

Министерство высшего и среднего специального образования  
РСФСР  
Ордена Трудового Красного Знамени  
НАУЧНО-ИССЛЕДОВАТЕЛЬСКИЙ РАДИОФИЗИЧЕСКИЙ  
ИНСТИТУТ

Препринт № 45



ИИРФИ

В.В.Бочаров, Л.П.Киселев, Г.Н.Пинчук

AMESPLOT - СИСТЕМА ВЫВОДА ИНФОРМАЦИИ  
В ГРАФИЧЕСКОЙ ФОРМЕ

В ы п у с к 2

Основные средства системы

г.Горький,  
1973

# О Г Л А В Л Е Н И Е

АЛФАВИТНЫЙ УКАЗАТЕЛЬ ОПЕРАТОРОВ .....	3
ВВЕДЕНИЕ .....	5
ОПЕРАТОРЫ ПЕРЕХОДА ЧЕРЕЗ УРОВНИ .....	7
КОМАНДЫ ОПИСАНИЯ ПЕРВИЧНЫХ ОСЕЙ .....	8
СРЕДСТВА СИСТЕМЫ .....	11
Средства системы для чертежа и субчертежа.	11
Средства системы для кривой ....	15
Средства для вывода текста ..	21
ПРИЛОЖЕНИЕ 1 .....	24
ПРИЛОЖЕНИЕ 2 .....	27

# I. Алфавитный указатель операторов

	Уровень	код	Стр.
angle ( $\varphi, x, y$ )	I-3	27	18
angls ( $\varphi$ )	I-3	37	21
atran ( $r, a, b$ )	I-3	30	18
begpl ( $x, y$ )	I-3	01	5
blank ( $i, p$ )	2-3	06	11
btran ( $r, a, b$ )	I-3	31	18
cross	I-3	05	12
curve ( $x, y, n, i, j, h$ )	3	33	19
dubax ( $i, p1, p2$ )	3	17	13
endgr	I-3	52	6
frame ( $i, m$ )	2-3	14	12
graf ( $xmin, str, ymin, sty$ )	2	11	8
graph ( $xmin, xmax, ymin, ymax$ )	2	10	7
heigh ( $h$ )	I-3	36	20
intrp ( $i, p$ )	I-3	20	14
lglg ( $xmin, tx, ymin, ty$ )	2	12	8
markr ( $i$ )	I-3	24	15
mess ( $s, n, x, y$ )	2-3	40	21
mrscd ( $l, q$ )	I-3	22	16
orign ( $physx, physy$ )	I	03	10
page ( $listx, listy$ )	I	02	10

	уровень	код	стр.
polar (k, str, plx, ply)	2	I3	9
realn (n, x, y)	2-3.	4I	22
rsblk (i)	2-3	07	II
rstrn	I-3	32	I9
scale (mx, my)	I-3	25	I7
shift (dx, dy)	I-3	26	I7
stlin (i, ipen)	I-3	2I	I6
title (dx, dy, s1, n1, s2, n2, s3, n3) I		04	6

## ВВЕДЕНИЕ

Данный выпуск посвящен описанию основных средств системы и может быть использован как справочное руководство по системе **AMESPLOT**.

Основная часть выпуска содержит сведения о семантике операторов и их синтаксисе в машинном коде. В приложении 1 собраны правила записи операторов на входном языке TA-2, а во втором приложении приведена таблица длин программных модулей, реализующих некоторые операторы.

При описании машинного формата мы будем придерживаться следующих обозначений:  
идентификатор (например, **dist** ) будет обозначать адрес значения или адрес начала массива значений,  
идентификатор с подстрочной восьмеркой (например, **S3<sub>8</sub>** ) будет обозначать само значение, задаваемое в виде целого восьмеричного числа, занимающего один адрес или некоторое место в адресе (например, **0i<sub>8</sub>00**, где **i<sub>8</sub>** есть восьмеричная цифра).

Для сокращения синтаксических правил, приведенных в приложении 1, введем два вспомогательных метапонятия:  
<массив> : := <ид-тор массива> | <ид-тор массива> '0' <число>  
<скаляр> : := <массив> | <ид-тор простой переменной> | <число>.

