕРСИЯ**) формируется автоматически.

1.7 Контур

Наряду с элементарными типами геометрических объектов в Техтроне имеется комбинированный тип – **КОНТУР**. Геометрические объекты этого типа представляют собой совокупности отрезков прямых и дуг окружностей. Они могут быть скопированы, преобразованы по матрице, использованы в командах непрерывного движения аналогично элементарным геометрическим объектам, т.е. инструмент может быть выведен на такой объект оператором **ИДИ**, а полный или частичный его обход осуществляется с помощью операторов непрерывного движения. Контур определяет геометрию области обработки при выполнении различных технологических переходов.

Элементы контура, используемые в различных командах и построениях, не имеют собственных символических имен и характеризуются своим порядковым номером или координатами.

Сегмент контура – отрезок прямой или дуга окружности, из которых состоит контур. Определяется именем контура и порядковым номером сегмента:

контур, номер сегмента.

Может использоваться, например, для указания участка контура, с которым пересекается прямая или окружность.

Опорная точка – точка сопряжения соседних сегментов. Определяется именем контура и своим порядковым номером

контур, номер точки.

Начальная точка контура имеет номер 0, конечная точка первого сегмента – номер 1, конечная точка второго сегмента – номер 2 и т. д.

Может использоваться, например, для задания технологической команды.

Точка контура – точка, лежащая на контуре (опорная точка или точка, принадлежащая сегменту контура). Определяется именем контура и порядковым номером или точкой.

контур, {номер точки/точка}.

Стык сегментов – часть контура, включающая сопряжение соседних сегментов. Определяется именем контура и сегментом, имеющим наименьший порядковый номер.

контур, номер сегмента.

Замкнутый контур – контур, начальная и конечная точки которого совпадают.

Темы этого раздела:

- [Определение контура](#)^[206]
- [Движение по контуру](#)^[228]
- [Встраивание фасок и скруглений в контур](#)^[236]
- [Траектория как контур](#)^[238]

1.7.1 Определение контура

Формат:

[**КОНТУР**] *имя = определение контура*

Служебное слово **КОНТУР** должно присутствовать в том случае, если определение типа не было сделано ранее.

Темы этого раздела:

- [Контур, построенный из элементов](#)^[207]
- [Контур, являющийся сплайном](#)^[216]
- [Контур, эквидистантный заданному](#)^[219]
- [Контур, имеющий форму прямоугольника](#)^[220]
- [Контур, имеющий форму окружности](#)^[221]
- [Контур, имеющий форму эллипса](#)^[222]
- [Контур, имеющий форму прямого паза](#)^[223]
- [Контур, имеющий форму радиусного паза](#)^[224]
- [Контур, имеющий форму сектора](#)^[225]
- [Контур, имеющий форму правильного многоугольника](#)^[226]
- [Габаритный контур](#)^[227]

■ [Изменение направления обхода контура](#)^[227]

1.7.1.1 Контур построенный из элементов

Описание контура состоит из последовательности операторов , заключенных между заголовком контура и ограничителем - служебным словом **КОНКОН**.

Формат:

```
[ КОНТУР ] имя = [ ИЗ точка, модификатор ]
оператор1
оператор2
.....
операторn-1
операторn
КОНКОН
```

Параметры:

Параметр	Описание
<i>имя</i>	Имя, присвоенное контуру.
<i>точка</i>	Начальная точка контура.
<i>модификатор</i>	Любой из модификаторов ХБ-ХМ-УБ-УМ , задающий направление, относительно которого выбирается оператор1.
<i>оператор_p ... оператор_n</i>	Участки контура, заданные операторами движения, фрагментами других контуров, специальные операторы.

Каждый *оператор_i* в описании контура имеет формат:

```
{ оператор поточечного движения }
{ оператор непрерывного движения }
{ оператор включения контура }
{ оператор постпроцессора }
{ вспомогательный оператор }
```

Начальная точка контура и модификатор являются обязательными в случае, когда оператор, задающий первый участок контура, является оператором поточечного или непрерывного движения.

Если первый оператор является оператором включения контура, начальную точку можно не задавать. Начальной точкой является в этом случае начало встраиваемого контура или

фрагмента контура. Если же начальная точка контура задана, то встраиваемый контур или фрагмент переносится своим началом в указанную начальную точку.

Операторы *постпроцессора* в определении контура можно использовать по обычным правилам. При этом следует соблюдать осторожность в том случае, когда направление обхода контура не соответствует направлению, выбранному при его описании, т.к. оператор постпроцессора, записанный перед некоторым сегментом контура, будет в действительности относиться к предыдущему сегменту.

Вспомогательные операторы дают возможность включить в описание контура циклы, условные операторы, геометрические определения, фрагменты текста, вызовы макросов.

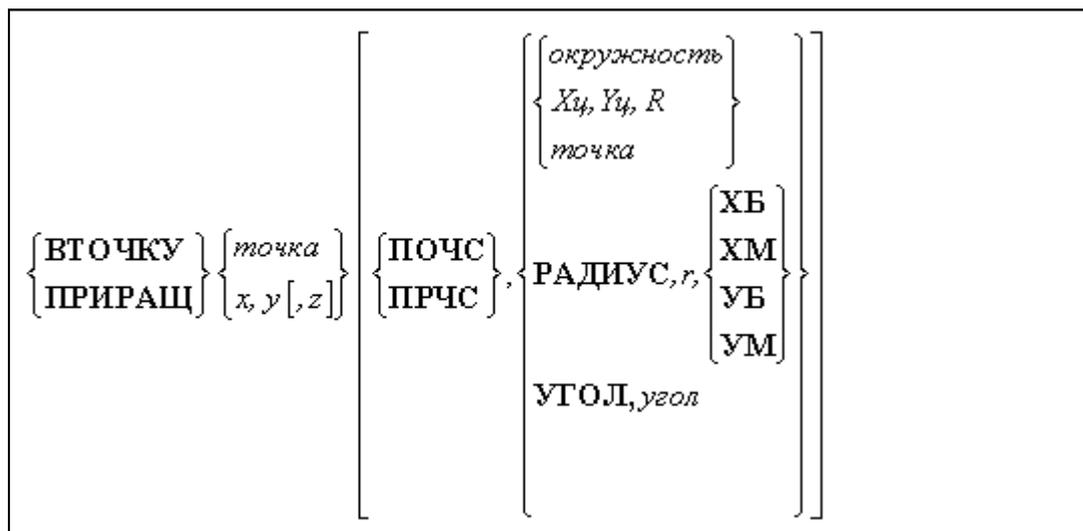
Темы этого раздела:

- [Оператор поточечного движения в определении контура](#)^[208]
- [Оператор непрерывного движения в определении контура](#)^[209]
- [Оператор включения контура](#)^[212]

1.7.1.1.1 Оператор поточечного движения в описании контура

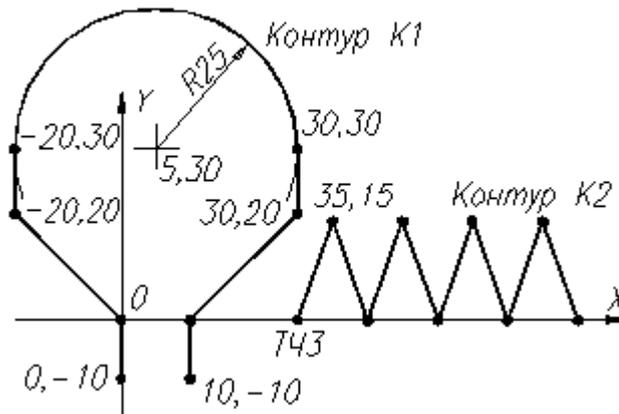
Поточечное движение в определении контура задается оператором **ВТОЧКУ**.

Формат:



Пример:

На следующем рисунке изображены контуры **К1** и **К2**, описания которых приведены ниже.



Контур **К1** программируется с использованием операторов поточечного движения следующим образом:

КОНТУР К1=ИЗ ТОЧКА(0,-10),УБ
ВТОЧКУ 0,0:-20,20
ПРИРАЩ 0,10
ВТОЧКУ 30,30 ПОЧС,ОКРУЖН(5,30,25)
ПРИРАЩ 0,-10:-20,-20:0,-10
КОНКОН

Контур **К2** программируется с использованием операторов поточечного движения и цикла следующим образом:

КОНТУР К2=ИЗ ТЧЗ,ХБ
ПОВТОР А=1,4
ПРИРАЩ 5,15:5,-15
КОНЦИКЛ А
КОНКОН

См. также:

- [Поточечное движение](#)^[159]

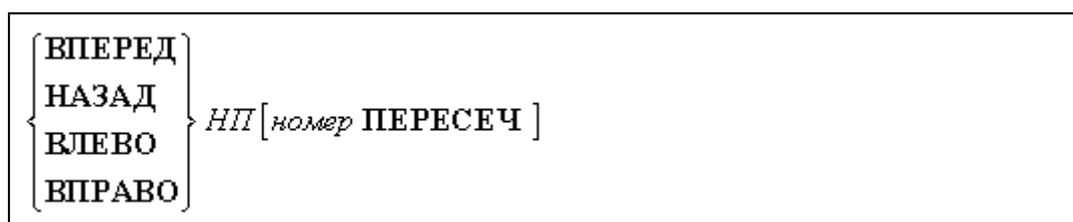
1.7.1.1.2 Операторы непрерывного движения в описании контура

Непрерывное движение в определении контура задается стандартными операторами **ВПЕРЕД, НАЗАД, ВЛЕВО, ВПРАВО**.

Формат оператора с двумя поверхностями:

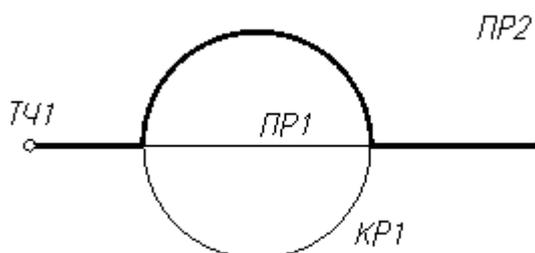
$$\left\{ \begin{array}{l} \text{ВПЕРЕД} \\ \text{НАЗАД} \\ \text{ВЛЕВО} \\ \text{ВПРАВО} \end{array} \right\} \text{НП} \left\{ \begin{array}{l} \text{ДО} \\ \text{НА} \\ \text{ЗА} \\ \text{КАС} \end{array} \right\} \left[\text{номер} [\text{ПЕРЕСЕЧ}] \right] \text{ОП}$$

Формат оператора с одной поверхностью:



Первый оператор движения в описании контура выбирается в соответствии с модификатором, указанным в заголовке контура. Проще всего, в качестве первого оператора использовать оператор **ВПЕРЕД**. Направление движения в последующих операторах указывается относительно предыдущего направления движения по правилам непрерывного движения.

В следующем примере приведено определение незамкнутого контура.



```

КОНТУР КОН1 = ИЗ ТЧ1, ХБ
  ВПЕРЕД ПР1
  ВЛЕВО КР1 2 ПЕРЕСЕЧ
  ВЛЕВО ПР1 НА ПР2
КОНКОН
  
```

Описание того же контура, но с использованием условного оператора для выбора обхода верхней или нижней полуокружности - в зависимости от знака переменной **Р**:

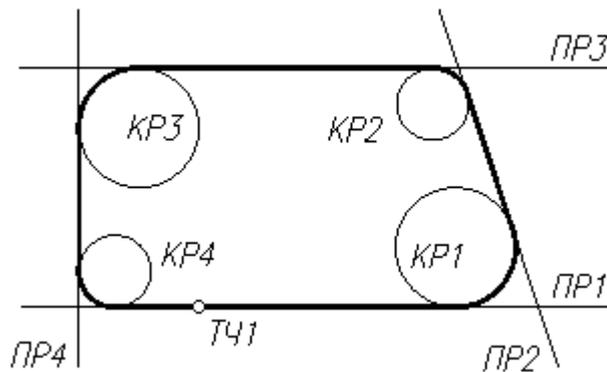
```

Р=-14.5
КОНТУР КОН1 = ИЗ ТЧ1, ХБ
  ВПЕРЕД ПР1
  ЕСЛИ (Р<0) ТО
    ВЛЕВО КР1 2 ПЕРЕСЕЧ
    ВЛЕВО ПР1 НА ПР2
  ИНАЧЕ
    ВПРАВО КР1 2 ПЕРЕСЕЧ
    ВПРАВО ПР1 НА ПР2
  КОНЕСЛИ
КОНКОН
  
```

При описании контура с использованием операторов непрерывного движения с одной поверхностью:

- Не требуется задавать операторы **ИНСТЛВ-ИНСТНА-ИНСТПР**, как это требуется при программировании траектории.
- Последний оператор движения в описании незамкнутого контура должен обязательно иметь явно заданную ограничивающую поверхность.
- Первый и последний операторы движения в описании замкнутого контура должны содержать в качестве направляющей одну и ту же поверхность.

В следующем примере приведено определение замкнутого контура с помощью операторов движения с одной поверхностью.



КОНТУР КОН2 = ИЗ ТЧ1, ХБ

**ВПЕРЕД ПР1
 ВПЕРЕД КР1
 ВПЕРЕД ПР2
 ВПЕРЕД КР2
 ВПЕРЕД ПР3
 ВПЕРЕД КР3
 ВПЕРЕД ПР4
 ВПЕРЕД КР4
 ВПЕРЕД ПР1**

КОНКОН

В определении контура допускается сокращенная запись операторов движения. Запись того же контура в сокращенной форме имеет вид:

КОНТУР КОН2 = ИЗ ТЧ1, ХБ

**ВПЕРЕД ПР1:КР1:ПР2:КР2:ПР3:КР3:ПР4:КР4:ПР1
 КОНКОН**

См. также :

- [Непрерывное движение](#)^[162]
- [Продолжение непрерывного движения](#)^[169]

1.7.1.1.3 Оператор включения контура

Оператор включения контура позволяет присоединять к определяемому контуру ранее определенный контур или фрагмент, ограниченный заданными точками.

Формат:

$$\left[\begin{array}{l} \text{ПО} \\ \text{ОБРАТН} \end{array} \right] \left[\text{матрица.} \right] \left[\text{ОТДО} \left\{ \begin{array}{l} \text{номер}_1 \\ \text{точка}_1 \end{array} \right\}, \left\{ \begin{array}{l} \text{номер}_2 \\ \text{точка}_2 \end{array} \right\}, \right] \text{контур}$$

Параметры:

Параметр	Описание
ПО	Присоединяемый контур или его часть не изменяет направления и присоединяется к последнему сегменту составного контура своим началом.
ОБРАТН	Присоединяемый контур или его часть изменяет направление на противоположное и присоединяется к последнему сегменту составного контура своим концом.
<i>матрица</i>	Матрица, задающая преобразование присоединяемого контура или фрагмента.
ОТДО	Присоединение фрагмента контура, ограниченного двумя точками.
<i>номер₁</i>	Номер начальной опорной точки присоединяемого фрагмента контура.
<i>номер₂</i>	Номер конечной опорной точки присоединяемого фрагмента контура.
<i>точка₁</i>	Имя начальной точки присоединяемого фрагмента контура.
<i>точка₂</i>	Имя конечной точки присоединяемого фрагмента контура.
<i>контур</i>	Имя присоединяемого контура.

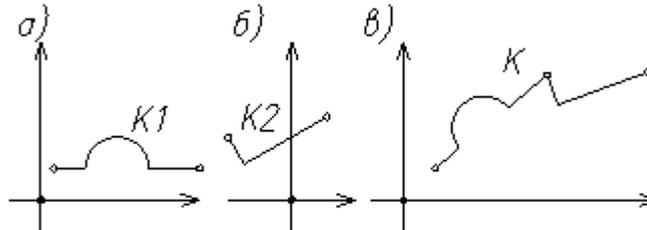
Матрица - любая плоская матрица -матрица поворота, симметрии или матрица, задающая их сочетание. Сдвиг, даже если он задается матрицей, не учитывается, т.к. контур все равно присоединяется своим началом или концом.

При включении контура выполняются следующие действия:

- выделяется фрагмент контура в соответствии с заданными границами;
- изменяется направление выделенного фрагмента или всего контура -в соответствии с модификатором **ПО(ОБРАТН)**;
- полученный контур преобразовывается в соответствии с матрицей;
- полученный контур пристыковывается своим началом к концу предыдущего.

Несколько операторов включения контура могут быть объединены в одной строке через запятую. Если при этом все определение размещается в одной строке, ограничитель **КОНКОН** задавать не требуется.

В следующем примере приведено определение контура **К(в)**, составленного из контуров **К1** и **К2** (*a* и *б*).



МАТР М1 = ПОВОРОТ,45

КОНТУР К1 = ИЗ ТЧ1, ...

.....

КОНКОН

КОНТУР К2 = ИЗ ТЧ2, ...

.....

КОНКОН

КОНТУР К =

ПО, М1,К1

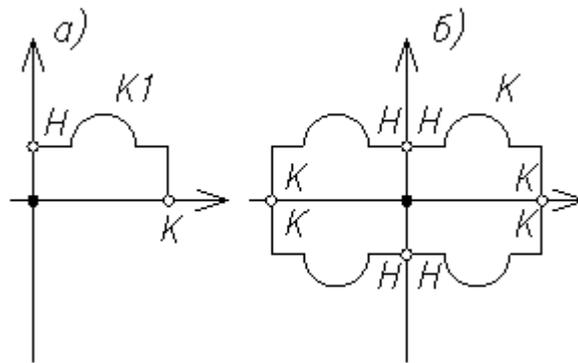
ПО, К2

КОНКОН

Описание того же контура в более компактной форме:

КОНТУР К = М1, К1, К2

Следующий пример иллюстрирует построение контура (*б*), который получается симметричными преобразованиями исходного контура (*a*).

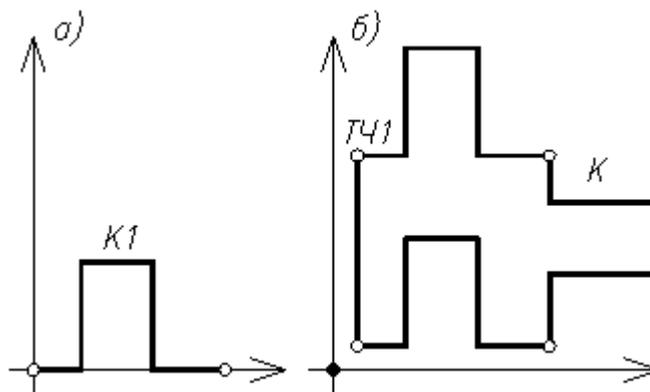


КОНТУР К, К1
МАТР МХ=ПРХ
МАТР МУ=ПРУ
МАТР МХУ=М1, М2
К1=
К=
ПО,К1
ОБРАТН, МХ, К1
ПО, МХУ, К1
ОБРАТН, МУ, К1
КОНКОН

Вариант описания того же контура:

К=К1, ОБРАТН, МХ, К1, МХУ, К1, ОБРАТН, МУ, К1

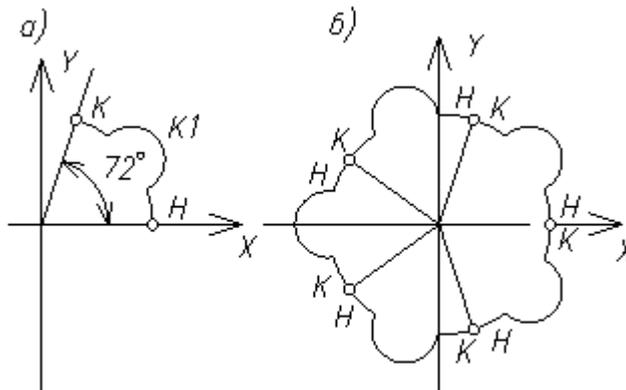
Следующий пример иллюстрирует совместное применение в определении контура оператора вставки контура и оператора поточечного движения.



МАТР М1=ПОВХУ,-90
КОНТУР К=ИЗ ТЧ1,ХБ
К1
М1,К1

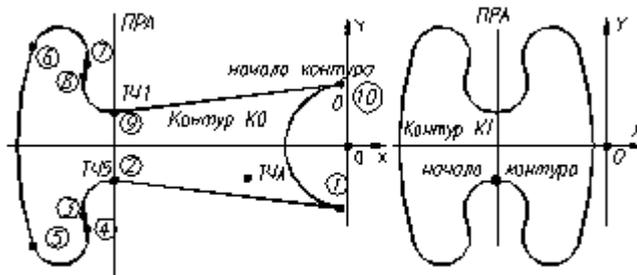
**ОБРАТН,К1
ВТОЧКУ ТЧ1
КОНКОН**

В определениях контуров можно использовать операторы **ЕСЛИ** и **ПОВТОР**. Пример построения контура **К(б)**, получающегося многократным повторением контура **К1**, приведен ниже.