Здесь конструкция '0', <число> является модификатором. Модификатор изменяет адрес начала массива, т.е. адрес начала массива получается в результате сложения по модулю  $2^{12}$  значения модификатора с адресом начала модифицируемого массива (массивы должны располагаться в ОЗУ).

## ОПЕРАТОРЫ ПЕРЕХОДА ЧЕРЕЗ УРОВНИ

### VEGRL (x, y)

В последовательности команд построения чертежа выполнение оператора **VEGRL** всегда должно быть первым, в противном случае действия системы неопределены.

Формат    01     $x$      $y$     0000

Оператор **VEGRL** выполняет следующие действия:

- 1) устанавливает уровень 1,
- 2) определяет параметры:
  - размеры листа 215 x 152 мм,
  - физическое начало координат (0,0),
  - резервируемые области на чертеже отсутствуют,
  - точка пересечения осей совпадает с ФНК,
  - сдвиги, сжатия и повороты отсутствуют,
  - дополнительные нестандартные преобразования отсутствуют,
  - линии - сплошные,
  - тип интерполяции линейный (см. **INTRP(0)**),
  - высота символа 2.5 мм,
  - наклон символов  $20^{\circ}$ ,
  - угол наклона строк  $0^{\circ}$  (горизонтальны) (это не относится к строкам текста, редактируемого операторами **TITLE** и командами описания первичных осей).
- 3) задает положение левого нижнего угла чертежа на столе графопостроителя в точке с координатами ( $x, y$ ), заданными в мм.

Ограничения:

$$0 \leq x, y \leq 400$$

При нарушении этого ограничения  $x$  и  $y$  полагаются равными нулю и оператор выполняется.

## ENDGR

Оператор **ENDGR** переводит систему на уровень 1 и может выполняться с любого уровня:

Формат    52            0000            0000            0000

**TITLE** (dx, dy, s1, n1, s2, n2, s3, n3)

Оператор **TITLE** определяет размеры субчертежа (длины первичных осей), наносит на субчертеж указанные заголовки и переводит систему с уровня 1 на уровень 2.

Формат

04	dx	dy	s1
00	n1g	s2g	n2g
00	s3	n3g	0000

Здесь  $dx$ ,  $dy$  – горизонтальный и вертикальный размеры субчертежа в мм,

$s1$  – массив текстовой информации, представляющей собой заголовок субчертежа,

$n1g$  – количество символов в этом заголовке,

$s2$  – массив текста, являющегося именем горизонтальной оси с количеством символов  $n2g$ ,

$s3, n3g$  – аналогичные параметры для вертикальной оси.

Если какое-либо  $nig = 0$ , то соответствующего текста на субчертеже не будет. О подготовке символьной информации см. описание оператора **MESS**.

Заголовок субчертежа помещается над серединой верхней стороны субчертежа, причем высота символов этой строки будет = 5 мм. Имя горизонтальной оси размещается под серединой нижней части субчертежа, а имя вертикальной оси – левее середины левой стороны субчертежа, причем направление этой строки – вертикальное. Заголовки осей изображаются символами высотой 3 мм, и все заголовки рисуются на расстоянии 10 мм от поля субчертежа.

Ограничения:

- 1) уровень 1
- 2)  $x + list\ x \leq 400$ ,  $phys\ x + dx \leq list\ x$
- 3) аналогично по  $y$ .

Эти неравенства проверяют соответственно:  
помещается ли чертеж на столе размера 400 x 400 мм,  
помещаются ли оси на листе по каждой из координат.

## КОМАНДЫ ОПИСАНИЯ ПЕРВИЧНЫХ ОСЕЙ

Все операторы этого класса задают тип первичных осей и переводят систему с уровня 2 на уровень 3. Масштабное преобразование для всех последующих кривых считается определенным. На субчертеж наносятся оси с разметкой, соответствующей их описанию и типу.

**GRAPH** ( $x\ min$ ,  $x\ max$ ,  $y\ min$ ,  $y\ max$ )

**GRAPH** задает линейный масштаб по обеим осям.

Формат

10  $x\ min$   $x\ max$   $y\ min$   
00  $y\ max$  0000 0000

( $x\ min$ ,  $x\ max$ ) - диапазон изменения переменной  $x$ ,  
( $y\ min$ ,  $y\ max$ ) - аналогичные величины для переменной  $y$ .  
Преобразование будет осуществляться по формулам:

$$x = dx \times (x - x\ min) / (x\ max - x\ min)$$

$$y = dy \times (y - y\ min) / (y\ max - y\ min)$$

Здесь  $dx$ ,  $dy$  - длины первичных осей, определенные оператором **TITLE**.

Ограничения:

- 1) уровень 2
- 2)  $x\ min < x\ max$
- 3)  $y\ min < y\ max$

т.е. запрещено отрицательное направление осей.



## GRAF ( $x_{min}$ , $stx$ , $y_{min}$ , $sty$ )

Задаёт линейный тип обеих осей, масштаб отсчитывается в исходных единицах на стандартный шаг  $H$  построения оси ( $H = 20$  мм).

Формат

```
11  x min  stx  y min
00  sty    0000 0000
```

$x_{min}$  - начало отсчета по оси  $x$ ,

$stx$  - количество исходных единиц, приходящихся на стандартный шаг построения оси  $H$ ,

$y_{min}, sty$  - аналогичные величины для оси  $y$ .

Преобразование будет осуществляться по формулам:

$$x = H \times (x - x_{min}) / stx$$

$$y = H \times (y - y_{min}) / sty$$

Ограничения:

- 1) уровень 2
- 2)  $stx > 0$
- 3)  $sty > 0$

## LGLG ( $x_{min}, tx, y_{min}, ty$ )

LGLG задает логарифмический тип обеих осей.

Формат

```
12  x min  tx  y min
00  ty    0000 0000
```

$x_{min}$  - начало отсчета по оси  $x$ ,

$tx$  - длина периода по оси  $x$  в мм,

$y_{min}, ty$  - аналогичные величины для оси  $y$ .

Масштабное преобразование имеет следующий вид:

$$x = tx \times lg (x / x_{min})$$

$$y = ty \times lg (y / y_{min})$$

Ограничения:

- 1) уровень 2
- 2)  $x_{min} > 0$
- 3)  $tx > 0$
- 4)  $y_{min} > 0$
- 5)  $ty > 0$

POLAR (k, str, plx, ply)

POLAR задает полярную систему координат.

Формат

i3 k str plx

00 ply 0000 0000

k - угловой масштабный множитель (например, если углы в градусах - следует положить  $k = \pi/180$ , или 1, если углы заданы в радианах),

str - количество радиальных единиц, помещающихся в стандартном шаге построения оси H ,

plx, ply - координаты полюса, заданные в мм относительно ФНК. Если он расположен в пределах субчертежа, то оси проводятся через полюс, в противном случае - через ФНК.

Масштабное преобразование имеет следующий вид: .

$$\begin{aligned}y &= y_x (H \times \sin(k \times \alpha) / str) \\x &= y_x (H \times \cos(k \times \alpha) / str)\end{aligned}$$

Ограничения:

- 1) уровень 2
- 2)  $str > 0$  .

## СРЕДСТВА СИСТЕМЫ

### Средства системы для чертежа и субчертежа

#### PAGE (listx, listy)

Оператор задает размеры листа в мм и используется на уровне 1.

Формат

02 listx listy 0000

listx, listy - горизонтальный и вертикальный размеры листа в мм. Как уже отмечалось, все точки с координатами, не попадающими на заданный лист, системой полностью игнорируются.

Ограничения:

- 1) уровень 1
- 2)  $0 < \text{listx} \leq 400$
- 3)  $0 < \text{listy} \leq 400$

#### ORIGIN (physx, physy)

Оператор задает физическое начало координат в мм относительно нижнего левого угла чертежа и исполняется только с уровня 1.

Формат

03 physx physy 0000

physx, physy - координаты субчертежа в мм.

Ограничения:

- 1) уровень 1
- 2)  $0 \leq \text{physx} \leq 400$
- 3)  $0 \leq \text{physy} \leq 400$

## BLANK ( i , p )

Оператор **BLANK**(*i*,*p*) определяет прямоугольную область *i* , внутри которой запрещены любые построения, иными словами построения "пропадают" внутри зарезервированной области, которая остается "белой" на фоне чертежа. Оператор выполняется с уровнем 2-3.