**КОНТУР К=
ПОВТОР И = 1,5
МАТР (ПОВХУ, 72*(И -1)), К1
КОНЦИКЛ
КОНКОН**

Пример составного контура с использованием фрагментов ранее определенного контура иллюстрируется следующим рисунком:



Первым фрагментом составного контура **К1** является участок контура **К0** с началом в точке **ТЧ5** (опорная точка 2) и концом в **ТЧ1** (опорная точка 9). Вторым фрагментом составного контура является этот же участок, но после симметричного преобразования. Начало и конец его будут в тех же точках. Для того, чтобы он состыковался с предыдущим участком своим началом, применяется модификатор **ОБРАТН**:

МАТР М1=ПРА

K1=
ОТДО,2,9,K0
ОБРАТН,М1,ОТДО,ТЧ5,9,K0
КОНКОН

1.7.1.2 Контур являющийся сплайном

Сплайн (табулированная кривая, таблично заданная кривая) - это гладкая кривая, интерполирующая заданный набор точек и состоящая из отрезков прямых и дуг окружностей. Табулированная кривая имеет направление от первой заданной точки к последней и при построении кривой в обратном направлении результаты совпадать не будут.

Табулированные кривые имеют тип **КОНТУР** и с ними можно осуществлять любые операции, имеющие смысл для незамкнутых контуров. Так, например, они могут быть использованы как фрагменты составного контура.

Существует два формата определения табулированной кривой: однострочный и многострочный. Первый формат используется при задании сплайна перечислением опорных точек.

Формат:

[**КОНТУР**] *имя* = **СПЛАЙН** [,*прямая1*] , *точка1* , *точка2* , ... , *точкаn* [,*прямая2*]

Параметры:

Параметр	Описание
СПЛАЙН	Задание способа построения контура (интерполяция набора точек сплайном).
<i>точка1</i> <i>точка2</i> ... <i>точкан</i>	Опорные точки кривой.
<i>прямая1</i> <i>прямая2</i>	Имена необязательных прямых, касательных к кривой в начальной и конечной точках.

Пример:

КОНТУР ТК = СПЛАЙН, ПР1, Т1, Т2, Т3, Т4, Т5, Т6, ПР2

Многострочный формат позволяет включать в определение сплайна массивы точек, условные операторы, операторы цикла, операторы постпроцессора.

Формат:

```
[ КОНТУР ] имя = СПЛАЙН [ ,прямая1 ]
оператор1
оператор2
...
операторn
[ ,прямая2 ]
КОНКОН
```

Параметры:

Параметр	Описание
СПЛАЙН	Задание способа построения контура (интерполяция набора точек сплайном).
оператор ₁ ... оператор _n	Опорные точки кривой, специальные операторы.
прямая ₁ прямая ₂	Имена необязательных прямых, касательных к кривой в начальной и конечной точках.

Каждый *оператор_i* в описании контура имеет формат:

```
{
  точка
  оператор постпроцессора
  вспомогательный оператор
}
```

Операторы *постпроцессора* в определении контура можно использовать по обычным правилам. При этом следует соблюдать осторожность в том случае, когда направление обхода контура не соответствует направлению, выбранному при его описании, т.к. оператор постпроцессора, записанный перед некоторым сегментом контура, будет в действительности относиться к предыдущему сегменту.

Вспомогательные операторы дают возможность включить в описание контура циклы, условные операторы, геометрические определения, фрагменты текста, вызовы макросов.

Пример:

Оператор **ПОВТОР** в определении табулированной кривой.

```
КОНТУР Т1 = СПЛАЙН, ПР1
  ПОВТОР И = 1,50
    ТЧ(И)
  КОНЦИКЛ И
  ПР2
КОНКОН
```

Оператор ЕСЛИ в определении табулированной кривой.

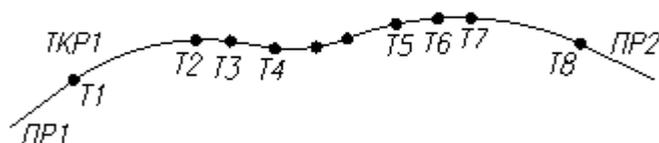
```

КОНТУР Т2 = СПЛАЙН, ПР1
  ЕСЛИ (A>=B) ТО
    ТЧ1, ТЧ2, ТЧ3
  ИНАЧЕ
    ТЧ4, ТЧ5, ТЧ6
  КОНЕСЛИ
  ПР2
КОНКОН

```

Оператор цикла, условный оператор и перечисление точек могут встречаться в определениях табулированной кривой в различных комбинациях.

Запись сплайна (см. рисунок) имеет следующий вид:



```

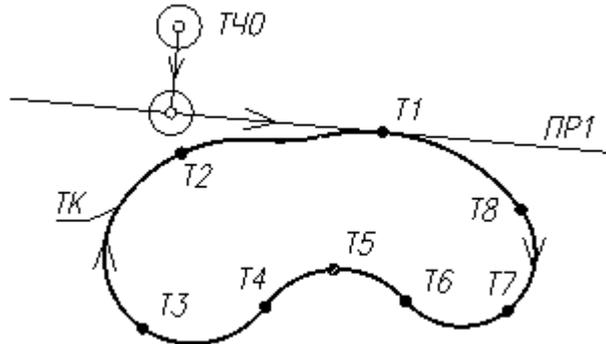
КОНТУР ТКР1 = СПЛАЙН, ПР1
  ТЧ1, ТЧ2, ТЧ3
  ЕСЛИ (A>=B) ТО
    ПОВТОР И=1, К
      ТЧ(И)
    КОНЦИКЛ И
  ИНАЧЕ
    ПОВТОР И=1, М
      ЕСЛИ (Л=1) ТО
        ТК(И)
      ИНАЧЕ
        ТК(М-И)
      КОНЕСЛИ
    КОНЦИКЛ И
  КОНЕСЛИ
  ТЧ4, ТЧ5, ТЧ6
  ПР2
КОНКОН

```

Табулированная кривая обязательно проходит через все опорные точки и, в случае неточного задания хотя бы одной из них, возможны “биения” - локальные искажения формы

кривой. По этой причине рекомендуется аналитическое задание точек - т.е. использование для расчета их координат формул, записанных в виде выражений на Техтроне.

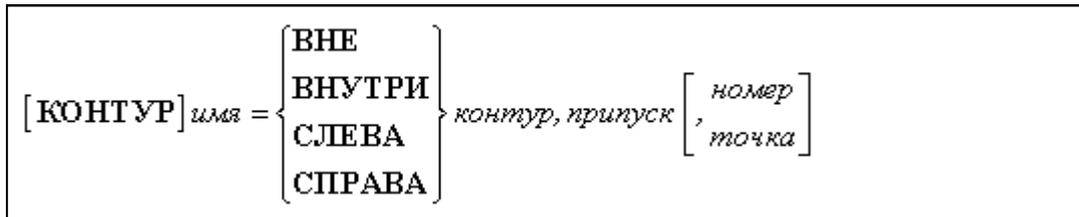
При построении замкнутой табулированной кривой необходимо в качестве начальной и конечной указывать одну и ту же точку, а также общую касательную, если не допускается излом в этой точке.



КОНТУР ТК = СПЛАЙН, ПР1, Т1, Т2, Т3, Т4, Т5, Т6, Т7, Т8, Т1, ПР1

1.7.1.3 Контур эквидистантный заданному

Формат:



Параметры:

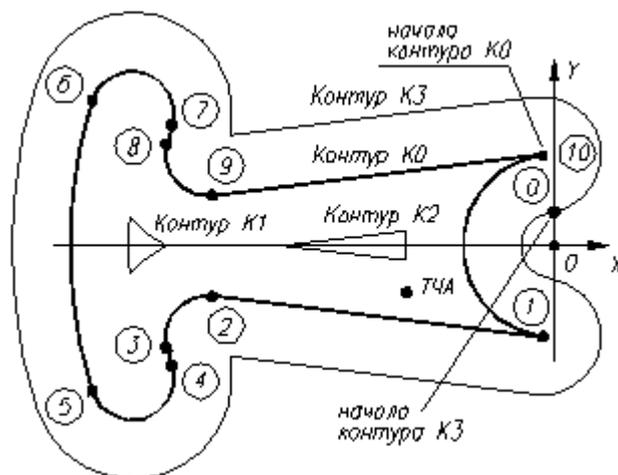
Параметр	Описание
<i>контур</i>	Имя контура, к которому строится эквидистанта.
<i>припуск</i>	Расстояние от контура до эквидистанты.
ВНЕ	Эквидистанта расположена вне контура.
ВНУТРИ	Эквидистанта расположена внутри контура.
СЛЕВА	Эквидистанта расположена слева от контура.
СПРАВА	Эквидистанта расположена справа контура.
<i>номер</i>	Номер опорной точки, ближайшей к требуемой эквидистанте.

Параметр	Описание
<i>точка</i>	Опорная точка, ближайшая к требуемой эквидистанте.

Необязательный параметр необходим в случае, когда эквидистанта распадается на две и более. Он задает точку, ближайшую к нужной части эквидистанты.

Пример:

В примере определяется внешняя и внутренняя эквидистанты к контуру **К0**. Внутренняя эквидистанта распадается на две. В определении контура **К1** необходимо задать **ТЧА**, ближайшую к правой части эквидистанты.



ПЕТЛИ ВКЛ
СГЛАЖ ВКЛ
К1=СПРАВА,К0,35,ТЧА
К2=СПРАВА,К0,35,5
К3=ВНЕ,К0,35

При построении эквидистанты действуют режимы, установленные операторами **ПЕТЛИ** и **СГЛАЖ**.

1.7.1.4 Контур, имеющий форму прямоугольника

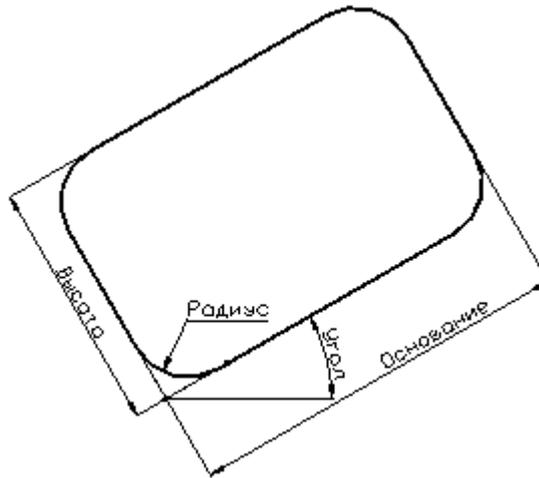
Формат:

`имя = ПРЯМУГОЛ, точка, РАЗМЕР, основ, высота [,СКРУТ, радиус][,ПОВОРОТ, угол]`

Параметры:

Параметр	Описание
<i>точка</i>	Базовая точка контура.
<i>основ</i>	Основание прямоугольника.

Параметр	Описание
<i>высота</i>	Высота прямоугольника.
<i>радиус</i>	Радиус скругления.
<i>угол</i>	Угол поворота контура относительно оси X.



Пример:

K0=ПРЯМУГОЛ,ТЧ10,РАЗМЕР,100,70,СКРУГ,10,ПОВОРОТ,30

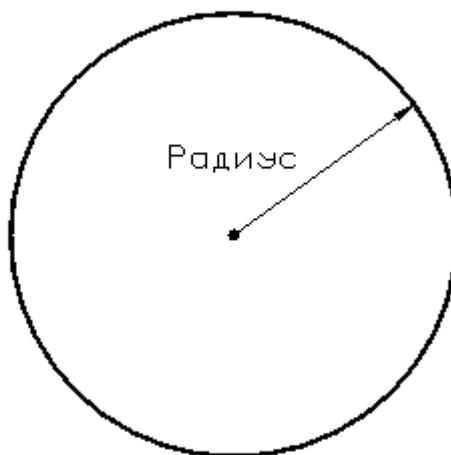
1.7.1.5 Контур, имеющий форму окружности

Формат:

имя = *окружность* [,**ПОВОРОТ**, *угол*]

Параметры:

Параметр	Описание
<i>окружность</i>	Окружность, определяющая контур.
<i>угол</i>	Угол поворота контура относительно оси X. Поворот осуществляется относительно центра окружности



Пример:

K10=KР1

K11=ОКРУЖН(ТОЧКА(0,0),50),ПОВОРОТ,30

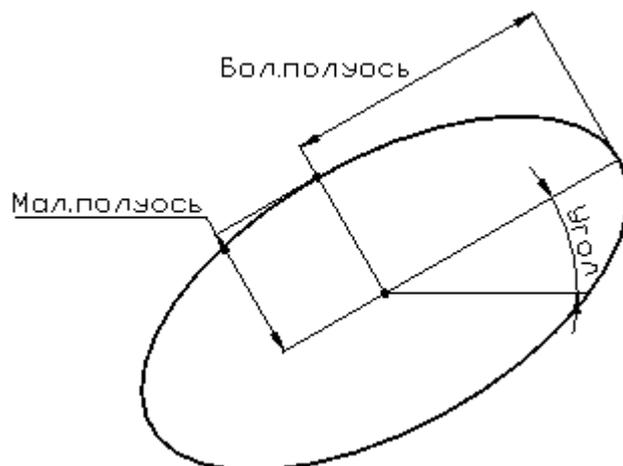
1.7.1.6 Контур, имеющий форму эллипса

Формат:

`имя = ЭЛЛИПС, точка, ПОЛУОСЬ, полуось1, полуось2 [, ТОЧНОСТЬ, точн] [, ПОВОРОТ, угол]`

Параметры:

Параметр	Описание
<i>точка</i>	Центр эллипса.
<i>полуось₁, полуось₂</i>	Полуоси эллипса.
<i>точн</i>	Точность построения эллипса. Задает максимальное отклонение от математической кривой.
<i>угол</i>	Угол поворота контура относительно оси X.



Пример:

K5=ЭЛЛИПС,ТЧ2,ПОЛУОСЬ,100,60,ПОВОРОТ,45

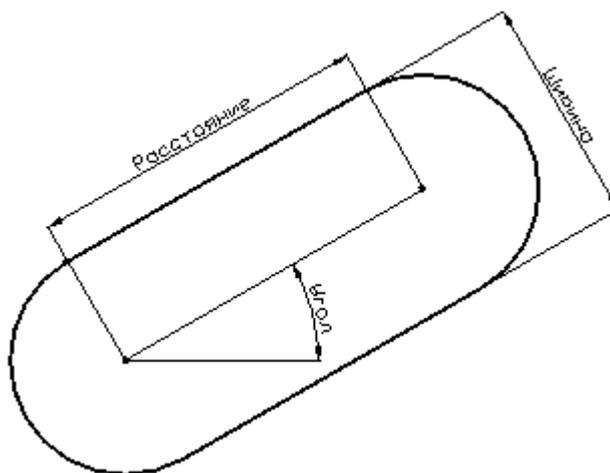
1.7.1.7 Контур, имеющий форму прямого паза

Формат:

имя = ПАЗ, точка, РАСТ, расстояние, ШИРИНА, ширина [,ПОВОРОТ, угол]

Параметры:

Параметр	Описание
<i>точка</i>	Базовая точка (центр окружности).
<i>расстояние</i>	Расстояние между центрами окружностей.
<i>ширина</i>	Ширина паза.
<i>угол</i>	Угол поворота контура относительно оси X.



Пример:

К7=ПАЗ,ТЧЗ,РАССТ,100,ШИРИНА,40,ПОВОРОТ,90

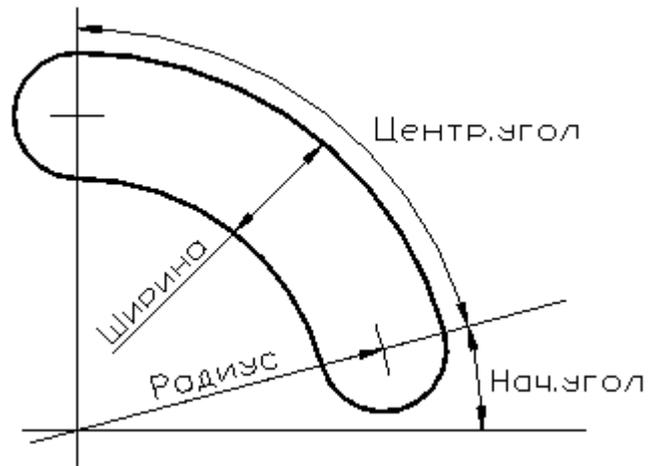
1.7.1.8 Контур, имеющий форму радиусного паза

Формат:

имя = ПАЗ, точка, РАДИУС, радиус, УГОЛ, угол_{центр}, ШИРИНА, ширина [,ПОВОРОТ, угол]

Параметры:

Параметр	Описание
<i>точка</i>	Базовая точка (центр окружности).
<i>радиус</i>	Радиус окружности.
<i>ширина</i>	Ширина паза.
<i>угол_{центр}</i>	Центральный угол
<i>угол</i>	Угол наклона прямой, проведенной через центр паза и центр скругляющей окружности, к оси X.



Пример:

K8=ПАЗ,ТЧ2,РАДИУС,100,УГОЛ,45,ШИРИНА,40,ПОВОРОТ,15

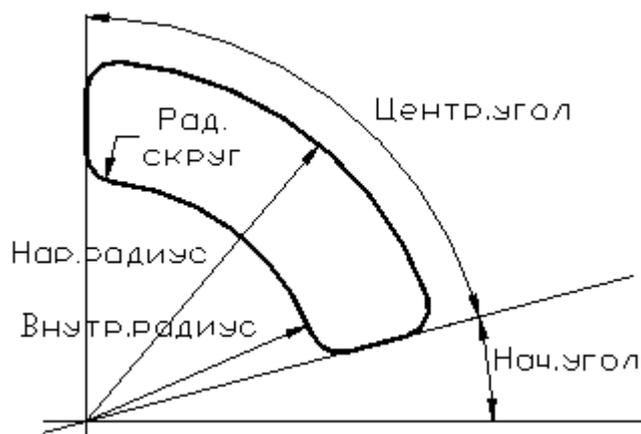
1.7.1.9 Контур, имеющий форму сектора

Формат:

имя = СЕКТОР, точка, РАДИУС, радиус_{нар}, радиус_{внутр}, УГОЛ, угол_{центр}, СКРУТ, радиус [,ПОВОРОТ, угол]

Параметры:

Параметр	Описание
<i>точка</i>	Базовая точка (центр окружности).
<i>радиус_{нар}</i>	Наружный радиус.
<i>радиус_{внутр}</i>	Внутренний радиус.
<i>радиус</i>	Радиус скругления.
<i>ширина</i>	Ширина паза.
<i>угол_{центр}</i>	Центральный угол.
<i>угол</i>	Угол наклона к оси X.



Пример:

K10=СЕКТОР,ТЧЗ,РАДИУС,120,100,УГОЛ,90,СКРУГ,5,ПОВОРОТ,30

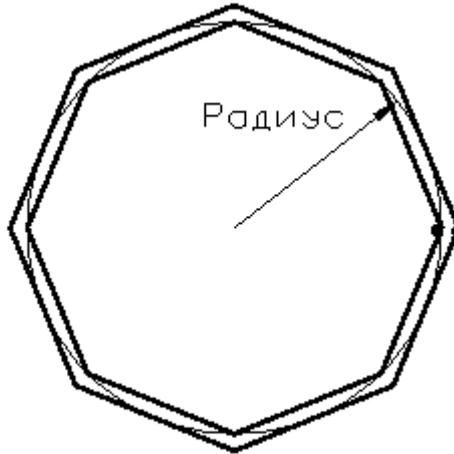
1.7.1.10 Контур, имеющий форму правильного многоугольника

Формат:

имя = **МНОГУГОЛ**, *число*, { **ВНЕ**
ВНУТРИ }, *окружность* [, **ПОВОРОТ**, *угол*]

Параметры:

Параметр	Описание
<i>число</i>	Число сторон.
<i>окружность</i>	Окружность, определяющая многоугольник
ВНЕ	Описанный многоугольник.
ВНУТРИ	Вписанный многоугольник
<i>угол</i>	Угол поворота контура относительно оси X. Поворот осуществляется относительно центра вписанной(описанной) окружности



Пример:

K8=МНОГУГОЛ,8,ВНЕ,КР1

K9=МНОГУГОЛ,8,ВНУТРИ,ОКРУЖН(ТЧ3,50)

1.7.1.11 Габаритный контур

Формат:

$$\text{имя} = \text{ПРЯМУГОЛ}, \left\{ \begin{array}{l} \text{контур} \\ \text{массив контуров} \end{array} \right\} [\text{СМЕЩЕНИЕ}, \text{раст}]$$

Параметры:

Параметр	Описание
<i>контур</i>	Контур
<i>массив контуров</i>	Массив контуров
СМЕЩЕНИЕ, <i>раст</i>	Величина смещения для габаритного контура

Пример:

КОНТУР K8=ПРЯМУГОЛ,К4,СМЕЩЕНИЕ,1

1.7.1.12 Изменение направления обхода контура

Формат:

$$\text{контур} = \left\{ \begin{array}{l} \text{ОБРАТН} \\ \text{ПОЧС} \\ \text{ПРЧС} \end{array} \right\}, \text{контур}$$

Параметры:

Параметр	Описание
<i>контур</i>	Контур, направление которого надо изменить.
ОБРАТН	Обход контура в порядке, противоположном описанию сегментов контура
ПОЧС	Обход контура по часовой стрелке (только для замкнутых контуров)
ПРЧС	Обход контура против часовой стрелки (только для замкнутых контуров)

При изменении направления обхода массива контуров операторы надо формировать для каждого элемента массива.