Формат

06 *i*<sub>8</sub> p 0000 .

*i*<sub>8</sub> - имя бланка, может принимать значение от 0 до 3, т.е. одновременно могут быть зарезервированы не более 4-х областей.

p - массив из 4-х ячеек ( A , B , C , D ); резервируемый прямоугольник определяется соотношениями:

$$A \leq x \leq B, \quad C \leq y \leq D$$

A , B , C , D заданы в мм относительно ФНК.

Оператор **RSBLK**(*i*) уничтожает зарезервированную область *i* .

Ограничения:

- 1) уровень 2+3
- 2) *i*<sub>8</sub> ≤ 3

## RSBLK ( i )

Оператор **RSBLK** позволяет снимать "табу" с запрещенной оператором **BLANK** области чертежа, выполняется с уровнем 1 + 3.

Формат

07 *i*<sub>8</sub> 0000 0000

*i*<sub>8</sub> - имя бланка (см. **BLANK** ).

Ограничения:

- 1) *i*<sub>8</sub> ≤ 3

## CROSS

Исполнение оператора CROSS приводит к тому, что оси (строящиеся операторами GRAPH или GRAF ) проводятся таким образом, что их пересечение попадает в точку  $X = 0$  или  $Y = 0$  (либо  $x=y=0$ ), если только эта точка попадает в пределы субчертежа. Оператор выполняется с уровнем 1 + 3.

Формат 05 0000 0000 0000

## FRAME (i, m)

Оператор FRAME рисует прямоугольную рамку и может исполняться с уровнем 2-3.

Формат 14  $i_g$  m 0000

Параметр  $i_g$  может принимать значения от 0 до 6 и при:

$i_g = 0$  - будет изображена рамка чертежа,

$i_g = 1$  - рамка субчертежа,

$i_g = 2$  - в рамку будет заключен бланк - 0 (если он есть),

$i_g = 3$  - бланк - 1,

$i_g = 4$  - бланк - 2,

$i_g = 5$  - бланк - 3,

$i_g = 6$  - будет изображена рамка, определяемая массивом из 4-х ячеек m :

$$m[0] \leq x \leq m[1], \quad m[2] \leq y \leq m[3]$$

Ограничения:

1) уровень 2 + 3

2)  $i_g \leq 6$ .

## DUBAX (i, p1, p2)

Оператор спецификации вторичных осей DUBAX используется на уровне 3. Выполнение этой команды может приводить к изменению текущего типа оси. В отличие от команд описания первичных осей "аргументом" этого оператора является только одна ось, координатное преобразование для другой переменной остается неизменным. Если оказалось, что одна из осей "голая", а другая — линейная или логарифмическая, то в дальнейшем (при выполнении оператора CURVE) будет зафиксирована ошибка. Заметим, что введение вторичных осей не сопровождается их изображением на субчертеже, как это делалось при спецификации первичных осей.

Формат

17  $i_g$  p1 p2

Смысл параметров p1 и p2 тот же, что в соответствующем операторе описания первичных осей (см. ниже). Параметр  $i_g$  может принимать значения от 0 до 5 и при

$i_g = 0$  — (p1, p2) — определяет диапазон изменения переменной  $x$ .

Преобразование по оси  $x$  будет такое:

$$x = dx \cdot (x - p1) / (p2 - p1)$$

здесь  $dx$  — горизонтальный размер субчертежа (см. GRAPH).

$i_g = 1$  — задает линейный тип оси  $y$ , работает аналогично  $i_g = 0$ ,

$i_g = 2$  — задается линейный тип оси  $x$ ,

p1 — начало отсчета по оси,

p2 — количество исходных единиц на стандартный шаг  $H$  построения оси ( $H = 20$  мм).

Преобразование переменной  $x$  будет такое:

$$x = H \cdot (x - p1) / p2$$

(см. GRAF).

$i_g = 3$  — задается линейный тип оси  $y$ , работает аналогично  $i_g = 2$ ,

$i_8 = 4$  - задается логарифмический тип оси  $x$ ,  
 $p1$  - начало отсчета по оси,  
 $p2$  - длина периода в мм.

Преобразование по оси  $x$  будет такое:

$$x = p2 \cdot \lg(x/p1)$$

(см. LGLG).

$i_8 = 5$  - задается логарифмический тип оси  $y$ ,  
работает аналогично  $i_8 = 4$ .

Ограничения:

- 1) уровень 3
- 2)  $i_8 \leq 5$
- 3) ограничения на  $p1$  те же, что и на соответствующие параметры в операторах описания первичных осей,
- 4) для  $p2$  аналогично.

### Средства системы для кривой

## INTRP ( $i$ , $p$ )

Оператор задает один из предусмотренных в системе способов аппроксимации функций и может выполняться с уровнями 1 + 3. Действие этого оператора распространяется только на оператор CURVE.

Формат 20  $i_8$   $p$  0000

Параметр  $i_8$ , принимающий значения от 0 до 4, определяет способ интерполяции:

$i_8 = 0$  - в качестве способа интерполяции задает последовательную выборку координат из массивов  $x[0:n-1]$  и  $y[0:n-1]$  (т.е. интерполяция как таковая отсутствует), описанных в обращении к CURVE.

$i_8 = 1$  - задает способ аналогичный предыдущему, только массив  $x$  должен состоять из двух элементов:  
 $x[0]$  - начальное значение аргумента,  
 $x[1]$  - шаг его изменения,

$i_g = 2$  - в качестве способа аппроксимации задает алгоритм кусочно-полиномиального субтабулирования однозначной функции  $(x[0:n-1], y[0:n-1])$  интерполяционными многочленами Лагранжа степени  $\min\{p, n-1\}$ , где  $i \leq p \leq 5$ . При этом частота дополнительных строящихся точек будет составлять  $\sim 1.6$  мм вдоль дуги кривой,

$i_g = 3$  - задает способ интерполяции аналогично способу при  $i_g = 2$ , только заданная функция описывается таблицей с постоянным шагом (как при  $i_g = 1$ ),

$i_g = 4$  - аналогично  $i_g = 2$ , но уже не предполагается однозначность функции, поскольку будет использован параметрический алгоритм интерполяции.

Ограничения:

- 1)  $i_g \leq 5$
- 2)  $0 \leq p \leq 4$

## MARKR (i)

Система AMES PLOT отличает образы исходных точек кривой (назовем их опорными) от других точек, дополнительно построенных ею (в результате субтабулирования). BCURVE задается параметр  $h$  (шаг маркировки), означающий, что каждая  $h$ -ая опорная точка, начиная с первой, будет помечена стандартным маркером (если  $h = 0$ , то никакие точки данной кривой маркироваться не будут), конкретный тип которого задается оператором MARKR. Оператор выполняется с уровнем 1 + 3.

Формат      24       $i_g$       0000      0000

Ниже приводится набор стандартных маркеров:

$i_g = 0$	1	2	3	4	5	6	7
□	◇	⊗	⊗	▽	▷		=

Размер маркера  $\sim 2$  мм x 2 мм, причем центр его совпадает с помечаемой точкой. Форма маркера не зависит от преобразования координат.



Ограничения:

$$1) i_g \leq 7$$

**STLIN** ( $i$ ,  $i_{pen}$ )

Оператор **STLIN** задает режим прерывания линии для оператора **CURVE** и выполняется с уровнем 1 + 3.

Формат 21  $i_g$   $i_{pen}$  0000

Параметр  $i_g$  определяет тип линии:

$i_g=0$  - линия сплошная,

$i_g=1$  - линия не изображается (т.е. будут наноситься на чертеж только маркеры, если в операторе **CURVE** параметр  $h \neq 0$ ),

$i_g=2$  - линия пунктирная (штрих - 1 мм, пробел - 2 мм),

$i_g=3$  - " " (штрих - 2 мм, пробел - 3 мм),

$i_g=4$  - " " (штрих - 5 мм, пробел - 2 мм),

$i_g=5$  - линия штрихпунктирная (штрих - 10 мм, пробел - 2 мм, штрих - 1 мм, пробел - 2 мм),

$i_g=6$  - линия штрихпунктирная (20 мм, 2 мм, 1 мм, 2 мм),

$i_g=7$  - в этом случае перед перемещением пера в очередную точку значение признака пера выбирается из ячейки  $i_{pen}$ , т.о. управление пером передается в руки программиста (напомним, что если 36 разряд  $i_{pen}=0$ , то перо опущено).