Пример:

K6=ОБРАТН,K6
K4(1)=ПОЧС,K4(1)
K4(2)=ПОЧС,K4(2)

1.7.1.13 Замыкание контура путем сопряжения его концов

Формат:

ЗАМКОН, контур [,ТОЧНОСТЬ, t]

Параметры:

Параметр	Описание
<i>контур</i>	Контур, который подлежит замыканию
ТОЧНОСТЬ, t	Точность сопряжения, используемая при замыкании

Производит замыкание контура путем сопряжения его концов или усечения самопересекающегося контура.

1.7.2 Движение по контуру

Движение по контуру задается аналогично непрерывному движению по элементарным геометрическим объектам. Сначала инструмент должен быть подведен к контуру при помощи оператора **ИДИ** или операторов непрерывного движения. Затем операторами непрерывного движения задается полный или частичный его обход.

Темы этого раздела:

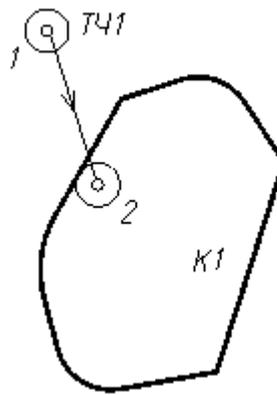
- [Подход к контуру](#)^[229]
- [Применение операторов непрерывного движения](#)^[230]
- [Движение по незамкнутому контуру](#)^[232]
- [Движение по табулированной кривой](#)^[233]

1.7.2.1 Подход к контуру

Существует два способа подхода к контуру:

- при помощи оператора **ИДИ**;
- при помощи операторов непрерывного движения **ВПЕРЕД, НАЗАД, ВЛЕВО, ВПРАВО**.

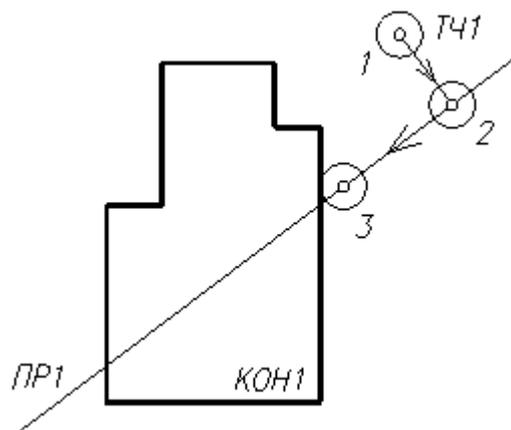
При использовании оператора **ИДИ** следует указывать направление подхода к контуру операторами **НАПРВ** или **НАПРТ**.



ИЗ ТЧ1
НАПРВ В1
ИДИ ЗА К1

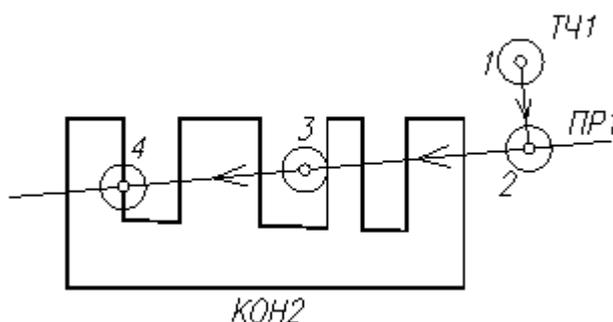
При использовании операторов непрерывного движения инструмент следует вывести на контур с помощью любой команды непрерывного движения с двумя поверхностями. Число пересечений не должно превышать восьми. Если поверхность подхода совпадает с одним из сегментов контура, то инструмент выводится в первую или последнюю точку сегмента.

В следующий примере показан подход к контуру по прямой.



ИЗ ТЧ1
ИДИ НА ПР1
ВПРАВО ПР1 ДО КОН1

В случае многократных пересечений вывод инструмента осуществляется следующим образом.



перемещение 1-2-3:

ИЗ ТЧ1
ИДИ НА ПР1
ВПРАВО ПР1 ЗА 4 ПЕРЕСЕЧ КОН2

перемещение 1-2-4:

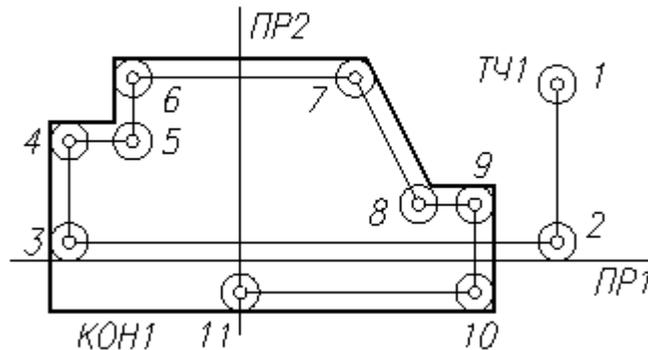
ИЗ ТЧ1
ИДИ НА ПР1
ВПРАВО ПР1 НА 7 ПЕРЕСЕЧ КОН2

1.7.2.2 Применение операторов непрерывного движения

Для программирования движения инструмента по контуру можно задать любой из операторов непрерывного движения с двумя поверхностями **ВПЕРЕД**, **НАЗАД**, **ВПРАВО**, **ВЛЕВО**, указав требуемое число пересечений. В качестве *НП* в них используется контур,

в качестве *ОП* - прямая, окружность, контур.

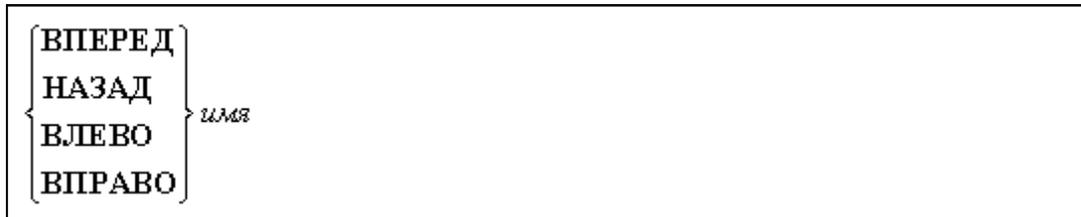
Например, для контура, показанного на следующем рисунке, перемещение инструмента из позиции 1 в позиции 2 - 11 программируется так:



**ИЗ ТЧ1
ИДИ ДО ПР1
ВПРАВО ПР1 ДО 2 ПЕРЕСЕЧ КОН1
ВПРАВО КОН1 НА 2 ПЕРЕСЕЧ ПР2**

Применение оператора с одной направляющей поверхностью допустимо только для замкнутого контура и задает его полный обход, при котором инструмент возвращается в точку подхода.

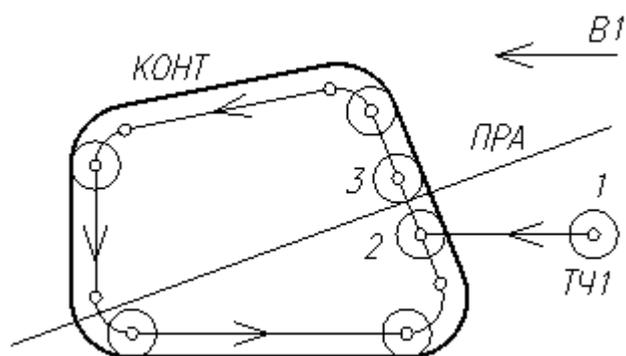
Формат:



Параметры:

Параметр	Описание
<i>имя</i>	Символическое имя контура.

Например, полный обход контура, представленный на следующем рисунке, задается операторами:



ИЗ ТЧ1	
НАПРВ В1	
ИДИ ЗА КОНТ	# поз 1-2
ВПРАВО КОНТ ЗА ПРА	# поз 2-3
ВПЕРЕД КОНТ	# поз 3-2-3

Необходимо помнить, что при непрерывном движении по контуру не только сами поверхности, но и их эквидистанты должны иметь требуемое число пересечений.

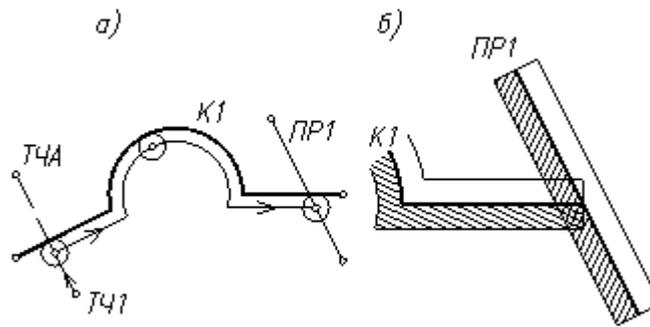
См. также:

- [Непрерывное движение](#)^[162]

1.7.2.3 Движение по незамкнутому контуру

Необходимо помнить, что при непрерывном движении по контуру не только сами поверхности, но и их эквидистанты должны иметь требуемое число пересечений. Движение по незамкнутому контуру имеет свои особенности, обусловленные тем, что эквидистанта незамкнутого существуют в области, ограниченной самим контуром и нормальными к контуру в граничных его точках. В связи с этим, может возникнуть ситуация, когда эквидистанта незамкнутого контура и эквидистанта ограничивающей поверхности не пересекаются. Это чаще всего происходит в тех случаях, когда ОП находится вблизи граничной точки незамкнутого контура.

Например, движение по объектам, представленным на рисунке, задано следующими операторами.



**ИЗ ТЧ1
НАПРТ ТЧА
ИДИ ДО К1
ВПРАВО К1 НА ПР1**

В случае, когда *ОП*, заданная прямой **ПР1**, гарантированно пересекает незамкнутый контур (рис. *а*), не возникает осложнений при расчете траектории.

Если **ПР1** проходит через граничную точку контура (рис. *б*), то все эквидистанты контура, существующие в заштрихованной области, не пересекаются с этой прямой, поэтому возникнет ошибочная ситуация, которая сопровождается сообщением системы:

“Отсутствует требуемое число пересечений”

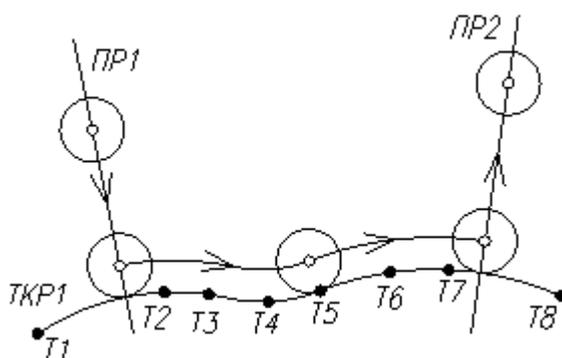
Также ошибочной будет ситуация при использовании в последнем операторе движения модификатора **ЗА (ВПРАВО К1 ЗА ПР1)** - т.к. эквидистанты тоже не пересекаются.

Если в последнем операторе приводимого текста программы будет использован модификатор **ДО**, то эквидистанты, существующие в заштрихованных областях, будут пересекаться и ошибки расчета траектории не произойдет.

1.7.2.4 Движение по табулированной кривой

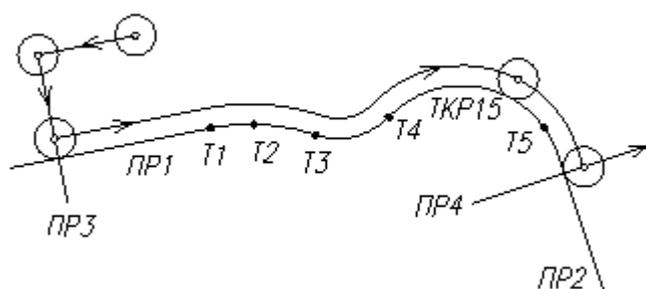
Поскольку табулированная кривая является разновидностью контура, то все сказанное выше о командах движения для контуров применимо и к табулированным кривым. Но, в связи с тем, что характерной особенностью сплайна является его гладкость и, как правило, требуется получение гладкой траектории, следует соблюдать приводимые ниже рекомендации.

Подход к сплайну и отход от него по секущей, не проходящей через граничные точки, является наиболее простым и надежным способом движения по табулированной кривой. Он позволяет исключить начальный и конечный участки сплайна, на которых наиболее вероятно возникновение биений.



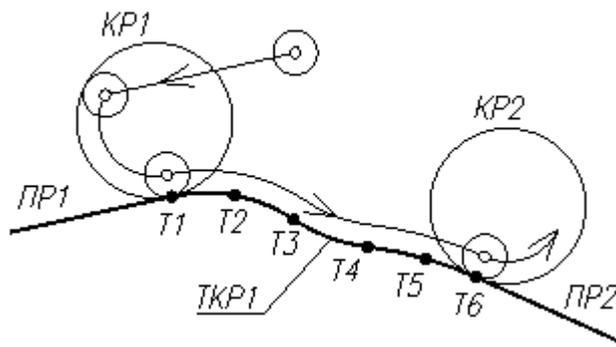
КОНТУР ТКР1=СПЛАЙН, T1, T2, T3, T4, T5, T6, T7, T8
ВПЕРЕД ПР1 ДО ТКР1
ВЛЕВО ТКР1 НА ПР2
ВЛЕВО ПР2 ...

Подход к сплайну по касательной осуществляется с использованием прямых, входящих в геометрическое определение сплайна и задающих касательные к нему в начальной и конечной точках.



КОНТУР ТКР15=СПЛАЙН, ПР1, T1, T2, T3, T4, T5, ПР2
ИДИ НА ПР3
ВЛЕВО ПР3 ДО ПР1
ВЛЕВО ПР1 КАС ТКР15
ВПЕРЕД ТКР15 КАС ПР2
ВПЕРЕД ПР2 НА ПР4

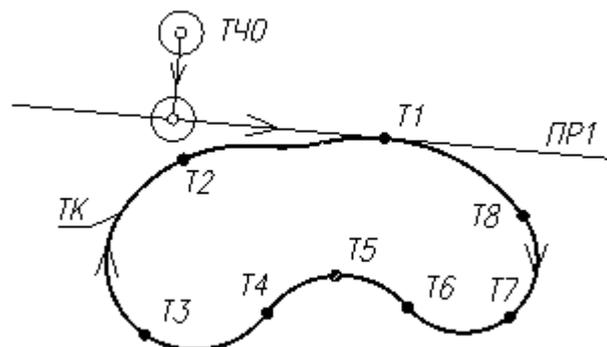
Выход по дуге окружности в начальную точку сплайна производится в следующей последовательности.



КОНТУР ТКР1=СПЛАЙН, ПР1, Т1, Т2, Т3, Т4, Т5, Т6, ПР2
 КР1=КАС, ПР1, УБ, Т1, РАДИУС, 20
 КР2=КАС, ПР2, УБ, Т6, РАДИУС, 20

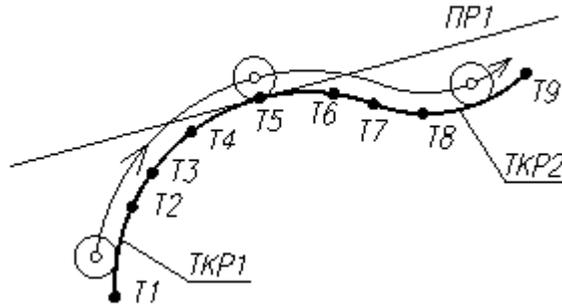
.....
 ВЛЕВО КР1 КАС ТКР1
 ВПЕРЕД ТКР1 КАС КР2
 ВПЕРЕД КР2

При подходе к замкнутому сплайну рекомендуется в качестве начальной и конечной указывать одну и ту же точку, а также общую касательную, если не допускается излома в этой точке.



КОНТУР ТК=СПЛАЙН, ПР1, Т1, Т2, Т3, Т4, Т5, Т6, Т7, Т8, Т1, ПР1
 ИЗ ТЧ0
 ИДИ НА ПР1
 ВЛЕВО ПР1 КАС ТК
 ВПЕРЕД ТК

Для гладкого сопряжения траектории при переходе с одного сплайна на другой, необходимо задать вырожденное движение по их общей касательной.



КОНТУР ТКР1, ТКР2

ТКР1=СПЛАЙН, Т1,Т2,Т3,Т4,Т5, ПР1

ТКР2=СПЛАЙН, ПР1, Т5,Т6,Т7,Т8,Т9

ВПЕРЕД ТКР1 КАС ПР1

ВПЕРЕД ПР1 КАС ТКР2

ВПЕРЕД ТКР2

1.7.3 Встраивание фасок и скруглений в контур

Для встраивания фасок в контур предназначен оператор **ФАСКА**.

Формат:

$$\text{контур}, \left\{ \begin{array}{l} \text{номер} \\ \text{точка} \end{array} \right\} : \left\{ \begin{array}{l} \text{ФАСКА, длина}_{\text{пред}}, \text{длина}_{\text{след}} \\ \text{ФАСКА} \left\{ \begin{array}{l} \text{ПО} \\ \text{ОБРАТН} \end{array} \right\}, \text{длина, угол} \end{array} \right\}$$

Параметры:

Параметр	Описание
<i>контур</i>	Имя контура, в который встраивается фаска.
<i>номер</i>	Номер точки контура, определяющей, где встраивается фаска.
<i>точка</i>	Имя точки контура, определяющей, где встраивается фаска.
<i>длина_{пред}</i>	Отступ по сегменту, предшествующему точке.
<i>длина_{след}</i>	Отступ по сегменту, следующему за точкой.
ПО	Отступ по описанию контура.

Параметр	Описание
ОБРАТН	Отступ против описания контура.
<i>длина</i>	Величина отступа.
<i>угол</i>	Угол наклона фаски к оси X.

При встраивании фаски, от точки контура откладываются отрезки или дуги, равные *длине след* и *длине пред*. Полученные точки соединяются отрезком. При задании фаски углом, от точки контура откладывается *длина* -по описанию контура или против его описания. Из полученной точки проводится отрезок под углом к оси X (см. пример):

Для встраивания скруглений между сегментами контура предназначен оператор **СКРУГ**.

Формат:

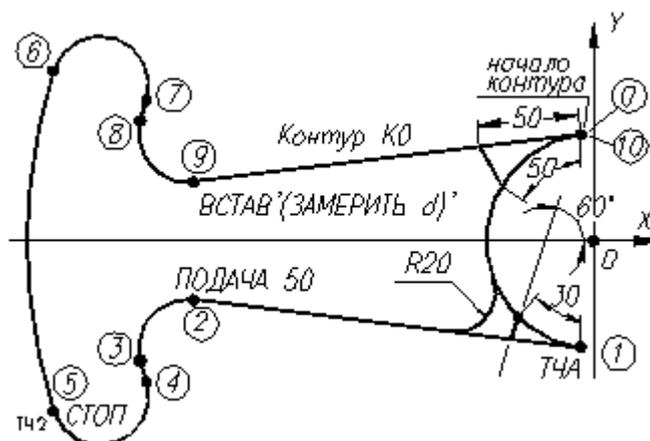
$\text{контур}, \left\{ \begin{array}{l} \text{номер} \\ \text{точка} \end{array} \right\} : \text{СКРУГ}, \text{радиус}$

Параметры:

Параметр	Описание
<i>контур</i>	Имя контура, в который встраивается скругление.
<i>номер</i>	Номер точки контура, определяющей , где встраивается скругление.
<i>точка</i>	Имя точки контура, определяющей, где встраивается скругление.
<i>радиус</i>	Радиус скругления.

Операторы **ФАСКА** и **СКРУГ** не вносят изменений в исходное описание контура: они только задают ссылку на точку контура и назначают вид и размеры встраиваемого участка, состоящего из одного сегмента (отрезка или дуги). Количество и нумерация сегментов и узлов изменяются - в связи с увеличением их на 1.

Пример:



К0,0:ФАСКА 50,50

К0,1:СКРУГ 20 или К0,ТЧА:СКРУГ 20

К0,ТЧА:ФАСКА, ОБРАТН, 30, 60

Скругления могут быть встроены между всеми сегментами контура. Для этого используется другой вариант оператора **СКРУГ**.

Формат:

СКРУГ *контур*, **РАДИУС**, *радиус*

Параметры:

Параметр	Описание
<i>контур</i>	Имя контура, в который встраиваются скругления.
<i>радиус</i>	Радиус скругления.

Скругления встраиваются между всеми соседними сегментами, если только это возможно. Действие оператора не распространяется на несмежные сегменты.

1.7.4 Траектория как контур

Объекты типа *ход* являются разновидностью *контуров*, поэтому они могут использоваться в том же контексте, что и *контур*, т.е. везде, где в формате указан объект типа *контур*, может быть использован объект типа *ход* и наоборот. Они могут применяться как:

- опорные элементы в геометрических определениях;
- *НП* и *ОП* в операторах движения;
- параметры стандартных функций;
- фрагменты составного контура;
- параметры операторов **ФАСКА** и **СКРУГ**;
- объекты при назначении технологии.