Осуществлять управление пером можно, используя оператор **btzan**.

Ограничения:

$$1) i_g \leq 7$$

**MRSCD** ( $l$ ,  $g$ )

Оператор **MRSCD** задает режим прерывания линии и выполняется с уровнем 1 + 3. Действие оператора распространяется на операторы построения координатных сеток и вывода кривой.

Формат 22  $l$   $q$  0000

Процесс прерывания линии, задаваемый оператором MRSCD, является периодичным и массив  $l$  описывает один период, состоящий из  $q$  элементов. Нечетный элемент массива соответствует длине штриха, а четный — длине пробела (в мм).

Ограничения:

$$1) 0 \leq l[i] \leq 63.5$$

$$2) 2 \leq q \leq 6$$

SCALE ( $m_x, m_y$ )

Оператор SCALE определяет дополнительное масштабное преобразование кривой вида:

$$x = m_x \cdot x$$

$$y = m_y \cdot y$$

и выполняется с уровнем 1 + 3.

Формат 25  $m_x$   $m_y$  0000

SHIFT ( $dx, dy$ )

Оператор SHIFT задает сдвиг кривой относительно ФНК:

$$x = x + dx$$

$$y = y + dy$$

и выполняется с уровнем 1 + 3.

Формат 26  $dx$   $dy$  0000

$dx$  — смещение по оси  $x$  (в мм относительно ФНК),

$dy$  — тоже по оси  $y$ .

## ANGLE ( $\varphi, x, y$ )

Оператор ANGLE определяет угол поворота кривой вокруг заданной точки и выполняется с уровнем 1 + 3.

Формат        27  $\varphi$   $x$   $y$

$\varphi$  - угол поворота, заданный в градусах (положительное направление - против часовой стрелки),  
 $x, y$  - координаты полюса поворота в мм относительно ФНК.

## ATRAN ( $z, a, b$ ), BTRAN ( $z, a, b$ )

Для того, чтобы программист имел возможность осуществлять нестандартные преобразования кривых, необходимы средства доступа к текущей точке график: . процессе его построения. Это достигается с помощью операторов ATRAN или BTRAN, которые работают с уровнем 2 + 3.

Формат ATRAN  
30  $z$   $a$   $b$

Формат BTRAN  
31  $z$   $a$   $b$

Параметры  $a$ ,  $b$  определяют  $n/n$  пользователя, которая осуществляет необходимые преобразования координат:

- $a$  - именуется точкой входа в  $n/n$ ,
- $b$  - точку выхода (в ячейку, помеченную меткой  $b$ , система посылает команду возврата),
- $z$  - массив из двух элементов, через который происходит обмен координатами СНК между системой и  $n/n$  пользователя:

$z[0] - x$  ,     $z[1] - y$  .

Если уход на произвольное преобразование задан оператором **BTRAN**, то при последующем выполнении операторов **CURVE** система будет обращаться к  $n/n$  пользователя, выдавая ему координаты выводимых точек (СНК), с шагом  $\sim 1,3$  мм вдоль кривой (т.е. дополнительно производится линейная интерполяция). В случае же задания преобразования оператором **ATRAN** дополнительной интерполяции производиться не будет и в  $n/n$  будут выдаваться лишь опорные точки и точки, вычисленные системой в результате работы алгоритмов интерполяции и прерывания линии (если эти алгоритмы включены в работу системы соответствующими операторами).

При написании  $n/n$  преобразования следует иметь в виду следующее:

1. Из  $n/n$  нельзя вызывать систему, поскольку в ней не предусмотрен режим рекурсивной работы,
2. Запрещается пользоваться буферным накопителем (т.е. нельзя печатать, перфорировать и т.д.),
3. При входе в  $n/n$  значение PA неопределенно.

Действие операторов **ATRAN** или **BTRAN** отменяется оператором **RSTRN**.

## RSTRN

**RSTRN** отменяет действия операторов **ATRAN** или **BTRAN** и выполняется с уровней 1 + 3.

Формат      32            0000            0000            0000

**CURVE** (x, y, n, i, j, h)

Оператор **CURVE** строит график функции, заданной таблично, и выполняется с уровня 3.

Формат      33      x      y      n  
                  00    0i,00    0j,00    h

Оператор **CURVE** всегда записывается в двух ячейках.

$x$  - массив аргументов функции (см. **INTRP**),

$y$  - массив значений функции,

$n$  - длина массива  $y$ ,

$i_0$  - номер куба МОЗУ, содержащего массив  $x$ ,

$j_0$  - номер куба МОЗУ, содержащего массив  $y$ ,  
(параметры  $i_0$  и  $j_0$  задаются одной восьмеричной цифрой),

$h$  - шаг маркировки (см. **MARKR**).

Точки выводимой кривой подвергаются преобразованиям в следующем порядке:

где  $\bar{x}' = \text{abtzan}(\text{angle}(\text{shift}(\text{mm}(\text{scale}(\text{axs}(\bar{x}))))))$ ,

$\text{axs}$  - преобразование, определяемое типом осей  
(логарифмирование или переход из полярной системы),

$\text{scale}$  - масштабирование,

$\text{mm}$  - перевод условных единиц в мм относительно ФНК,

$\text{shift}$  - сдвиг,

$\text{angle}$  - поворот,

$\text{abtzan}$  - произвольное преобразование.

Ограничения:

- 1) уровень 3
- 2)  $1 \leq n \leq 4095$
- 3)  $0 \leq h \leq 4095$

### Средства для вывода текста

## HEIGHT ( $h$ )

Символы текста рисуются в некотором квадрате размером  $h \times h$  мм<sup>2</sup>. Оператор **HEIGHT** позволяет устанавливать нужное значение  $h$ .

Формат      36     $h$       0000      0000

$h$  - высота символа в мм. Выполняется оператор с  
уровней 1 ÷ 3.

## ANGLS (4)

Оператор **ANGLS** задает угол направления строк текста в градусах, отсчитываемый от направления "горизонтально вправо" против хода часовой стрелки (положительное направление) и выполняется с уровней 1 + 3.

Формат    37    φ            0000            0000

## MESS (s, n, x, y)

Оператор **MESS** выводит строку символов и выполняется с уровней 2 + 3.

Формат    40    s                    n<sub>s</sub>                    x  
          00    y                    0000                0000

s - массив символов (выводимая строка) в коде УПП, упакованных в ячейке слева направо, начиная с s[0], т.о. для n символов требуется

$$k = \text{ent} \left( \frac{n-1}{6} \right) + 1$$

ячеек (ячейка s[k-1] может быть не полной),

n<sub>s</sub> - число символов в массиве (включая управляющие, см. ниже),

x, y - координаты левого нижнего угла первого символа строки (см. **MEIGH**), заданные в мм относительно ФНК.

Допустимыми символами являются:

1. Арабские цифры от 0 до 9,
2. Буквы русского алфавита (исключая ъ), буквы латинского алфавита, несовпадающие по написанию с русскими,
3. Символы: + - / , . (пробел). Недопустимые символы заменяются пробелами без сообщения об ошибке. Управляющими символами являются круглые скобки ( , ).

Если в тексте встретится символ (, то высота последующих выводимых символов увеличится вдвое, символ ) уменьшает высоту выводимых символов (тоже вдвое). Управляющие символы не изображаются, их число в строке произвольно.

Следует иметь в виду , что изменение  $n$  с помощью управляющих символов приведет к тем же последствиям, что и выполнение соответствующих операторов **HEIGHT** .

Ограничения:

1) уровень  $\geq 2$ .

**REALN** ( $n$ ,  $x$ ,  $y$ )

Оператор **REALN** изображает значение числа  $n$  (шесть значащих цифр в формате с плавающей запятой), причем нижний левый угол этой строки помещается в точку ( $x$ ,  $y$ ), заданную в мм относительно ФНК. Выполняется с уровнем  $2 + 3$ .

Формат      41     $n$      $x$      $y$

$n$  - выводимое число, представленное в машинном виде (нормализованное).

Например, если содержимое  $n = 1.23456789$  то будет изображено:      . 1 23457E1 .

Ограничение:

1) уровень  $\geq 2$ .