1.8 Макросы

При написании программ на Техтрane можно использовать отдельно подготовленные части исходного текста – **макросы**. В макросе разрешается применять любые операторы языка, но вместо конкретных значений переменных могут использоваться параметры.

Подключение макроса к программе осуществляется при помощи оператора **ВЫЗОВ**. При обработке системой текста, содержащего вызов макроса, вместо оператора **ВЫЗОВ** подставляется текст вызываемого макроса – тело макроса, в котором формальные параметры заменяются фактическими значениями, заданными в операторе **ВЫЗОВ**.

Текст макроса содержится, как правило, во внешнем файле. В этом случае макрос является доступным для любой программы, содержащей его вызов. Допускается определение макросов внутри программы. В этом случае вызов макроса должен производиться только из этой же программы.

По сравнению с оператором вставки текста (*), макросы являются мощным средством параметризации и программирования типовых технологических задач.

Темы этого раздела:

- [Структура макроса](#)^[239]
- [Параметры макроса](#)^[240]
- [Заголовок макроса](#)^[242]
- [Вызов макроса](#)^[243]
- [Вложенные вызовы](#)^[246]
- [Локализация переменных в макросе](#)^[247]
- [Использование меток в макросе](#)^[248]

1.8.1 Структура макросов

Макрос состоит из заголовка, тела и ограничителя.

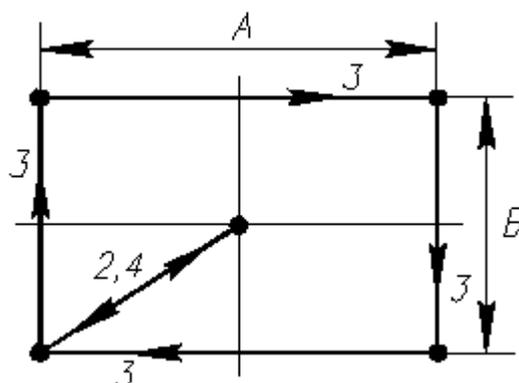
Заголовок макроса – оператор **МАКРО**. Заголовок может содержать имя макроса и значения параметров по умолчанию. Имя макроса обязательно при описании макроса внутри использующей его программы, т.к. только в этом случае это имя является идентификатором макроса.

Тело макроса – произвольная последовательность операторов Техтрane (кроме операторов **ДЕТАЛЬ** и **КОНЕЦ**), в которых могут быть использованы параметры макроса. Тело макроса может содержать другие макросы и обращения к ним -т.е. вложенные макросы. Операторы тела макроса являются описательными и исполняются только при вызове макроса.

Ограничитель макроса – служебное слово **КОНМАК**. Ограничитель обязателен при описании макроса внутри использующей его программы.

Пример.

В примере рассматривается простейший макрос, задающий обход прямоугольника со сторонами **А** и **Б**. Макрос написан с использованием в качестве параметров переменных головной программы.



ДЕТАЛЬ		1
МАКРО ПРЯМОУГ		2
ПРИРАЩ $-A/2, -B/2$		3
ПРИРАЩ $0, B; A, 0; 0, -B; -A, 0$		4
ПРИРАЩ $A/2, B/2$		5
КОНМАК		6
#		7
$A=100; B=50$		8
ВЫЗОВ ПРЯМОУГ		9
КОНЕЦ		10

Текст макроса находится в строках со 2 по 6. В заголовке макроса в строке 2 задано имя макроса **ПРЯМОУГ**. Тело макроса, занимающее строки с 3 по 5, содержит операторы, которые формируют начальное движение, обход сторон прямоугольника и возврат в исходное положение. Текст макроса запоминается системой без проверки; выполнение отложено до вызова макроса. Оператор **ВЫЗОВ** в строке 9 содержит только имя вызываемого макроса. Его исполнение равнозначно выполнению строк 3-5 тела макроса. Варьируя значения переменных **A** и **B** в головной программе, можно, не меняя текста макроса, изменять параметры траектории.

Если макрос из предыдущего примера находится во внешнем файле **ПРЯМОУГ.**, то программа будет иметь вид:

```

ДЕТАЛЬ
ВЫЗОВ PR.МАК(A=100, B=50)
КОНЕЦ

```

1.8.2 Параметры макроса

Параметры макроса – могут быть операторными и текстовыми. Параметры записываются в произвольном порядке и разделяются запятыми. При перечислении параметров действуют стандартные ограничения на длину строки и правила продолжения строк.

Операторный параметр представляет собой оператор Техтрана, заключенный в скобки. При вызове макроса операторные параметры выполняются непосредственно перед телом

макроста. В качестве операторных параметров запрещено использовать операторы **ДЕТАЛЬ, КОНЕЦ, МАКРО, КОНМАК, ВЫЗОВ, КОНТУР, КОНКОН, НАМЕТКУ, ПОВТОР, КОНЦИКЛ, ЕСЛИ**. Операторы присваивания, не содержащие разделителей в правой части, в скобки можно не заключать.

Примеры использования различных операторных параметров:

Операторный параметр -оператор простого присваивания:

ВЫЗОВ ТЕСТ(А=100, Б=50)

Операторный параметр -геометрическое определение:

ВЫЗОВ ТЕСТ((ПРА=ХПАР,100), (КР1=0,0,50))

Операторный параметр- логическое выражение, арифметическое выражение, специальный оператор, оператор объявления массива:

**ВЫЗОВ ТЕСТ((ЛОГИЧ П=ЛОЖЬ), А=12.5*3-14.3, %
(ПЧТ1 А), (ТОЧКА МТЧ(50))**

Текстовый параметр имеет свое имя и значение. Имя и значение текстового параметра заключаются в апострофы и соединяются знаком равенства. Текстовые параметры, кроме назначения в вызове макроса, могут быть заданы и в заголовке макроса. В этом случае они определяют значения текстовых переменных по умолчанию -значения, которые будут использованы в случае их отсутствия в вызове. Имя текстового параметра -произвольное имя, допустимое в Техтране, значением текстового параметра является произвольная последовательность символов, не содержащая символа #.

При вызове макроса вместо имени параметра в текст подставляется его значение.

Признаком текстовой подстановки является символ **&**, который должен предшествовать имени текстового параметра - каждый раз при его использовании в теле макроса.

Пример использования текстовых параметров:

ДЕТАЛЬ	1
МАКРО ПРИМЕР('ПАР1'='10')	2
КР&ПАР1=А, Б, РАД	3
КР&ПАР1&ПАР1=&ПАР1, &ПАР1, 2*&ПАР1	4
КОНМАК	5
.	
.	
ВЫЗОВ ПРИМЕР('ПАР1'='20', А=15, Б=-3, РАД=20)	6
ВЫЗОВ ПРИМЕР(А=15, Б=-3, РАД=20)	7
КОНЕЦ	8

В строках 3,4 текста макроса **ПРИМЕР** задаются текстовые подстановки. Текстовый параметр с именем **ПАР1** заменяется своим значением, заданным либо в вызове, либо в заголовке макроса.

Первому случаю соответствует строка 5 приводимого примера. При подстановке значения параметра '20', заданного в вызове, будут получены геометрические определения:

КР20=15,-3,20	3
КР2020=20,20,40	4

В связи с отсутствием значения параметра в вызове, находящемся в строке 6, подставляется значение параметра по умолчанию '10', заданное в заголовке макроса:

КР10=15,-3,20	3
----------------------	---

1.8.3 Заголовок макроса

Заголовок представляет собой специальный оператор **МАКРО** и имеет следующий формат:

```
МАКРО имя [(список значений)]
```

Параметры:

Параметр	Описание
<i>имя</i>	Имя макроса. Любое имя, разрешенное в Техтроне.
<i>список значений</i>	Значения текстовых параметров, установленных по умолчанию.

Если значения по умолчанию не заданы, скобки должны быть опущены. Значения по умолчанию используются в тех случаях, когда в операторе **ВЫЗОВ** значение данного текстового параметра не задано. Элементы списка значений отделяются друг от друга запятыми.

Значения по умолчанию могут быть заданы одним из следующих способов:

- литералом;
- текстовой подстановкой;
- объединением литералов и текстовых параметров.

Значения текстовых параметров макроса, заданные литералом задаются в виде:

```
'имя' = литерал
```

Параметр	Описание
<i>имя</i>	Имя определяемого текстового параметра.
<i>литерал</i>	Значение текстового параметра.

Способ аналогичен назначению текстового параметра в вызове.

Пример:

```
МАКРО ПРИМЕР( 'ИМЯ1'='КР10', 'ИМЯ2'='КР20' )
```

Параметры с именами **ИМЯ1** и **ИМЯ2** по умолчанию принимают соответственно значения: **КР10** и **КР20**.

Значения текстовых параметров, заданные текстовой подстановкой задаются в виде:

```
'имя' = &имя1
```

Параметр	Описание
<i>имя</i>	Имя определяемого текстового параметра.
<i>имя1</i>	Имя текстового параметра, определенного ранее.

Пример:

МАКРО ПРИМЕР('П1'='КР10', 'П2'='КР20', %
'ИМЯ1'=&П1, 'ИМЯ2'=&П2)

Параметры с именами **ИМЯ1** и **ИМЯ2** по умолчанию принимают те же значения: **КР10** и **КР20**.

Значения текстовых параметров, заданные объединением литералов и текстовых параметров задаются в виде:

$$имя' = \left\{ \begin{array}{l} \&имя_1 \\ литерал_1 \end{array} \right\} \left\{ \begin{array}{l} \&имя_2 \\ литерал_2 \end{array} \right\} \dots \left\{ \begin{array}{l} \&имя_n \\ литерал_n \end{array} \right\}$$

Параметр	Описание
<i>имя</i>	Имя определяемого текстового параметра.
<i>имя₁, ..., имя_n</i>	Имена текстовых параметров, определенных ранее.

Примеры задания текстовых параметров в вызове и по умолчанию:

МАКРО МАК1

Значений по умолчанию нет.

МАКРО М2 ('П1'='1', 'П2'='ПР'&П1, 'П3'='КР'&П1&П1)

В результате использования значений по умолчанию: П1=1, П2=ПР1, П3=КР11

МАКРО ПРИМ('П1'='1', 'П2'='ПР'&П4&П4, 'П3'='КР'&П4)

В результате использования значений по умолчанию: П1=1, П2=ПР

*значениеП4*значениеП4, П3=КР*значениеП4*

Значение параметра **П4** обязательно должно быть определено при вызове макроса или по умолчанию, в противном случае возникнет ошибка:

ВЫЗОВ ПРИМ('П4'='2')

или

МАКРО ПРИМ('П1'='1', 'П4'='2', 'П2'='ПР'&П4&П4, 'П3'='КР'&П4)

1.8.4 Вызов макроса

Для выполнения макроподстановки, т.е. для вставки в текст исходной программы на Техтране тела макроса и для замены формальных параметров фактическими, предназначен оператор **ВЫЗОВ**.

Формат:

$$\text{ВЫЗОВ} \left\{ \begin{array}{l} \text{имя} \\ \text{литерал} \end{array} \right\} [(\text{список параметров})]$$

Параметры:

Параметр	Описание
<i>имя</i>	Имя макроса, использованное в заголовке макроса - операторе МАКРО .
<i>литерал</i>	Спецификация файла, в котором находится текст макроса.
<i>список параметров</i>	Последовательность операторных параметров и значений текстовых параметров, разделенных запятыми.

 Если параметры в вызове отсутствуют, скобки должны быть опущены.

Имя макроса используется в том случае, когда макрос определяется в самой программе:

ВЫЗОВ МАКРО1

Спецификация файла макроса используется в том случае, когда макрос содержится в отдельном файле:

ВЫЗОВ 'D:\MAKROS\MAKRO1.МАК'

Умолчание для имени макроса действует при считывании макроса из внешнего файла. Это свойство позволяет не задавать в вызове макроса диск, папку и тип файла, содержащего макрос. Без назначения умолчаний макрос считывается из файла с указанным именем, находящегося в текущей папке. Умолчания для имени макроса задаются с помощью оператора двоеточие (:).

Формат:

$$: \text{литерал}_1, \text{литерал}_2$$

Параметры:

Параметр	Описание
<i>литерал₁</i>	Строка символов, присоединяемая к имени макроса слева.
<i>литерал₂</i>	Строка символов, присоединяемая к имени макроса справа.

Например, применение оператора **'D:\MAKROS'**, **'МАК'** позволяет задавать в программе вызовы макросов оператором, вида **ВЫЗОВ имя** вместо оператора **ВЫЗОВ 'D:\MAKROS\имя.МАК'**:

Приведенные ниже тексты дают одинаковый результат.

.....
ВЫЗОВ 'D:\MAKROS\ZUB.MAK'
ВЫЗОВ 'D:\MAKROS\ELLIPS.MAK'

.....
: 'D:\MAKROS\', '.MAK'

.....
ВЫЗОВ ZUB
ВЫЗОВ ELLIPS

Параметры вызова задаются в скобках после имени или спецификации файла вызываемого макроса и перечисляются через запятую. Если параметры в вызове отсутствуют, скобки должны быть опущены. Последовательность параметров, как правило, произвольна. Она имеет значение, если назначаемый параметр вызова использует в качестве операнда другой параметр вызова.

Используемый параметр должен быть назначен до его применения. Например:

ВЫЗОВ МАК(Б=5, А=12+Б)

Значение текстового параметра, заданное в вызове, отменяет значение по умолчанию. Например:

ДЕТАЛЬ
МАКРО М1 ('П1' = '1', 'П2' = & П1'0')

А = &П1 + &П2

ТЧ1 = А, В

КР1 = ТЧ1, В + &П1

ПЧТ1 А, В, ТЧ1, КР1

КОНМАК

ВЫЗОВ М1 ('П1' = '2', 'П2' = '25', В = 0) # ВЫЗОВ N1

ВЫЗОВ М1 ('П1' = '2', В = 3) # ВЫЗОВ N2

ВЫЗОВ М1 ('П2' = '35', В = 2.5) # ВЫЗОВ N3

ВЫЗОВ М1 (В = 7.4) # ВЫЗОВ N4

КОНЕЦ

Значения параметров и переменных, которые они приобретают при выполнении программы, приведены в таблице:

Номер вызова	П1	П2	В	А	ТЧ1		КР1	
					x y	x R	y	
N1	2	25	0	27	27	0	27	0 2
N2	2	20	3	22	22	3	22	3 5
N3	1	35	2.5	36	36	2.5	36	2.5 3.5

Номер вызова	П1	П2	В	А	ТЧ1		КР1	
					x y	x R	y	
N4	1	10	7.4	11	11 7.4	11	7.4 8.4	

См. также:

- [Заголовок макроса](#)^[242]

1.8.5 Вложенные вызовы

В теле макроса может находиться оператор **ВЫЗОВ** с именем другого макроса. В этом случае происходит вставка тела вызываемого макроса в тело вызывающего его макроса. Запрещена рекурсия - вызов макроса внутри самого себя.

В теле вызываемого макроса нельзя использовать текстовые параметры внешнего макроса, но их можно применять в операторе **ВЫЗОВ** для задания значений внутренних текстовых параметров, например:

МАКРО M1 ('П1'="", 'П2'="", 'П3'="")

ВЫЗОВ M2 ('П1'='&П1')

ВЫЗОВ M2 ('П1'='&П2+&П3')

КОНМАК

МАКРО M2

A=&П1-1

КОНМАК

ВЫЗОВ M1 ('П1'='10')

влож. вызовы M2 -N1,2

ВЫЗОВ M1 ('П2'='1', 'П3'='2')

влож. вызовы M2 -N3,4

Значения параметра **П1** и переменной **A**, приобретаемые ими при четырех последовательных вызовах макроса **M2**, приведены в таблице:

Номер вложенного вызова макроса M2	П1	A
N1	10	A=101
N2	+	A=+ 1
N3		A= 1
N4	1+2	A=1+21

При выполнении 2-го вызова макроса **M2** формируется синтаксически неправильный оператор **A=+1** и системой будет сформировано сообщение об ошибке компиляции.

1.8.6 Локализация переменных в макросе

Для обеспечения полной независимости макросов от вызывающей программы предусмотрена возможность локализации переменных внутри макросов.

Переменная является **локализованной** в макросе, если ее значение доступно только внутри этого макроса. В этом случае вне макроса может существовать другая переменная с тем же именем, но с другими атрибутами -типом и значением.

Для локализации переменной внутри макроса, необходимо явно описать ее в теле макроса - указать тип и значение. В противном случае в макросе будет использоваться переменная, определенная вне макроса, либо произойдет ошибка интерпретации, если эта переменная не определена до вызова макроса.

Таким образом, если переменная используется впервые до вызова макроса, то она сохраняет свой тип и значение при входе в макрос. При выходе из макроса она имеет значение, приобретенное при выполнении тела макроса. Если переменная используется впервые в теле макроса, то она считается локализованной на текущем уровне вложения макросов и при выходе из него оказывается неопределенной. Для вложенного макроса тип и значение такой переменной сохраняются при входе во вложенный макрос.

Далее приведены примеры локализации переменных.

Пример 1:

```
A=1
МАКРО М
  ВЕЩ А
  A=2
КОНМАК
ВЫЗОВ М
ПЧТ1 А
```

Поскольку в макросе **М** имеется явное задание типа переменной **А**, то она считается локальной в макросе **М**. Вне макроса ее значение, равное **2**, недоступно, т.к. вне тела макроса определена другая переменная **А**, которой было присвоено значение, равное **1**. Именно это значение будет выведено на печать после обработки макроса.

Пример 2:

```
A=1
МАКРО М
  A=2
КОНМАК
ВЫЗОВ М
ПЧТ1 А
```

В этом случае печатается значение **А=2**, т.к. в теле макроса используется переменная, определенная вне его. Расположение тела макроса относительно оператора **А=1** здесь не существенно.

Пример 3:

```
МАКРО М
  A=2
КОНМАК
ВЫЗОВ М
```

ПЧТ1 А

В данном примере допущена ошибка так как значение переменной **А** вне макроса не определено.

Пример 4:

```

А=1
МАКРО М
  ПРЯМАЯ А=ХПАР,12
  ПЧТ1 А
КОНМАК
ВЫЗОВ М
ПЧТ1 А

```

В макросе **М** выполнено явное задание типа и значения локальной переменной **А**, как прямой. По оператору **ПЧТ1**, находящемуся в макросе, будут распечатаны параметры этой прямой. Вне тела макроса определена другая переменная **А** (вещественная), которой присвоено значение, равное **1**. Именно это значение будет выведено на печать после отработки макроса.

Операторные параметры, заданные в вызове, рассматриваются как операторы, находящиеся внутри макроса. Переменные, определяемые в операторных параметрах, должны удовлетворять тем же соглашениям, что и переменные тела макроса. Они обрабатываются в порядке следования в вызове, поэтому их последовательность имеет значение, например:

<i>неверно:</i>	<i>верно:</i>
А=1	А=1
ВЫЗОВ М(ВЕЩ А), Б=А+15)	ВЫЗОВ М(Б=А+15)

Первый операторный параметр в вызове макроса объявляет переменную **А** как локальную в макросе. Поскольку ее значение не задано до второго операторного параметра, в котором она используется, последует сообщение об ошибке:

“Значение переменной не определено”

1.8.7 Использование меток в макросах

Все используемые в макросе метки локализованы внутри этого макроса. Это означает, что передачи управления в макрос и из него запрещены. Разрешены только переходы внутри тела макроса.

1.9 Управление выводом результатов

Процессор в ходе выполнения программы формирует протокол работы, содержащий сообщения об ошибках и отладочные данные. Эти данные можно просмотреть в окне *Протокол выполнения*. Для управления объемом выводимой информации в Текстрене используются операторы **ПЕЧТР**, **СЛЕД** и **ПЧТ1**.

Темы этого раздела:

- [Установка режима вывода траектории движения инструмента](#)^[251]
- [Трассировка программы](#)^[249]

- [Вывод значений переменных или параметров геометрических объектов](#)^[252]

1.9.1 Трассировка программы

Для **трассировки программы**, т.е. вывода результатов выполнения операторов, используется оператор **СЛЕД**. Этот оператор выводит значения переменных и выражений, результаты логических проверок и геометрических построений, позволяет контролировать выполнение циклов и вызов макросов. Он позволяет вывести результат выполнения практически любого исполняемого оператора.

Формат:

$\text{СЛЕД} \left\{ \begin{array}{l} \text{ВКЛ} \\ \text{ВСЕ} \\ \text{ВЫКЛ} \end{array} \right\} \left\{ \begin{array}{l} 1 \\ [, 0] \end{array} \right\}$

Параметры:

Параметр	Описание
ВКЛ	Включение вывода результатов выполнения операторов, кроме составляющих текстовую вставку или тело макроса в протокол выполнения.
ВСЕ	Включение вывода результатов выполнения всех операторов, в том числе операторов текстовых вставок и макросов в протокол выполнения.
ВЫКЛ	Выключение вывода результатов выполнения операторов в протокол выполнения (выключение режима трассировки).
0 или 1	Уровни вывода данных в протокол выполнения.

Перечень исполняемых операторов и результаты их выполнения, выводимые в протокол, приведены в следующей таблице:

Содержание строки	Данные, выводимые в протокол
НАМЕТКУ	ПЕРЕХОД НА МЕТКУ.
ЕСЛИ (логическое выражение)	ЛОЖЬ или ИСТИНА .
ПОВТОР <i>счетчик</i>	При входе в цикл: управляющая переменная=начальное значение ЦИКЛ ДО =конечное значение ШАГ =приращение управляющей переменной

Содержание строки	Данные, выводимые в протокол
	При исполнении цикла: ШАГ ЦИКЛА ПО ПЕРЕМЕННОЙ.... ЛОЖЬ При завершении цикла: ШАГ ЦИКЛА ПО ПЕРЕМЕННОЙ.... ИСТИНА ВЫХОД ИЗ ЦИКЛА ПО ПЕРЕМЕННОЙ.... .
КОНЦИКЛ	управляющая переменная=текущее значение+шаг.
ПОВТОР ПОКА	При входе в цикл: НАЧАЛО УСЛОВНОГО ОПЕРАТОРА ИСТИНА или ЛОЖЬ При исполнении цикла: ИСТИНА или ЛОЖЬ При завершении цикла: ИСТИНА или ЛОЖЬ КОНЕЦ УСЛОВНОГО ОПЕРАТОРА.
ВЫЗОВ	ВХОД В МАКРО ВЫХОД ИЗ МАКРО.
<i>Арифметическое выражение</i>	Результат.
<i>Геометрическое определение</i>	Имя и параметры геометрического объекта.
КОНТУР	Параметры сегментов контура; координаты опорных точек сплайна.

Трассировка программы может быть выполнена на уровне программы или на уровне макроса. В первом случае (уровень вывода 0), в протокол будут включены результаты выполнения только управляющих операторов **ЕСЛИ**, **НАМЕТКУ**, **ПОВТОР**, **КОНЦИКЛ**, **ВЫЗОВ**, **КОНЕЦ МАКРОСА.**, находящихся вне макроса. Во втором случае (уровень вывода 1) – результаты выполнения всех операторов **ЕСЛИ**, **НАМЕТКУ**, **ПОВТОР**, **КОНЦИКЛ**, **ВЫЗОВ**, **КОНЕЦ МАКРОСА**, **КОНТУР**, арифметических выражений, геометрических определений. Перед результатом выводится номер соответствующей строки.

См. также:

- Протокол выполнения

- Настройка протокола процессора

1.9.2 Установка режима вывода траектории движения инструмента

Для установки режима вывода траектории движения инструмента используется оператор **ПЕЧТР**.

Формат:

ПЕЧТР	$\left. \begin{array}{l} \text{ВКЛ} \\ \text{ВСЕ} \\ \text{ВЫКЛ} \end{array} \right\}$
--------------	----------------------------------------------------------------------------------------

Параметры:

Параметр	Описание
ВКЛ	Включение вывода результатов обработки в протокол выполнения.
ВСЕ	Включение вывода всех результатов обработки в протокол выполнения.
ВЫКЛ	Выключение вывода результатов обработки в протокол выполнения.

Если включен вывод траектории, т.е. использован оператор **ПЕЧТР ВКЛ**, то в протокол выводятся следующие параметры выполняемого движения:

- данные об инструменте;
- метод интерполяции и точность линейной аппроксимации;
- параметры геометрических объектов, образующих контур;
- координаты начальной точки;
- координаты точек пересечения эквидистант;
- координаты опорных точек.

При наличии в программе таблично заданных кривых и контуров формируются сообщения "НАЧАЛО КОНТУРА", "КОНЕЦ КОНТУРА", между которыми выводятся параметры сегментов контура и координаты точек их сопряжения.

Если задан режим **ВСЕ**, т.е. использован оператор **ПЕЧТР ВСЕ**, то в протокол дополнительно выводятся следующие параметры:

- координаты точек разбиения окружностей на квадранты;
- промежуточные точки аппроксимации окружности отрезками прямых;
- сообщение "ДВИЖЕНИЕ ПО КОНТУРУ";
- траектория движения по контуру или по таблично заданной кривой.

Использование в программе оператора **ПЕЧТР ВЫКЛ** или отсутствие оператора **ПЕЧТР** приводит к подавлению вывода траектории до конца программы или до следующего

оператора ПЕЧТР ВКЛ (ПЕЧТР ВСЕ). Исключения составляют строки, содержащие ошибки движения, которые выводятся всегда.

Данные обо всех геометрических элементах выводятся в соответствии с их внутренним представлением.

См. также :

- [Внутреннее представление геометрических объектов](#)^[33]

1.9.3 Вывод значений переменных или параметров геометрических объектов

Для вывода в протокол значения некоторой переменной или параметров геометрического объекта (точки, окружности, вектора или прямой) используется оператор ПЧТ1.

Формат :

```
ПЧТ1 имя1, имя2, имя3, ...
```

Параметры:

Параметр	Описание
имя1, имя2, имя3,	Имена геометрических объектов или переменных.

Результат применения оператора ПЧТ1 аналогичен использованию оператора СЛЕД: результаты арифметических выражений и геометрических построений выводятся в протокол выполнения системы в следующих форматах.

Арифметически выражения:

```
номер строки имя переменной = значение
```

Геометрические определения:

```
номер строки имя объекта = внутреннее представление
```

1.10 Операторы постпроцессора

Операторы постпроцессора выполняют следующие функции:

- задают режимы обработки деталей на станке с ЧПУ: подачу, частоту вращения шпинделя, коррекцию, включение охлаждения и т.д.;
- управляют формированием технологических команд: сменой инструмента, поворотом стола, зажимом пиноли или направляющих стола и т.д.;
- управляют формированием УП и программносителя: выводом заголовков программы и подпрограмм, нумерацией кадров, пометкой главных кадров, выводом комментариев и т.д.

Формат:

$$\text{главное слово} \left\{ \begin{array}{l} \text{вспомогательное слово} \\ \text{литерал} \\ \text{число} \end{array} \right\}, \dots$$

Параметры:

Параметр	Описание
<i>главное слово</i>	Оператор постпроцессора.==
<i>вспомогательное слово</i>	Служебное слово.
<i>число</i>	Число или арифметическое выражение.
<i>литерал</i>	Литерал.

Технологические команды, задаваемые операторами постпроцессора в описанном выше формате, выполняются в текущей точке траектории. Имеется возможность задавать произвольную точку привязки технологических команд.

Формат:

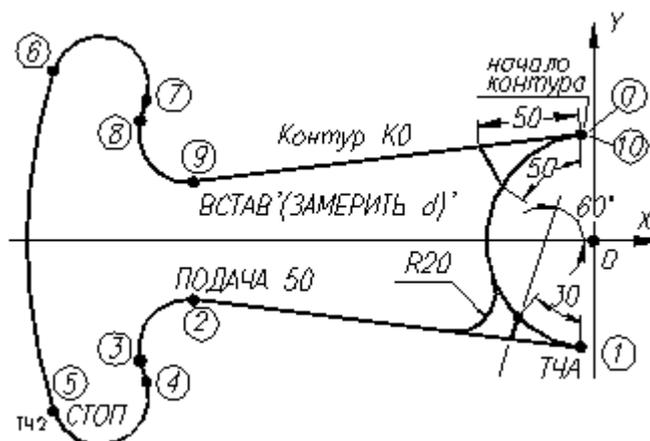
$$\text{контур}, \left\{ \begin{array}{l} \text{номер} \\ \text{точка} \end{array} \right\} : \text{оператор}$$

Параметры:

Параметр	Описание
<i>контур</i>	Имя контура, в точке которого должна быть выполнена технологическая команда.
<i>номер</i>	Номер точки контура, куда вводится технологическая команда.
<i>точка</i>	Имя точки, где вводится технологическая команда.
<i>оператор</i>	Оператор постпроцессора.

Пример:

Пример иллюстрирует два способа задания технологических команд, в текущей точке траектории и в произвольной точке, для траектории обхода контура, изображенного на следующем рисунке.



В первом случае, операторы постпроцессора задаются при описании траектории движения инструмента одновременно с операторами движения.

ХОД(1) = ИЗ 0,0
ИНСТЛВ; НАПРТ ТЧ0
ИДИ ДО КР1
ВЛЕВО КР1
НАЗАД ПР1 КАС КР2
ПОДАЧА 50
ВПЕРЕД КР2 : ПР2 : КР3 КАС КР4
СТОП
ВПЕРЕД КР4 : КР5 : ПР3 : КР6 КАС ПР4
ВСТАВ '(ЗАМЕРИТЬ d)'
ВПЕРЕД ПР4 ЗА КР1

Во втором случае, сначала задается траектория без назначения технологии, а затем в требуемых точках задаются технологические команды.

ХОД(2) = ИЗ 0,0
НАПРТ ТЧ0
ИДИ ДО КР1
ВЛЕВО КР1
НАЗАД ПР1 КАС КР2
ВПЕРЕД КР2 : ПР2 : КР3 : КР4 : КР5 : ПР3 : КР6 : ПР4 ЗА КР1

ХОД(2),2: ПОДАЧА 50
ХОД(2),ТЧ2: СТОП
ХОД(3),9: ВСТАВ '(ЗАМЕРИТЬ d)'

Темы этого раздела:

- [Начало и конец программы](#)^[255]
- [Вызов постпроцессора](#)^[255]
- [Управление программноносителем](#)^[256]
- [Управление сменой инструмента](#)^[259]
- [Ввод и отмена коррекции](#)^[261]
- [Управление подачей](#)^[264]

- [Управление шпинделем](#)^[266]
- [Управление охлаждением](#)^[268]
- [Стандартные циклы](#)^[269]
- [Останов станка](#)^[275]
- [Подготовительные и вспомогательные функции](#)^[276]
- [Управление поворотным столом](#)^[277]
- [Вставка текста](#)^[278]
- [Система координат станка](#)^[279]
- [Управление резкой](#)^[282]
- [Специальные функции постпроцессора](#)^[284]
- [Пример использования операторов постпроцессора](#)^[285]

1.10.1 Начало и конец программы

Начало программы

Программа на Техтране начинается оператором **ДЕТАЛЬ**.

Формат:

```
ДЕТАЛЬ 'имя детали'
```

Конец программы

Программа заканчивается оператором **КОНЕЦ**. По нему формируются завершающие кадры УП.

Формат:

```
КОНЕЦ
```

1.10.2 Вызов постпроцессора

Вызов постпроцессора осуществляется оператором **СТАНОК**.

Формат:

```
СТАНОК 'имя', a [ , p1 [ , p2 ] ]
```

Параметры:

Параметр	Описание
<i>имя</i>	Наименование оборудования с ЧПУ. Имеет характер

Параметр	Описание
	комментария и выводится в заголовке УП.
<i>a</i>	Номер паспорта станка.
<i>P₁ P₂</i>	Параметры вызова постпроцессора.

Оператор **СТАНОК** должен стоять за оператором **ДЕТАЛЬ**. Он настраивает постпроцессор на конкретную комбинацию станок / устройство ЧПУ. По нему в УП выводятся начальные кадры.

Назначение параметров оператора **СТАНОК** определяется для конкретного постпроцессора индивидуально. Так например, постпроцессор для системы ЧПУ H22, обрабатывая оператор **СТАНОК '16K20Ф3', 109,1** формирует по его параметру (p1) команду **M041** -команду выбора первого диапазона оборотов шпинделя. Оператором **СТАНОК 'ЛФ260МФ3', 740,3** вызывается постпроцессор для системы ЧПУ 2С85; параметр оператора (p1) используется в качестве номера программы -в результате формируется заголовок программы: **%3**.

1.10.3 Управление программносителем

Операторы этой группы позволяют управлять формированием управляющей программы и дают возможность изменять нумерацию кадров, отмечать кадры для условного пропуска, разбивать программу на части, управлять перемоткой носителя и др.

Темы этого раздела:

- [Разбиение программы](#)^[256]
- [Изменение нумерации кадров](#)^[257]
- [Пропуск кадров](#)^[258]
- [Задание длины лидера](#)^[258]
- [Перемотка ленты](#)^[258]
- [Создание маркера в программе](#)^[258]

1.10.3.1 Разбиение программы

Операторы этой группы предназначены для разбиения управляющей программы, если изготовить единый программноноситель по каким-либо причинам невозможно (например, из-за ограничения по длине). Имеется два оператора разбиения программы: **РАЗБПР** и **АБЗАЦ**.

Выделение частей программы

Для выделения частей программы без разделения носителя предназначен оператор **РАЗБПР**.

Формат:

РАЗЪПР

При необходимости постпроцессор для каждой отдельной части УП вставляет комментарии и ограничители.

Завершение замкнутого цикла обработки

Для завершения замкнутого цикла обработки предназначен оператор **АБЗАЦ**.

Формат:

АБЗАЦ

Данный оператор обеспечивает формирование нового программносителя (вместе с новым заголовком).

1.10.3.2 Изменение порядка нумерации кадров

Для изменения нумерации кадров предназначен оператор **ПОРНОМ**. Он позволяет изменять порядковый номер следующего кадра и шаг нумерации.

Формат:

<p>ПОРНОМ { <i>номер</i> ПРИРАЦ, <i>шаг</i> <i>номер</i>, ПРИРАЦ, <i>шаг</i> }</p>

Параметры:

Параметр	Описание
<i>номер</i>	Номер следующего кадра.
ПРИРАЦ , <i>шаг</i>	Шаг нумерации.

По умолчанию кадры УП нумеруются с единицы, шаг нумерации равен единице. Если оператор задан с единственным параметром *номер*, то следующему кадру УП присваивается указанный номер, а все последующие кадры нумеруются с шагом, равным 1. Если в операторе указан дополнительно параметр *шаг*, то последующие кадры нумеруются с указанным шагом. Отсутствие параметра *номер* (второй вариант) означает изменение шага нумерации без изменения порядка нумерации.

Оператор **ПОРНОМ** являются модальными, т.е. каждое условие остается в силе, пока не будет отменено другим оператором **ПОРНОМ**.

1.10.3.3 Пропуск кадров

Для пометки кадров УП символом пропуска кадра предназначен оператор **ПРОПБЛ**.

Формат:

ПРОПБЛ {	ВКЛ	}
	ВЫКЛ	

Параметры:

Параметр	Описание
ВКЛ	Включение режима пометки кадров символом пропуска. В начале каждого кадра выводится символ (/).
ВЫКЛ	Выключение режима пометки кадров символом пропуска.

1.10.3.4 Задание длины лидера

Для задания длину заправочной части перфоленты предназначен оператор **ЛИДЕР**.

Формат:

ЛИДЕР <i>длина</i>

Параметры:

Параметр	Описание
<i>длина</i>	Величина заправочной части перфоленты в см.

1.10.3.5 Перемотка ленты

Для перемотки программносителя предназначен оператор **ПЕРЕМТК**.

Формат:

ПЕРЕМТК

1.10.3.6 Создание маркера в программе

Для создания маркера в программе предназначен оператор **МАРКЕР**. Он позволяет делать пометки на программносителе, которые используются либо для возврата к ним по оператору **ПЕРЕМТК**, либо для поиска определенного участка ленты.

Формат:

МАРКЕР $\left[\begin{array}{l} \{ \text{ВКЛ} \} \\ \{ \text{ВЫКЛ} \} \\ \text{число} \end{array} \right]$

Параметры:

Параметр	Описание
ВКЛ	Включение маркировки участка ленты.
ВЫКЛ	Выключение маркировки участка ленты.
<i>число</i>	Идентификатор метки на ленте.

Оператор **МАРКЕР** без параметров используется для маркировки главного кадра символом “:”.

1.10.4 Управление сменой инструмента

Эта группа операторов предназначена для поиска и загрузки инструмента. Предварительно инструмент должен быть описан оператором **ИНСТР** (см. [Описание инструмента](#)^[154]). Если станок имеет магазин инструментов, то предварительный поиск инструмента задается оператором **ВЫБОРИН**. Загрузка найденного инструмента (или поворот револьверной головки) осуществляется оператором **ЗАГРУЗ**.

Темы этого раздела:

- [Выбор инструмента](#)^[259]
- [Смена инструмента](#)^[260]
- [Разгрузка инструмента](#)^[260]
- [Возврат инструмента](#)^[261]

1.10.4.1 Выбор инструмента

Предварительный поиск инструмента для последующей загрузки осуществляется оператором **ВЫБОРИН**. Он позволяет совместить поиск инструмента в магазине с обработкой.

Формат:

ВЫБОРИН <i>имя</i> [,РУЧН][,НОМКОР...]

Параметры:

Параметр	Описание
----------	----------

<i>имя</i>	Имя выбираемого инструмента.
РУЧН	Признак выбора инструмента вручную, т.е. уведомление постпроцессора о том, что инструмент уже подготовлен к загрузке и в УП ничего выводить не нужно.
НОМКОР	Описание корректоров, связанных с инструментом.

В управляющей программе оператору **ВЫБОРИН**, как правило, соответствует адрес **T**.

См. также:

- [Ввод и отмена коррекции](#)^[261]

1.10.4.2 Смена инструмента

Смена инструмента задается оператором **ЗАГРУЗ**.

Формат:

ЗАГРУЗ *имя* [, **РУЧН**][, **НОМКОР**...]

Параметры:

Параметр	Описание
<i>имя</i>	Имя загружаемого инструмента.
РУЧН	Признак загрузки инструмента вручную, т.е. уведомление постпроцессора о том, что инструмент уже загружен и в УП ничего выводить не нужно.
НОМКОР	Описание корректоров, связанных с инструментом.

В управляющей программе оператору **ЗАГРУЗ**, как правило, соответствует команда **M06**.

См. также:

- [Ввод и отмена коррекции](#)^[261]

1.10.4.3 Разгрузка инструмента

Возврат последнего загруженного инструмента в магазин задается оператором **РАЗГРУЗ**.

Формат:

РАЗГРУЗ

1.10.4.4 Возврат инструмента

Для возврата инструмента в точку смены используется оператор **ВОЗВРАТ**.

Формат:

ВОЗВРАТ

Оператор постпроцессора **ВОЗВРАТ** применяется только в том случае, когда станок имеет специальную команду перемещения в точку смены инструмента или фиксированную позицию для смены инструмента.

Оператор постпроцессора **ВОЗВРАТ,ТЕКУЩ** применяется, если инструмент уже загружен, и следующий переход выполняется без смены инструмента. При необходимости инструмент может быть отведен в безопасную позицию без разгрузки

1.10.5 Ввод и отмена коррекции

В Техтроне с каждым инструментом могут быть связаны определенные корректоры. Ввод коррекции, в зависимости от возможностей оборудования, осуществляется либо по команде смены инструмента, либо с помощью оператор **КОРРЕК**. Этот оператор позволяет задавать коррекцию на длину инструмента, на радиус, на отдельные линейные перемещения. Для многоинструментальных станков оператор **КОРРЕК** может быть задан только после загрузки инструмента, т.е. после операторов **ВЫБОРИН** и **ЗАГРУЗ**. При этом в операторе **КОРРЕК** можно не указывать тип коррекции и номер корректора, если эти данные были заданы в операторах управления инструментом.

Темы этого раздела:

- [Коррекция на радиус](#)^[261]
- [Коррекция на длину](#)^[262]
- [Коррекция по осям](#)^[262]
- [Парная коррекция](#)^[263]
- [Отмена коррекции](#)^[263]

1.10.5.1 Коррекция на радиус

Формат:

КОРРЕК { **СПРАВА**
СЛЕВА }, **РАДИУС**, *номер*

Параметры:

Параметр	Описание
СПРАВА	Инструмент справа от обрабатываемого контура.
СЛЕВА	Инструмент слева от обрабатываемого контура.

Параметр	Описание
<i>номер</i>	Номер корректора .

1.10.5.2 Коррекция на длину

Формат:

КОРРЕК ВКЛ, ДЛИНА, *номер*

Параметры:

Параметр	Описание
<i>номер</i>	Номер корректора, вводимого на длину инструмента.

1.10.5.3 Коррекция по осям

Формат:

КОРРЕК ВКЛ[, ХКООРД, *номер*₁][[, УКООРД, *номер*₂][[, ЗКООРД, *номер*₃]

Параметры:

Параметр	Описание
ХКООРД	Коррекция по координате X .
<i>номер</i> ₁	Номер корректора по координате X .
УКООРД	Коррекция по координате Y .
<i>номер</i> ₂	Номер корректора по координате Y .
ЗКООРД	Коррекция по координате Z .
<i>номер</i> ₃	Номер корректора по координате Z .

1.10.5.4 Парная коррекция

Формат:

$$\text{КОРРЕК ВКЛ} \left\{ \begin{array}{l} \text{ХУПЛ} \\ \text{УЗПЛ} \\ \text{ЗХПЛ} \end{array} \right\}, \text{номер}$$

Параметры:

Параметр	Описание
ХУПЛ	Коррекция в плоскости XY .
УЗПЛ	Коррекция в плоскости YZ .
ЗХПЛ	Коррекция в плоскости ZX .
<i>номер</i>	Номер корректора.

1.10.5.5 Отмена коррекции

Формат:

$$\text{КОРРЕК} \left\{ \begin{array}{l} \text{ОТМЕН} \\ \text{ВЫКЛ} \end{array} \right\} \left[\begin{array}{l} \text{ХКООРД} \\ \text{УКООРД} \\ \text{ЗКООРД} \\ \text{ХУПЛ} \\ \text{УЗПЛ} \\ \text{ЗХПЛ} \\ \text{ДЛИНА} \\ \text{РАДИУС} \end{array} \right]$$

Параметры:

Параметр	Описание
ХКООРД	Отмена коррекции по координате X .
УКООРД	Отмена коррекции по координате Y .
ЗКООРД	Отмена коррекции по координате Z .
ХУПЛ	Отмена парной коррекции в плоскости XY .

Параметр	Описание
УЗПЛ	Отмена парной коррекции в плоскости YZ .
ЗХПЛ	Отмена парной коррекции в плоскости ZX .
ДЛИНА	Отмена коррекции на длину.
РАДИУС	Отмена коррекции на радиус

 Если в операторе отмены коррекции задано только слово **ВЫКЛ**, то отменяются все введенные ранее корректоры.

1.10.6 Управление подачей

Для управления подачей используются два оператора: **БЫСТРО** и **ПОДАЧА**. Обработка этих операторов зависит от особенностей станка и системы ЧПУ, т.е. определяется паспортными данными станка.

Темы этого раздела:

- [Ускоренная подача](#)^[264]
- [Рабочая подача](#)^[265]

1.10.6.1 Ускоренная подача

Для задания ускоренных перемещений инструмента предназначен оператор **БЫСТРО**.

Формат:

БЫСТРО

Задание данного оператора означает, что все последующие движения производятся ускоренно до появления оператора **ПОДАЧА**. Для последующих быстрых перемещений после окончания действия оператора **ПОДАЧА** необходимо повторить оператор **БЫСТРО**. Обычно этот оператор применяют для подвода/отвода режущего инструмента, подхода к обрабатываемой поверхности, выхода в начальную точку.

Если система ЧПУ имеет специальную команду быстрой подачи, то постпроцессор проверяет, является ли заданное направление движения допустимым, и в кадр УП заносится код быстрой подачи. В противном случае выбирается максимально допустимое для заданного направления значение рабочей подачи.

Если система ЧПУ не имеет специальной команды быстрой подачи, то значением подачи становится значение максимальной для данного направления движения подачи из паспорта станка.

Если перемещение в заданном направлении на быстром ходу запрещено, то после отработки текущего кадра на выбранной из паспорта подаче, быстрая подача будет восстановлена.

Оператор **БЫСТРО** нельзя задавать при движении по окружности. В этом случае будет

выдано сообщение об ошибке, а в качестве значения рабочей подачи будет выбрано прежнее значение подачи или минимальная подача из паспорта (если рабочая подача не определена). При последующих линейных перемещениях быстрая подача будет восстановлена.

1.10.6.2 Рабочая подача

Для задания рабочей подачи предназначен оператор **ПОДАЧА**.

Формат:

$$\text{ПОДАЧА} \left\{ \begin{array}{l} [\text{ММИН}] \\ \text{ММОБ} \end{array} \right\}, \text{подача}$$

Параметры:

Параметр	Описание
ММИН	Подача в мм/мин.
ММОБ	Подача в мм/об.
<i>число</i>	Величина рабочей подачи.

Запрограммированное значение подачи сохраняется до появления оператора **БЫСТРО** или нового оператора **ПОДАЧА**.

Оператор обрабатывается так: значение подачи проверяется на допустимость. Если подача меньше минимальной, то выбирается паспортное значение минимальной подачи. Если подача слишком велика, то выбирается максимально допустимое для данного направления значение подачи.

В Техтроне реализованы два метода проверки подачи, которые задаются в паспорте (параметры **КлючПодач** и **МаксПодачи**).

Первый метод проверки подач используется для станков, которые имеют одинаковые приводы по осям X,Y,Z.. В этом случае в паспорте должны быть заданы максимальные значения подач при движении по одной, двум и трем координатам. Постпроцессор определяет, по какому числу координат задано движение, сравнивает запрограммированное значение подачи с соответствующим значением из паспорта и, при необходимости, выбирает значение из паспорта.

Второй метод целесообразно использовать для станков, приводы которых имеют разные характеристики. В этом случае в паспорте задаются максимальные значения подач для каждой оси. Постпроцессор вычисляет проекции вектора подачи на оси координат и сравнивает их с паспортными значениями. Если одна из компонент вектора подачи превышает допустимое значение, то выбирается соответствующее значение из паспорта, и подача корректируется.

1.10.7 Управление шпинделем

Для управления шпинделем предназначены операторы **ШПИИДЛ** и **ПОВШП**. Они позволяют задавать частоту вращения шпинделя, скорость резания, угол поворота и специальные функции.

Темы этого раздела:

- [Включение и выключение шпинделя](#)^[266]
- [Скорость вращения шпинделя](#)^[266]
- [Вспомогательные функции](#)^[267]
- [Поворот шпинделя на фиксированный угол](#)^[268]

1.10.7.1 Включение и выключение шпинделя

Включение шпинделя без задания частоты вращения и выключение шпинделя производится оператором **ШПИИДЛ** в формате:

ШПИИДЛ { ВКЛ ВЫКЛ }

Параметры:

Параметр	Описание
ВКЛ	Включение шпинделя.
ВЫКЛ	Выключение шпинделя.

Оператор **ШПИИДЛ ВКЛ** может использоваться для восстановления ранее заданных параметров после останова станка.

1.10.7.2 Скорость шпинделя

Вариант оператора **ШПИИДЛ**, описанный в данном разделе, предназначен для задания скорости вращения шпинделя, направления вращения и диапазона оборотов. Скорость может быть задана либо частотой вращения, либо скоростью резания. Второй вариант допустим только для систем ЧПУ, допускающих программирование скорости резания. В этом случае может быть задана для контроля максимальная частота вращения.

Формат :

ШПИИДЛ [ОБМИН ПОВММ] <i>обороты</i> [ПОЧС ПРЧС] [, ДИАП , <i>номер</i>] [, МАКСОБМ , <i>кредел</i>]

Параметры:

Параметр	Описание
ОБМИН	Частота вращения шпинделя в об/мин.
ПОВММ	Скорость резания в м/мин.
<i>обороты</i>	Скорость вращения шпинделя.
ПОЧС	Вращение шпинделя по часовой стрелке (правое).
ПРЧС	Вращения шпинделя против часовой стрелки (левое).
<i>номер</i>	Номер диапазона оборотов шпинделя.
<i>предел</i>	Максимально допустимое число оборотов шпинделя.

1.10.7.3 Вспомогательные функции

Задание вспомогательных функций управления шпинделем производится оператором **ШПИНДЛ** в формате:

ШПИНДЛ	}	ФИКС
		НЕЙТРЛ
		ОРИЕНТ
		ПОЧС
		ПРЧС

Параметры:

Параметр	Описание
ФИКС	Зажим шпинделя.
НЕЙТРЛ	Отключение привода поворота шпинделя.
ОРИЕНТ	Останов шпинделя в ориентированном положении и его зажим.
ПОЧС	Включение вращения шпинделя по часовой стрелке.
ПРЧС	Включение вращения шпинделя против часовой стрелки.

 Функции, задающие направление вращения, позволяют изменять направление без задания частоты вращения. Это обычно используется, если станок не имеет программного управления частотой вращения шпинделя.

1.10.7.4 Поворот шпинделя на фиксированный угол

Поворот шпиндельной головки в определенную позицию или на заданный угол производится оператором **ПОВШП**.

Формат:

$$\text{ПОВШП} \left\{ \begin{array}{l} \text{УГОЛ} \\ \text{ПРИРАЩ} \end{array} \right\}, \text{угол} \left\{ \begin{array}{l} [\text{ПОЧС}] \\ \text{ПРЧС} \end{array} \right\} [, \text{ФИКС}]$$

Параметры:

Параметр	Описание
УГОЛ	Задание величины угла абсолютным значением.
ПРИРАЩ	Задание величины угла приращением относительно предыдущего положения.
<i>угол</i>	Угол поворота шпинделя в градусах.
ПОЧС	Поворот шпинделя по часовой стрелке.
ПРЧС	Поворот шпинделя против часовой стрелки.
ФИКС	Фиксация шпиндельной головки. =

1.10.8 Управление охлаждением

Для управления включением и типом охлаждения предназначен оператор **ОХЛАД**.

Формат:

$$\text{ОХЛАД} \left\{ \begin{array}{l} \text{ВКЛ} \\ \text{ВЫКЛ} \\ \text{ЖИДК} \\ \text{ТУМАН} \\ \text{ВОЗДУХ} \\ \text{ВНУТРИ} \\ \text{ВЫСОК} \end{array} \right\}$$

Формат оператора, задающий номер трубопровода охлаждения:

$$\text{ОХЛАД номер}$$

Параметры:

Параметр	Описание
ВКЛ	Включение охлаждения.
ВЫКЛ	Выключение охлаждения.
ЖИДК	Включение охлаждения жидкостью.
ТУМАН	Включение охлаждения туманом.
ВОЗДУХ	Включение охлаждения воздухом
ВНУТРИ	Включение внутреннего охлаждения инструмента
ВЫСОК	Включение охлаждения высоким давлением
<i>номер</i>	Номер трубопровода, подающего охлаждающий компонент.

1.10.9 Стандартные циклы

Для задания параметров стандартных циклов сверлильно-расточной обработки предназначен оператор **ЦИКЛ**. Стандартные циклы выполняются в каждой точке траектории, заданной командами поточечного движения, до выключения цикла.

Темы этого раздела:

- [Задание безопасного расстояния](#)^[269]
- [Сверление](#)^[270]
- [Глубокое сверление](#)^[270]
- [Цекование и зенкование](#)^[271]
- [Коническое зенкование](#)^[271]
- [Нарезание резьбы](#)^[272]
- [Растачивание](#)^[273]
- [Развертывание](#)^[273]
- [Зенкерование](#)^[274]
- [Включение и выключение цикла](#)^[274]

1.10.9.1 Задание безопасного расстояния

Безопасное расстояние, т.е. высота над поверхностью детали, на которой происходит быстрое позиционирование при перемещении из одной точки в другую и на которую выводится инструмент по окончании каждого цикла, должна задаваться оператором **БЕЗОПРСТ**.

Формат:

БЕЗОПРСТ <i>расстояние</i>

Параметры:

Параметр	Описание
<i>расстояние</i>	Величина безопасного расстояния от поверхности детали в мм.

1.10.9.2 Сверление

Цикл сверления (G81) задается оператором **ЦИКЛ** следующего формата:

ЦИКЛ СVERЛ , <i>глубина</i> , { ММИН }, { ММОБ }, <i>подача</i> , <i>недоход</i> [, ВЫСТОЙ , <i>пауза</i>]

Параметры:

Параметр	Описание
<i>глубина</i>	Общая глубина обработки.
ММИН	Подача в мм/мин.
ММОБ	Подача в мм/об.
<i>подача</i>	Рабочая подача в цикле.
<i>недоход</i>	Недоход до поверхности детали.
<i>пауза</i>	Выдержки времени в цикле в сек.

1.10.9.3 Глубокое сверление

Цикл глубокого сверления задается оператором **ЦИКЛ** следующего формата:

ЦИКЛ { ГЛУБОК }, { ГЛУБОК1 }, <i>глубина</i> [, ПРИРАЩ , <i>шаг</i>], { ММИН }, { ММОБ }, <i>подача</i> , <i>недоход</i> [, ВЫСТОЙ , <i>пауза</i>]

Параметры:

Параметр	Описание
<i>глубина</i>	Общая глубина обработки.
ММИН	Подача в мм/мин.
ММОБ	Подача в мм/об.
<i>подача</i>	Рабочая подача в цикле.

Параметр	Описание
<i>недоход</i>	Недоход до поверхности детали.
<i>пауза</i>	Выдержки времени в цикле в сек.

1.10.9.4 Цекование и зенкование

Циклы цекования и зенкования (G82) задается оператором **ЦИКЛ** следующего формата:

ЦИКЛ ТОРЦОВ , <i>глубина</i> , { ММИН , ММОБ }, <i>подача</i> , <i>недоход</i> [, ВЫСТОЙ , <i>пауза</i>]

Параметры:

Параметр	Описание
<i>глубина</i>	Общая глубина обработки.
ММИН	Подача в мм/мин.
ММОБ	Подача в мм/об.
<i>подача</i>	Рабочая подача в цикле.
<i>недоход</i>	Недоход до поверхности детали.
<i>пауза</i>	Выдержки времени в цикле в сек.

1.10.9.5 Коническое зенкование

Для задания цикла зенкования (G82) с расчетом глубины перемещения зенковки предназначен оператор **ЦИКЛ** следующего формата:

ЦИКЛ КЗЕНК , <i>диаметр</i> , <i>угол</i> , { ММИН , ММОБ }, <i>подача</i> , <i>недоход</i> [, ВЫСТОЙ , <i>пауза</i>]

Параметры:

Параметр	Описание
<i>диаметр</i>	Диаметр зенкера на итоговой глубине.
<i>угол</i>	Рабочий угол зенкера.
ММИН	Подача в мм/мин.

Параметр	Описание
ММОБ	Подача в мм/об.
<i>подача</i>	Рабочая подача в цикле.
<i>недоход</i>	Недоход до поверхности детали.
<i>пауза</i>	Выдержки времени в цикле в сек.

Глубина обработки вычисляется по формуле:

$$глубина = \frac{диаметр}{2 \operatorname{tg}(угол / 2)}$$

1.10.9.6 Нарезание резьбы

Цикл нарезания резьбы задается оператором **ЦИКЛ** следующего формата:

ЦИКЛ { РЕЗБМЕТ РЕЗБМЕТ1 }, <i>глубина</i> , ММОБ , <i>шаг</i> , <i>недоход</i> [, ВЫСТОЙ , <i>пауза</i>]

Параметры:

Параметр	Описание
РЕЗБМЕТ	Цикл нарезания правой резьбы (G84).
РЕЗБМЕТ1	Цикл нарезания левой резьбы (G63).
<i>глубина</i>	Общая глубина обработки.
<i>шаг</i>	Шаг резьбы.
<i>недоход</i>	Недоход до поверхности детали.
<i>пауза</i>	Выдержки времени в цикле в сек.

 Величина шага резьбы равна значению подачи в мм/об.

1.10.9.7 Растачивание

Цикл растачивания (G85-G89) задается оператором **ЦИКЛ** следующего формата:

$$\text{ЦИКЛ} \left\{ \begin{array}{l} \text{РАСТОЧ} \\ \text{РАСТОЧ5} \\ \text{РАСТОЧ6} \\ \text{РАСТОЧ7} \\ \text{РАСТОЧ8} \\ \text{РАСТОЧ9} \end{array} \right\}, \text{глубина}, \left\{ \begin{array}{l} \text{ММИН} \\ \text{ММОБ} \end{array} \right\}, \text{подача}, \text{недоход} [\text{,ВЫСТОЙ}, \text{пауза}]$$

Параметры:

Параметр	Описание
РАСТОЧ	Цикл растачивания G85.
РАСТОЧ5	Цикл растачивания G85.
РАСТОЧ6	Цикл растачивания G86.
РАСТОЧ7	Цикл растачивания G87.
РАСТОЧ8	Цикл растачивания G88.
РАСТОЧ9	Цикл растачивания G89.
<i>глубина</i>	Общая глубина обработки.
ММИН	Подача в мм/мин.
ММОБ	Подача в мм/об.
<i>подача</i>	Рабочая подача в цикле.
<i>недоход</i>	Недоход до поверхности детали.
<i>пауза</i>	Выдержки времени в цикле в сек.

1.10.9.8 Развертывание

Цикл развертывания (G86) задается оператором **ЦИКЛ** следующего формата:

$$\text{ЦИКЛ} \text{ РАЗВЕР}, \text{глубина}, \left\{ \begin{array}{l} \text{ММИН} \\ \text{ММОБ} \end{array} \right\}, \text{подача}, \text{недоход} [\text{,ВЫСТОЙ}, \text{пауза}]$$

Параметры:

Параметр	Описание
<i>глубина</i>	Общая глубина обработки.
ММИН	Подача в мм/мин.
ММОБ	Подача в мм/об.
<i>подача</i>	Рабочая подача в цикле.
<i>недоход</i>	Недоход до поверхности детали.
<i>пауза</i>	Выдержки времени в цикле в сек.

1.10.9.9 Зенкерование

Цикл зенкерования (G86) задается оператором **ЦИКЛ** следующего формата:

<p>ЦИКЛ РАСТОЧ6, <i>глубина</i>, { ММИН } { ММОБ }, <i>подача</i>, <i>недоход</i> [, ВЫСТОЙ , <i>пауза</i>]</p>

Параметры:

Параметр	Описание
<i>глубина</i>	Общая глубина обработки.
ММИН	Подача в мм/мин.
ММОБ	Подача в мм/об.
<i>подача</i>	Рабочая подача в цикле.
<i>недоход</i>	Недоход до поверхности детали.
<i>пауза</i>	Выдержки времени в цикле в сек.

1.10.9.10 Включение и выключение цикла

Вариант оператора **ЦИКЛ**, описанный в данном разделе предназначен для:

- повторного включения цикла;
- выключения цикла;
- ручного управления циклом.

Формат:

ЦИКЛ	{	ВКЛ	}
		ВЫКЛ	
		ОТМЕН	
		РУЧН	}

Параметры:

Параметр	Описание
ВКЛ	Повторное включение стандартного цикла. Включает ранее выключенный цикл с теми же параметрами.
ВЫКЛ ОТМЕН	Выключение стандартного цикла. Отменяет выполнение цикла в каждой точке траектории. Если необходимо возобновить выполнение цикла, следует задать оператор ЦИКЛ ВКЛ .
РУЧН	Ручное управление циклом. Оператор вызывает останов станка в каждой точке траектории для выполнения цикла вручную.

1.10.10 Останов станка

Операторы этой группы применяются для задания безусловного и условного остановов станка, а также выдержки времени.

Безусловный останов

Для безусловного останова станка предназначен оператор **СТОП**.

Формат:

СТОП

При последующем пуске станка обработка будет продолжена. Оператор используют в том случае, когда в процессе обработки требуется остановка (например, для контрольного замера детали).

Технологический останов

Для условного останова станка предназначен оператор **ТЕХОСТ**.

Формат:

ТЕХОСТ

Оператор позволяет задать останов станка и считывающего устройства включением соответствующего тумблера на пульте системы ЧПУ.

Пауза

Для задания паузы в работе станка предназначен оператор **ПАУЗА**.

Формат:

$$\text{ПАУЗА [ОБОРОТ] } \left\{ \begin{array}{l} \text{интервал} \\ \text{количество} \end{array} \right\}$$

Параметры:

Параметр	Описание
ОБОРОТ	Задание паузы числом оборотов шпинделя. Предварительно должна быть задана частота вращения шпинделя.
<i>интервал</i>	Величина паузы в секундах.
<i>количество</i>	Величина паузы в оборотах.

См. также:

- [Скорость вращения шпинделя](#) ²⁶⁶

1.10.11 Подготовительные и вспомогательные функции

Операторы этой группы предназначены для вставки в текущий кадр управляющей программы кодов подготовительных (типа G) и вспомогательных (типа M) функций. Эти операторы не следует использовать для задания функций, формируемых автоматически или по специальной команде, так как заданные значения не проверяются, что может привести к несоответствию состояния станка состоянию постпроцессора. Например, недопустимо задание таким способом кодов интерполяции, команд включения шпинделя, коррекции и т.п.

Вставка кодов подготовительных функций

Для вставки в кадр кодов подготовительных функций предназначен оператор **ПДГФУН**.

Формат:

$$\text{ПДГФУН } код_1 \left[\left[код_2 \left[код_3 \dots \left[код_9 \right] \right] \right] \right] \left[\text{ТЕКУЩ} \right]$$

Параметры:

Параметр	Описание
$код_p, \dots, код_9$	Коды подготовительных функций.

Параметр	Описание
ТЕКУЩ	Завершение формирования кадра.

Вставка кодов вспомогательных функций

Для вставки в кадр кодов вспомогательных функций предназначен оператор **ВСПФУН**.

Формат:

$$\text{ВСПФУН } код_1 \left[\left[код_2 \left[код_3 \dots \left[код_9 \right] \right] \right] \right] \left[\text{ТЕКУЩ} \right]$$

Параметры:

Параметр	Описание
$код_p, \dots, код_9$	Коды вспомогательных функций.
ТЕКУЩ	Завершение формирования кадра.

1.10.12 Управление поворотным столом

Для управления поворотным столом предназначен оператор **СТОЛ**. Он позволяет повернуть стол в определенную позицию или на заданный угол.

Формат:

$$\text{СТОЛ} \left\{ \begin{array}{l} \text{угол} \\ \text{ПРИРАЩ, угол}_1 \end{array} \right\} \left[\begin{array}{l} \text{ПОЧС} \\ \text{ПРЧС} \end{array} \right] \left[\text{ММИН, скорость} \right] \left[\begin{array}{l} \text{ОСЬХ} \\ \text{ОСЬУ} \\ \text{ОСЬЗ} \end{array} \right] \left[\text{ФИКС} \right]$$

Параметры:

Параметр	Описание
<i>угол</i>	Абсолютное значение угла поворота стола в градусах.
ПРИРАЩ , <i>угол</i> ₁	Приращение угла поворота стола в градусах
ПОЧС	Поворот стола по часовой стрелке.
ПРЧС	Поворота стола против часовой стрелки.
ММИН , <i>скорость</i>	Линейная скорости поворотного стола в мм/мин.

Параметр	Описание
ОСЬХ ОСЬУ ОСЬЗ	Ось вращения стола.
ФИКС	Фиксация поворотного стола.

1.10.13 Вставка текста

Операторы этой группы предназначены для вставки текста в управляющую программу и выходные документы, формируемые постпроцессором. К ним относятся операторы **ВСТАВ**, **ППЕЧ** и **ТЕКСТ**.

Темы этого раздела:

- [Вставка текста в управляющую программу](#)^[278]
- [Вставка текста в выходной документ](#)^[279]
- [Комментарий](#)^[279]

1.10.13.1 Вставка текста в управляющую программу

Для вставки текста в управляющую программу предназначен оператор **ВСТАВ**.

Формат:

[ВСТАВ] строка [,ТЕКУЩ]

Параметры:

Параметр	Описание
<i>строка</i>	Текст, вставляемый в управляющую программу в конец текущего кадра.
ТЕКУЩ	Вывод строки отдельным кадром.

 1. Если оператор состоит только из одного параметра строка, то это текст будет выведен отдельным кадром.

2. Для отмены нумерации кадра, следует задать первый символ в строке – ().

3. Оператором надо пользоваться с осторожностью, так как вставка текста в управляющую программу не контролируется и может привести к нежелательным последствиям.

Пример:

Для добавления текста в конец текущего кадра следует задать:

ВСТАВ 'М30'

Для вывода текста отдельным кадром без номера следует задать:

ВСТАВ '\O0010',ТЕКУЩ

То же действие произведет оператор:

'\O0010'

1.10.13.2 Вставка текста в выходной документ

Для вставки текста в выходные документы, формируемые постпроцессором, предназначен оператор **ППЕЧ**. Использование этого оператора регламентируется разработчиком модуля станка.

Формат:

ППЕЧ строка

Параметры:

Параметр	Описание
<i>строка</i>	Текст, вставляемый в выходной документ.

1.10.13.3 Комментарий

Оператор **ТЕКСТ** предназначена для передачи строки символов модулю станка. Использование этой команды регламентируется разработчиком модуля станка. Обычно команда используется для вставки произвольного текста в управляющую программу без нумерации. В отличие от команды вставки текста в УП этот процесс контролируется модулем станка.

Формат:

ТЕКСТ строка

Параметры:

Параметр	Описание
<i>строка</i>	Текст комментария.

1.10.14 Система координат станка

Операторы этой группы предназначены для задания местных систем координат (МСК), сдвига начала координат и для управления дополнительными осями. К ним относятся операторы **КООРДСТ** и **СДВИГ**. Способ расчета координат в постпроцессоре задается оператором **РЕЖИМ**.

Темы этого раздела:

- [Задание МСК смещениями](#)^[280]

- [Задание МСК функцией](#)^[280]
- [Дополнительная ось](#)^[280]
- [Сдвиг системы координат станка](#)^[281]
- [Способ расчета координат](#)^[281]

1.10.14.1 Задание местной системы координат смещениями

Вариант оператора **КООРДСТ**, описанный в данном разделе, предназначен для задания местной системы координат (МСК) посредством указания смещения ее основных и дополнительных осей относительно системы координат детали (абсолютного нуля).

Формат:

```
КООРДСТ x, y [,z] [,u] [,v] [,w]
```

Параметры:

Параметр	Описание
<i>x, y, z, u, v, w</i>	Величины осевых смещений начала системы координат станка относительно абсолютного нуля.

1.10.14.2 Задание местной системы координат функцией

Вариант оператора **КООРДСТ**, описанный в данном разделе, предназначен для задания местной системы координат (МСК) посредством указания ее кода.

Формат:

```
КООРДСТ РЕГ, номер
```

Параметры:

Параметр	Описание
<i>код</i>	Кода местной системы координат станка.

1.10.14.3 Дополнительная ось

Вариант оператора **КООРДСТ**, описанный в данном разделе, предназначен для управления дополнительными осями станка. Перемещение по каждой из трех осей системы координат может быть задано под различными адресами в соответствии с кинематикой конкретного станка. Команда позволяет для оси системы координат детали задать рабочий орган осуществляющий движение по этой оси.

Формат:

$$\text{КООРДСТ} \left\{ \begin{array}{l} \text{ОСЬХ} \\ \text{ОСЬУ} \\ \text{ОСЬЗ} \end{array} \right\} \left[\left\{ \begin{array}{l} \text{СУППОРТ} \\ \text{ПИНОЛЬ} \\ \text{ПОВСИС} \end{array} \right\} \right]$$

Параметры:

Параметр	Описание
ОСЬХ ОСЬУ ОСЬЗ	Ось в системе координат детали.
СУППОРТ	Движение по выбранной оси за счет перемещения суппорта.
ПИНОЛЬ	Движение по выбранной оси за счет перемещения пиноли
ПОВСИС	Движение по выбранной оси за счет перемещения стола

1.10.14.4 Сдвиг системы координат станка

Для сдвига системы координат станка относительно системы координат детали предназначен оператор **СДВИГ**. Смещения, заданные в этом операторе, суммируются с координатами точек.

Формат:

$$\text{СДВИГ } x, y [,z]$$

Параметры:

Параметр	Описание
x, y, z	Смещения системы координат.

 Повторные операторы **СДВИГ** заменяют предыдущий оператор.

1.10.14.5 Способ расчета координат

Вариант оператора **РЕЖИМ**, описанный в данном разделе, предназначен для выбора способа расчета координат: в абсолютных значениях или в приращениях.

Формат:

РЕЖИМ	{ АБСОЛЮТ ПРИРАЩ }
-------	-----------------------

Параметры:

Параметр	Описание
АБСОЛЮТ	Расчет координат в абсолютных значениях.
ПРИРАЩ	Расчет координат в приращениях.

1.10.15 Управление резкой

Операторы, описанные в данном разделе, используются для задания параметров электроэрозионной обработки и для управления резаками в машинах термической резки.

Темы этого раздела:

- [Управление электроэрозионной обработкой](#)^[282]
- [Управление термической резкой](#)^[282]

1.10.15.1 Управление электроэрозионной обработкой

Для задания параметров электроэрозионной обработки предназначен оператора **РЕЗКА**.

Формат:

РЕЗКА	{ ВКЛ ВЫКЛ }	, число ₁ , число ₂
-------	-----------------	-------------------------------------------

Параметры:

Параметр	Описание
ВКЛ	Включение резки.
ВЫКЛ	Выключение резки.
число ₁ , число ₂	Параметры электроэрозионной обработки. Определяется разработчиком модуля станка.

1.10.15.2 Управление термической резкой

Для управления резаками предназначен оператор **РЕЗАК**.

Темы этого раздела:

- [Задание ширины реза](#)^[283]

- [Включение и выключение резаков](#)^[283]
- [Управление боковыми резаками](#)^[283]
- [Управление разметкой](#)^[284]

1.10.15.2.1 Задание ширины реза

Ширина реза (диаметр режущего пламени или струи) задается оператором **РЕЗАК** следующего формата:

Формат:

РЕЗАК ДИАМЕТР, *диаметр*

Параметры:

Параметр	Описание
<i>диаметр</i>	Ширина реза.

1.10.15.2.2 Включение и выключение резаков

Включение и выключение резаков задается оператором **РЕЗАК** следующего формата:

Формат:

РЕЗАК { **ВКЛ**
ВЫКЛ }, [**СЛЕВА**
СРЕДН
СПРАВА]

Параметры:

Параметр	Описание
ВКЛ	Включение резака.
ВЫКЛ	Выключение резака.
СЛЕВА	Левый резак.
СРЕДН	Средний резак. Действует по умолчанию
СПРАВА	Правый резак.

1.10.15.2.3 Управление боковыми резаками

Смещение и наклон боковых резаков задаются оператором **РЕЗАК** следующего формата:

Формат:

$$\text{РЕЗАК} \left\{ \begin{array}{l} \text{СЛЕВА} \\ \text{СПРАВА} \end{array} \right\}, \text{УГОЛ}, \text{угол}, \text{БОК}, \text{высота}$$

Параметры:

Параметр	Описание
СЛЕВА	Левый резак.
СПРАВА	Правый резак.
<i>угол</i>	Угол наклона бокового резака.
<i>высота</i>	Высота фаски

1.10.15.2.4 Управление разметкой

Для управления устройством разметки предназначен оператор **РАЗМЕТ**.

Формат:

$$\text{РАЗМЕТ} \left\{ \begin{array}{l} \text{ВКЛ} \\ \text{ВЫКЛ} \end{array} \right\}$$

Параметры:

Параметр	Описание
ВКЛ	Включение устройства разметки.
ВЫКЛ	Выключение устройства разметки.

1.10.16 Специальные функции постпроцессора

Для программирования действий, которые не поддерживаются стандартными операторами, служат оператор **ППФУН**. Применение этого оператора регламентируется разработчиком модуля станка.

Формат:

$$\text{ППФУН} \text{ литерал} \left[\begin{array}{l} \text{служебное слово}_1 \\ \text{число}_1 \end{array} \right] \dots \left[\begin{array}{l} \text{служебное слово}_{10} \\ \text{число}_{10} \end{array} \right]$$

Параметры:

Параметр	Описание
<i>литерал</i>	Название специальной функции постпроцессора (до 6 символов).
<i>число₁ ...число₁₀</i>	Числовой параметр.
<i>служебное слово₁ ... служебное слово₁₀</i>	Служебное слово.

 Всего может быть задано до 10 параметров (чисел и служебных слов).

1.10.17 Пример использования операторов постпроцессора

В примере, приведенном ниже, показана типичная последовательность операторов постпроцессора, необходимых при программировании обработки на многоинструментальном станке:

```

ДЕТАЛЬ 'ПРОГРАММА 1'           # Начало программы
СТАНОК '2206 ВМФ4', 720       # Вызов постпроцессора
# *****
# *****
ОПИСАНИЕ ИНСТРУМЕНТОВ

ИНСТР Т1 = ЦЕНТРОВ 1, 10, ДЛИНА, 200, НОМКОР, ХКООРД, 1,
УКООРД, 2, ЗКООРД, 3
ИНСТР Т2 = ЦЕНТРОВ 2, 7, ДЛИНА, 150, НОМКОР, ХКООРД, 11,
УКООРД, 12, ЗКООРД, 13
ИНСТР Т3 = ФРЕЗА 3, 20, ДИАМЕТР, 40
ИЗ 0, 0                       # Начало движения
# Переход 1: Инструмент Т1
ВЫБОРИН 1                     # Выбор 1-го инструмента
ЗАГРУЗ Т1                     # Загрузка 1-го инструмента
КОРРЕК ВКЛ                   # Ввод коррекции положения
БЫСТРО                       # Быстрая подача
ВТОЧКУ ТЧ1
ПОДАЧА 100                   # Подача 100 мм/мин
ОХЛАД ВКЛ                   # Включение охлаждения
ШПИНДЛ 600                   # Включение шпинделя вправо
КОРРЕК СПРАВА, РАДИУС, 6     # Ввод коррекции радиуса
ИДИ ДО ПР1
. . . . .
ВЫБОРИН Т2                   # Выбор 2-го инструмента
. . . . .
КОРРЕК ВЫКЛ, РАДИУС         # Отмена коррекции радиуса
КОРРЕК ВЫКЛ                 # Отмена всех корректоров
ВОЗВРАТ                     # Возврат в исходную позицию
# Переход 2: Инструмент Т2
ЗАГРУЗ Т2                   # Загрузка 2-го инструмента
ТЕХОСТ                      # Технологический останов

```

СТОЛ 15, ПОЧС	# Поворот стола на 15 град по час.стр.
ПОДАЧА 200	# Подача 200 мм/мин
ШПИНДЛ 1200	# Обороты шпинделя
.....	
ВЫБОРИН Т3	# Выбор 3-го инструмента
.....	
# Переход 3: Инструмент Т3	
ЗАГРУЗ Т3	# Загрузка 3-го инструмента
ПОДАЧА ММИН,50	# Подача 50 мм/мин
ШПИНДЛ 800	# Обороты шпинделя
.....	
ШПИНДЛ ВЫКЛ	# Выключение шпинделя
СТОП	# Безусловный останов
КОНЕЦ	

1.11 Примеры программирования обработки

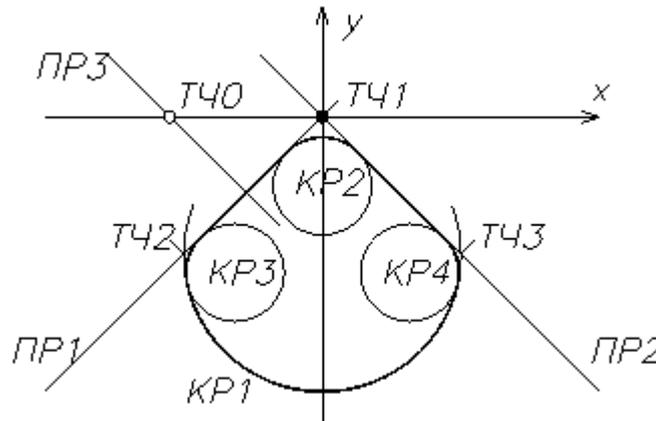
Данная глава содержит примеры программирования, иллюстрирующие основные возможности языка и приемы рационального написания программ. Примеры использования дополнительных возможностей языка, доступных при программировании фрезерной, токарной и электроэрозионной обработки приведены в соответствующих руководствах.

Темы этого раздела:

- [Пример программирования простейшей траектории](#)^[287]
- [Пример использования матриц](#)^[288]
- [Пример использования контуров](#)^[289]
- [Пример использования циклов и преобразования траектории](#)^[290]
- [Пример программирования сплайнов](#)^[293]
- [Пример использования макросов](#)^[294]

1.11.1 Пример программирования простейшей траектории

Ниже приведен пример рисунок простейшей траектории движения инструмента и программа ее обработки:



ДЕТАЛЬ 'ПРИМЕР 1'

#

ОПИСАНИЕ ЭЛЕМЕНТОВ КОНТУРА

#

ТЧ0 = -30, 0

ТЧ1 = 0, 0

ПР1 = ТЧ1, 45

ПР2 = ТЧ1, -45

ТЧ2 = ПР1, ТЧ1, 50, УМ

ТЧ3 = ПР2, ТЧ1, 50, УМ

ПР3 = ТЧ0, ПЕРП, ПР1

КР1 = ТЧ1, ТЧ2, ТЧ3

КР2 = УМ, ПР1, УМ, ПР2, РАДИУС, 10

КР3 = УМ, ПР1, ХМ, ВНУТРИ, КР1, 10

КР4 = УМ, ПР2, ХБ, ВНУТРИ, КР1, 10

#

ИНСТР 6

ДИАМЕТР ИНСТРУМЕНТА

МЕТОД ЛИНКРУГ

МЕТОД ИНТЕРПОЛЯЦИИ

ИНСТПР

ИНСТРУМЕНТ СПРАВА

#

***** ОПИСАНИЕ ТРАЕКТОРИИ ДВИЖЕНИЯ ИНСТРУМЕНТА

#

ИЗ ТЧ0

НАЧАЛО ДВИЖЕНИЯ

ИДИ ДО ПР1

ПОДХОД К КОНТУРУ

ВПРАВО ПР1

ОБХОД КОНТУРА СНАРУЖИ

ВПЕРЕД КР3:КР1:КР4:ПР2: % # ПРОТИВ ЧАСОВОЙ СТРЕЛКИ

КР2:ПР1 НА ПР3 #

ВТОЧКУ ТЧ0

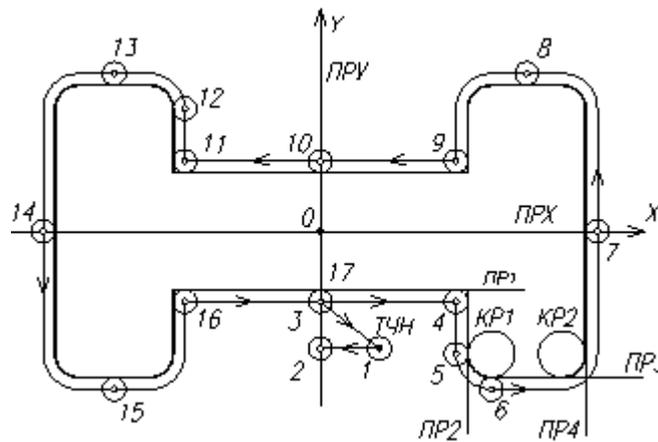
ВОЗВРАТ В ИСХ. ТОЧКУ

КОНЕЦ

КОНЕЦ ПРОГРАММЫ

1.11.2 Пример использования матриц

Ниже приведен пример использования матриц симметрии. Контур, изображенный на рисунке, имеет две оси симметрии - X и Y. В качестве базового выбран элемент, лежащий в 4-ом квадранте. Обход элементов, находящихся в 1-ом и 3-м квадрантах, задается симметричным отображением траектории обработки базового элемента относительно осей X и Y соответственно. Преобразование траектории для обработки части контура, лежащей во 2-м квадранте, осуществляется по матрице M3, являющейся произведением матриц симметрии относительно осей X и Y.



ДЕТАЛЬ 'ПРИМЕР 2'

ПРХ = ХПАР,0

ПРУ = УПАР, 0

ПР1 = ХПАР, -10

ПР2 = УПАР, 40

ПР3 = ХПАР, -35

ПР4 = УПАР, 70

КР1 = ХБ, ПР2, УБ, ПР3, 5

КР2 = ХМ, ПР4, УБ, ПР3, 5

ТЧН = 10, -20

ИНСТПР

ИНСТР 3

МАТР М1, М2, М3

#

М1 = ПРХ

СИММЕТРИЯ

ОТНОСИТЕЛЬНО ОСИ X

М2 = ПРУ

СИММЕТРИЯ

ОТНОСИТЕЛЬНО ОСИ Y

М3 = М1, М2

ПРОИЗВЕДЕНИЕ МАТРИЦ

СИММЕТРИИ

ИЗ ТЧН

ПОЗ. 1, ИСХОДНАЯ

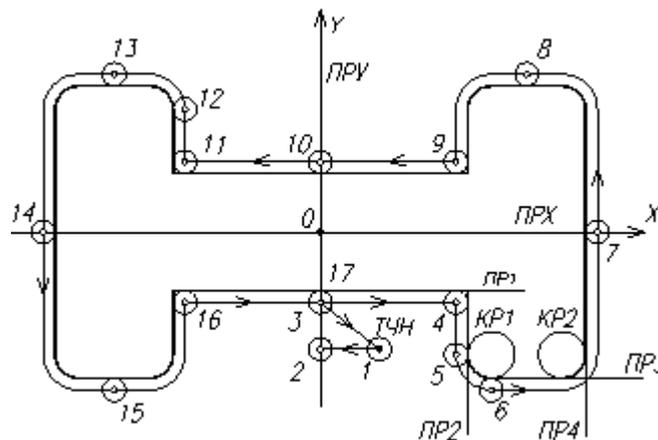
ИДИ НА ПРУ

ПОЗ. 1 - 2

ВПРАВО ПРУ:ПР1:ПР2	# ПОЗ. 2 - 3 - 4 - 5
ВПЕРЕД КР1:ПР3:КР2:ПР4 НА ПРХ	# ПОЗ. 5 - 6 - 7
ТРАНС М1	
ВПЕРЕД ПР4:КР2:ПР3:КР1:ПР2	# ПОЗ. 7 - 8 - 9
ВЛЕВО ПР1 НА ПР	# ПОЗ. 9 - 10
ТРАНС М3	
ВПЕРЕД ПР1	# ПОЗ. 10 - 11
ВПРАВО ПР2	# ПОЗ. 11 - 12
ВПЕРЕД КР1:ПР3:КР2:ПР4 НА ПРХ	# ПОЗ. 12 - 13 - 14
ТРАНС М2	
ВПЕРЕД ПР4:КР2:ПР3:КР1:ПР2	# ПОЗ. 14 - 15 - 16
ВЛЕВО ПР1 НА ПРУ	# ПОЗ. 16 - 17
ТРАНС ОТМЕН	
ВТОЧКУ ТЧН	# ПОЗ. 17 - 1, ВОЗВРАТ В ТЧН
КОНЕЦ	

1.11.3 Пример использования контуров

Ниже приведен пример использования объектов типа **КОНТУР**. В качестве базового выбран элемент, лежащий в 4-ом квадранте. Обход элементов, находящихся в 1-ом и 3-м квадрантах, задается симметричным отображением траектории обработки базового элемента относительно осей **X** и **Y** соответственно. Преобразование траектории для обработки части контура, лежащей во 2-м квадранте, осуществляется по матрице **М3**, являющейся произведением матриц симметрии относительно осей **X** и **Y**.



ДЕТАЛЬ 'ПРИМЕР 2А'
 ПРХ=ХПАР,0
 ПРУ=УПАР,0
 ПР1=ХПАР,-10
 ПР2=УПАР,40
 ПР3=ХПАР,-35
 ПР4=УПАР,70
 КР1=ХБ,ПР2,УБ,ПР3,5
 КР2=ХМ,ПР4,УБ,ПР3,5

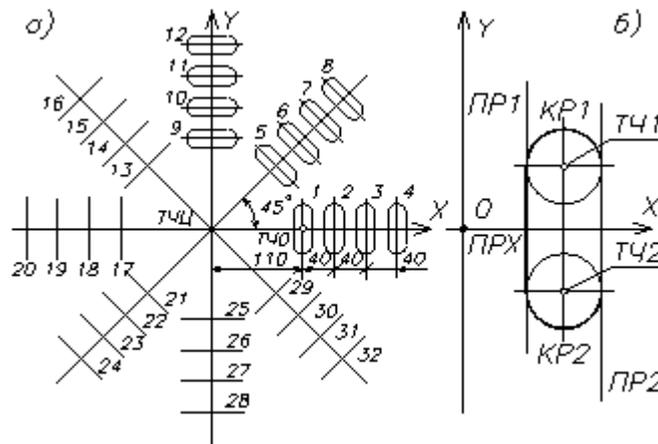
```

ТЧ1=ПР1,ПРУ
ТЧН=10,-20
ИНСТПР
ИНСТР 3
МАТР М1, М2, М3
М1=ПРХ # СИММЕТРИЯ
ОТНОСИТЕЛЬНО ОСИ Х
М2=ПРУ # СИММЕТРИЯ
ОТНОСИТЕЛЬНО ОСИ У
М3=М1, М2 # ПРОИЗВЕДЕНИЕ МАТРИЦ
СИММЕТРИИ
#
КОНТУР К=ИЗ ТЧ1,УБ # ИСХОДНЫЙ КОНТУР (В 4-ОМ
КВАДРАНТЕ)
 ВПРАВО ПР1:ПР2
 ВПЕРЕД КР1:ПР3:КР2:ПР4 НА ПРХ
КОНКОН
КОНТУР К1= К, % # 4-ЫЙ КВАДРАНТ
 ОБРАТН, М1, К, % # 1-ЫЙ КВАДРАНТ
 М3, К, % # 2-ОЙ КВАДРАНТ
 ОБРАТН, М2, К # 3-ИЙ КВАДРАНТ
#
ИЗ ТЧН # ПОЗ. 1, ИСХОДНАЯ
ИДИ НА ПРУ # ПОЗ. 1-2
ВПРАВО ПРУ ДО К1 # ПОЗ. 2-3, ПОДХОД К КОНТУРУ
ВПРАВО К1 # ОБХОД КОНТУРА
ВТОЧКУ ТЧН # ПОЗ. 17-1, ВОЗВРАТ В ИСХОДНУЮ
ТОЧКУ
КОНЕЦ

```

1.11.4 Пример использования циклов и преобразования траектории

Ниже приводится текст программы, в которой используются матричные преобразования и вложенные циклы. Запрограммирована обработка 32 элементов (рис. а), полученных сдвигом и поворотом базового элемента (рис. б).



Траектория обработки задана с помощью двух вложенных циклов: внешний цикл - поворот на 0, 45, 90, ... и т.д. градусов; внутренний цикл - перенос на 0, 40, 80 и 120 мм. Первое выполнение тела цикла обеспечивает обход четырех элементов без их разворота, затем величина сдвига сбрасывается и внутренний цикл выполняется для нового угла поворота, отличного от нуля. Элементы обрабатываются в следующем порядке: 1-2-3-4-5-6-7-8-9-10... и т.д.

ДЕТАЛЬ 'ПРИМЕР 3'

МАТР МПЕРЕНОС, МПОВОРОТ, М

ТЧЦ=0,0

ТЧ0=110,0

ПРХ=ХПАР,0

ПР1=УПАР,100

ТЧ1=110,20

ТЧ2=110,-20

КР1=ЦЕНТР,ТЧ1,КАС,ПР1

КР2=ТЧ2,ПР1

ПР2=СЛЕВА,КР1,СЛЕВА,КР2

МПЕРЕНОС=ПЕРЕНОС,0,0

МПОВОРОТ=ПОВОРОТ,0

ИНСТР 5

ИНСТПР

ИЗ ТЧЦ

ПОВТОР НОМПОВ=1,8

СЧЕТЧИК ЧИСЛА

ПОВОРОТОВ

ПОВТОР НОМПЕР=1,4

СЧЕТЧИК ЧИСЛА

ПЕРЕНОСОВ

М=МПОВОРОТ,МПЕРЕНОС

ПРОИЗВЕДЕНИЕ

МАТРИЦ

ТРАНС М

ПРЕОБРАЗОВАНИЕ

ТРАЕКТОРИИ

ВТОЧКУ ТЧ0

ВЫХОД В ЦЕНТР

ОЧЕРЕДНОГО ПАЗА

ПРИРАЩ -50

ОПУСКАНИЕ ПО Z

ИДИ ДО ПР1

ОБХОД

```

ВПРАВО ПР1 # ЭЛЕМЕНТОВ КОНТУРА
ВПЕРЕД КР1:ПР2:КР2:ПР1 НА ПРХ
ПРИРАЩ 50 # ПОДЪЕМ ПО Z
МПЕРЕНОС=ПЕРЕНОС, 40*НОМПЕР, 0 # НАРАЩИВАНИЕ
ПЕРЕНОСА НА 40 ММ
КОНЦИКЛ НОМПЕР
МПЕРЕНОС=ПЕРЕНОС,0,0 # СБРОС ПЕРЕНОСА
МПОВОРОТ=ПОВОРОТ, 45*НОМПОВ # НАРАЩИВАНИЕ
ПОВОРОТА НА 45 ГРАД
КОНЦИКЛ НОМПОВ
ТРАНС ОТМЕН
ВТОЧКУ ТЧЦ # ВОЗВРАТ В ИСХОДНУЮ
ТОЧКУ
КОНЕЦ

```

Программа обработки тех же элементов, написанная с использованием контуров и массивов объектов, имеет вид:

```

ДЕТАЛЬ 'ПРИМЕР 3А'
ТЧЦ=0,0
ТОЧКА МТЧ1()=ТОЧКА(110-40,0),ПЕРЕНОС,40,0,4 # МАССИВ
ТОЧЕК 1,2,3,4
ТОЧКА МТЧ2()=МТЧ1,МАТР(ПОВОРОТ,45),8 #
КОПИРУЕМ МАССИВ 8 РАЗ С ПОВОРОТОМ
ПРХ=ХПАР,0
ПР1=УПАР,100
ТЧ1=110,20
ТЧ2=110,-20
КР1=ЦЕНТР,ТЧ1,КАС,ПР1
КР2=ТЧ2,ПР1
ПР2=СЛЕВА,КР1,СЛЕВА,КР2
КОНТУР К0=ИЗ ТОЧКА ТЧ3(ПРХ, ПР1), УБ # КОНТУР
ОДНОГО ПАЗА
    ВПЕРЕД ПР1:КР1:ПР2:КР2:ПР1 НА ТЧ3
КОНКОН
КОНТУР МК0()=К0,ПЕРЕНОС,МТЧ1,МТЧ1(1),ПОВОРОТ,0 # МАССИВ
КОНТУРОВ
КОНТУР МК1()=МК0,МАТР(ПОВОРОТ,45),8 #
КОПИРУЕМ МАССИВ 8 РАЗ С ПОВОРОТОМ
#
ИНСТР 5
ИНСТПР
ИЗ ТЧЦ # ИСХОДНОЕ
ПОЛОЖЕНИЕ
ПОВТОР КОЛ=1,32 # СЧЕТЧИК ЧИСЛА
ПАЗОВ
    ВТОЧКУ МТЧ2(КОЛ) # ВЫХОД В ЦЕНТР
ОЧЕРЕДНОГО ПАЗА
    ПРИРАЩ -50 # ОПУСКАНИЕ ПО

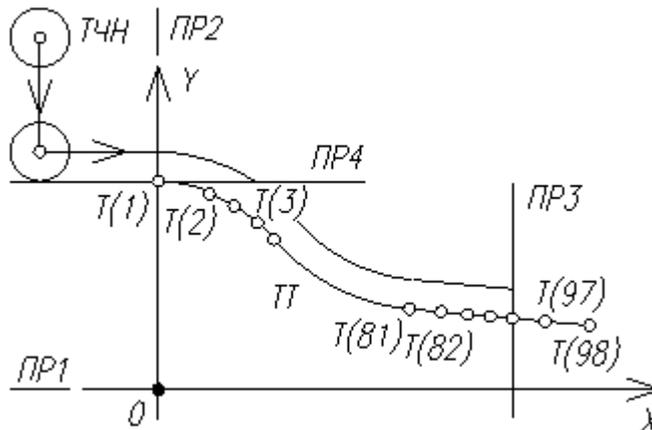
```

```

Z
  НАПРТ ТЧЦ
  ИДИ ДО МК1(КОЛ)
  ВПРАВО МК1(КОЛ)
  ПАЗА
  ПРИРАЩ 50
  КОНЦИКЛ КОЛ
  ВТОЧКУ ТЧЦ
  ИСХОДНУЮ ТОЧКУ
  КОНЕЦ
  # ПОДХОД
  # И ОБРАБОТКА
  # ПОДЪЕМ ПО Z
  # ВОЗВРАТ В
    
```

1.11.5 Пример программирования сплайнов

Следующий пример иллюстрирует применение таблично заданных кривых и расчет координат точек, заданных формулой. Контур детали представляет собой сложную кривую.



Контур детали задан в параметрическом виде:

$$y = l / 2 \cdot \sqrt{k1 \cdot \cos^2 t + k2 \cdot \cos t}$$

$$x = y \cdot \operatorname{tg} t$$

где: $k1 = 12544$; $k2 = 9956$ для $0 \leq x \leq 250$ и $0 \leq t \leq 89^\circ$

```

ДЕТАЛЬ'ПРИМЕР 4'
МЕТОД ЛИНЕЙН
ВЕЩ ПРИРУГ
ТОЧКА Т(100)
НОМЕРТОЧ=0
УГОЛП=0
УГЛА
ПРИРУГ=1
# ***** ЦИКЛ ВЫЧИСЛЕНИЯ КООРДИНАТ X и Y ТОЧЕК
*****
ПОВТОР ПОКА УГОЛП <= 88
  # НАЧАЛЬНЫЙ НОМЕР ТОЧКИ
  # НАЧАЛЬНОЕ ЗНАЧЕНИЕ
  # ШАГ ПРИРАЩЕНИЯ УГЛА
    
```

```

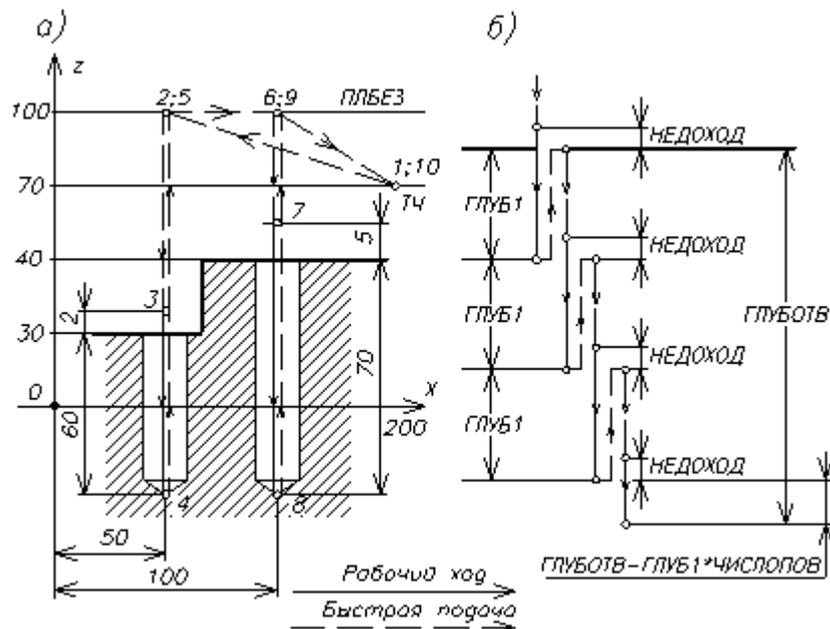
НОМЕРТОЧ=НОМЕРТОЧ+1          # ТЕКУЩИЙ НОМЕР ТОЧКИ
У=0.5*КВКОР(12544*КОС(УГОЛП)**2+9956*КОС(УГОЛП))
Х=У*ТАН(УГОЛП)
Т(НОМЕРТОЧ)=Х,У
ЕСЛИ (УГОЛП>80) ПРИРУГ=0.5    # ИЗМЕНЕНИЕ ШАГА
ПРИРАЩЕНИЯ

                                # УГЛА НА ПОЛОГОМ УЧАСТКЕ
                                # ТЕКУЩЕЕ ЗНАЧЕНИЕ
    УГОЛП=УГОЛП+ПРИРУГ
УГЛА
КОНЦИКЛ
ПР1=ХПАР,0
ПР2=УПАР,0
ПР3=УПАР,250
ПР4=Т(1),0
ТЧН=-10,100
КОНТУР ТТ=СПЛАЙН, ПР4
    ПОВТОР И=1,НОМЕРТОЧ
        Т(И)
    КОНЦИКЛ
КОНКОН
#
ИЗ ТЧН
ИНСТР 10
ИДИ ДО ПР4
ВЛЕВО ПР4 ДО ТТ
ВПЕРЕД ТТ НА ПР3
КОНЕЦ

```

1.11.6 Пример использования макросов

Ниже приведен пример программирования обработки с помощью макроса. На следующем рисунке изображена траектория сверления двух глубоких отверстий (рис. *а*), каждое из которых обрабатывается по циклу глубокого сверления с выводом инструмента (рис. *б*).



Программа содержит описание и два вызова макроса **ГЛУБСВ**, заданного в теле программы и определяющего траекторию глубокого сверления одного отверстия. Операторный параметр вида **ТОЧКА ТЧ=50,100,30** находящийся в первом вызове, определяет переменную **ТЧ** как локальную в макросе **ГЛУБСВ**. Благодаря этому другая переменная с таким же именем, но определенная в головной программе, сохраняет свое первоначальное значение после выхода из макроса. Остальные параметры будут локальными переменными макроса, если переменные с такими же именами не будут использоваться в головной программе.

Параметр **ПЛБЕЗ** задан как глобальная переменная, значение которой доступно в макросе. Поэтому для каждого из вызовов начальное и конечное значение инструмента по координате **Z** равны 100 мм. Параметр **НЕДОХОД** -текстовый. Его значение по умолчанию равно 2, что определено в заголовке макроса, поэтому для первого из вызовов недоход до поверхности детали составит 2 мм, а для второго будет равно значению параметра **НЕДОХОД**, заданному в вызове, т.е. 5 мм.

ДЕТАЛЬ 'ПРИМЕР 5'

МАКРО ГЛУБСВ('НЕДОХОД'='2')

```
#
# ***** Г Л У Б О К О Е  С В Е Р Л Е Н И Е  *****
#
# ПАРАМЕТРЫ МАКРОСА :
# ТЧ          - ТОЧКА, В КОТОРОЙ НУЖНО ВЫПОЛНИТЬ ЦИКЛ
# ПЛБЕЗ       - ПЛОСКОСТЬ БЕЗОПАСНОСТИ -РАССТОЯНИЕ ДО
# ПЛОСКОСТИ
#             БЕЗОПАСНОСТИ ОТ ПЛОСКОСТИ ХОУ
# НЕДОХОД    - ВЕЛИЧИНА НЕДОХОДА ДО МАТЕРИАЛА;
#             ПО УМОЛЧАНИЮ - 2ММ
# ГЛУБОТВ    - ОБЩАЯ ГЛУБИНА ОТВЕРСТИЯ
# ГЛУБ1      - ГЛУБИНА ОДНОГО ПРОХОДА -ШАГ СВЕРЛЕНИЯ
```

```

# ПОД          - РАБОЧАЯ ПОДАЧА НА КАЖДОМ ШАГЕ СВЕРЛЕНИЯ
#
# ЛОКАЛЬНЫЕ ПЕРЕМЕННЫЕ :
# ЧИСЛПОВ      - ЧИСЛО ВРЕЗАНИЙ НА ГЛУБИНУ ГЛУБ1
# ТЧ1          - ТОЧКА НА ПЛОСКОСТИ БЕЗОПАСНОСТИ, В КОТОРОЙ
#              НАЧИНАЕТСЯ И В КОТОРОЙ ЗАКАНЧИВАЕТСЯ ЦИКЛ
#
ВЕЩ ЧИСЛПОВ; ТОЧКА ТЧ1          #ЛОКАЛЬНЫЕ ПЕРЕМЕННЫЕ
ТЧ1=КООРД(1,ТЧ),КООРД(2,ТЧ),ПЛБЕЗ
БЫСТРО
ВТОЧКУ ТЧ1                      # ВЫХОД В ПЛ.
БЕЗОПАСНОСТИ
ПРИРАЩ -(ПЛБЕЗ-(КООРД(3,ТЧ)+&НЕДОХД))
ЧИСЛПОВ=ЦЕЛ(ГЛУБОТВ/ГЛУБ1)    # ЧИСЛО ВРЕЗАНИЙ
ПОВТОР И=1,ЧИСЛПОВ # НА ГЛУБИНУ ГЛУБ1
ПОДАЧА ММИН,ПОД
ПРИРАЩ -(ГЛУБ1+&НЕДОХД) # ВРЕЗАНИЕ
БЫСТРО
ПРИРАЩ ГЛУБ1 # ПОД'ЕМ
ПРИРАЩ -(ГЛУБ1-&НЕДОХД) # ОПУСКАНИЕ
КОНЦИКЛ
ПОДАЧА ММИН,ПОД                # ПОСЛЕДНЕЕ ВРЕЗАНИЕ
ПРИРАЩ -(ГЛУБОТВ-ГЛУБ1*ЧИСЛПОВ+&НЕДОХД)
БЫСТРО
ВТОЧКУ ТЧ1                      # ВЫХОД В ПЛ.
БЕЗОПАСНОСТИ
КОНМАК
#
ТЧ=200,100,70
ТЧ100=100,100,40
ИЗ ТЧ                            # ИСХОДНАЯ ТОЧКА
ПЛБЕЗ=100                        # ГЛОБАЛЬНАЯ ПЕРЕМЕННАЯ:
                                # ЕЕ ЗНАЧЕНИЕ ДОСТУПНО В

МАКРОСЕ
ВЫЗОВ ГЛУБСВ((ТОЧКА ТЧ=50,100,30),ГЛУБОТВ=60, %
              ГЛУБ1=18,ПОД=50)
ВЫЗОВ ГЛУБСВ((ТОЧКА ТЧ=ТЧ100),'НЕДОХД'='5', %
              ГЛУБОТВ=70,ГЛУБ1=18,ПОД=70)
ВТОЧКУ ТЧ                          # ВОЗВРАТ В ИСХОДНОЕ
ПОЛОЖЕНИЕ
КОНЕЦ

```

 Приведенный в примере макрос можно использовать при программировании глубокого сверления на станках, у которых отсутствует такой стандартный цикл.

Предметный указатель

- В -

вывод результатов 248
выражения 21
 арифметические 21
 логические 22
 условные 24

- Г -

геометрические определения 32
 вектора 92
 вложенное определение 115
 внутреннее представление 33
 массивы контуров 142
 массивы окружностей 139
 массивы прямых 132
 массивы точек 115
 матрицы 99
 окружности 74
 плоскости 110
 прямые 56
 точки 35

- И -

инструмент 154
 перемещение 156

- К -

контур 205
 движение по контуру 228
 определение 206
 редактирование 236
 траектория как контур 238

- М -

макросы 239
 вложенные вызовы 246
 вызов 243
 заголовок 242
 локализация переменных 247
 метки 248
 параметры 240
 структура 239
массивы
 контуров 142
 окружностей 139
 прямых 132
 точек 115

- О -

операторы
 постпроцессора 252
 специальные 203
 управляющие 194

- П -

перемещение инструмента 156
 по элементам 156
 преобразование 178
 управление формированием 183
присваивание 30

- С -

специальные операторы 203
стандартные функции 24
строки 24

- Т -

Техпост
 операторы прерывания выполнения 203

- У -

управляющие операторы 194

- Ф -

функции 24