СИМТАКСИС AMESPLOT - ОПЕРАТОРОВ

< angle > :: = 'angle' , < φ > , < x > , < y >  
    < φ > , < x > , < y > :: = < скаляр >  
< angle > :: = 'angle' , < φ >  
    < φ > :: = < скаляр >  
< atran > :: = 'atran' , < r > , < a > , < b >  
    < r > :: = < массив >  
    < a > , < b > <sup>1)</sup> :: = < метка >  
< begpr > :: = 'begpr' , < x > , < y >  
    < x > , < y > :: = < скаляр >  
< blank > :: = 'blank' , < i > , < p >  
    < i > :: = < число >  
    < p > :: = < массив >  
< btran > :: = 'btran' , < r > , < a > , < b >  
    < r > :: = < массив >  
    < a > , < b > <sup>1)</sup> :: = < метка >  
< cross > :: = 'cross'  
< curve > :: = 'curve' , < x > , < y > , < n > , < h >  
    < x > , < y > :: = < массив >  
    < n > , < h > :: = < скаляр >  
< dubax > :: = 'dubax' , < i > , < p1 > , < p2 >  
    < i > :: = < число >  
    < p1 > , < p2 > :: = < скаляр >  
< endgr > :: = 'endgr'

<sup>1)</sup> Метка < b > должна помечать оператор cod('return')



< frame > :: = ' frame ' , < i > , < m >  
 < i > :: = < ЧИСЛО >  
 < m > :: = < МАССИВ >  
 < graf > :: = ' graf ' , < xmin > , < stx > , < ymin > , < sty >  
 < xmin > , < stx > , < ymin > , < sty > :: = < скаляр >  
 < graph > :: = ' graph ' , < xmin > , < xmax > , < ymin > , < ymax >  
 < xmin > , < xmax > , < ymin > , < ymax > :: = < скаляр >  
 < heigh > :: = ' heigh ' , < h >  
 < h > :: = < скаляр >  
 < intrp > :: = ' intrp ' , < i > , < p >  
 < i > :: = < ЧИСЛО >  
 < p > :: = < скаляр >  
 < lglg > :: = ' lglg ' , < xmin > , < tx > , < ymin > , < ty >  
 < xmin > , < tx > , < ymin > , < ty > :: = < скаляр >  
 < markr > :: = ' markr ' , < i >  
 < i > :: = < ЧИСЛО >  
 < mess > :: = ' mess ' , < v > , < n > , < x > , < y >  
 < v > :: = < МАССИВ >  
 < n > :: = < ЧИСЛО >  
 < x > , < y > :: = < скаляр >  
 < mrgcd > :: = ' mrgcd ' , < l > , < q >  
 < l > :: = < МАССИВ >  
 < q > :: = < скаляр >  
 < origin > :: = ' origin ' , < physx > , < physy >  
 < physx > , < physy > :: = < скаляр >

```

< page > ::= 'page' , < listx > , < listy >
    < listx > , < listy > ::= < скаляр >
< polar > ::= 'polar' , < k > , < str > , < plx > , < ply >
    < k > , < str > , < plx > , < ply > ::= < скаляр >
< realn > ::= 'realn' , < n > , < x > , < y >
    < n > , < x > , < y > ::= < скаляр >
< rsblk > ::= 'rsblk' , < i >
    < i > ::= < число >
< rstrn > ::= 'rstrn'
< scale > ::= 'scale' , < mx > , < my >
    < mx > , < my > ::= < скаляр >
< shift > ::= 'shift' , < dx > , < dy >
    < dx > , < dy > ::= < скаляр >
< stlin > ::= 'stlin' , < i > , < ipen >
    < i > ::= < число >
    < ipen > ::= < ид-тор простой переменной >
< title > ::= 'title' , < dx > , < dy > , < s1 > , < n1 > , < s2 > , < n2 > ,
    < s3 > , < n3 >
    < dx > , < dy > ::= < скаляр >
    < s1 > , < s2 > , < s3 > ::= < массив >
    < n1 > , < n2 > , < n3 > ::= < число >

```

**Примечание;**

Последние параметры операторов можно опустить (вместе с разделяющими их запятыми), если их значения безразлично или нулевое.

Ниже будут даны длины программных модулей, реализующих некоторые AMESPLOT -операторы. При расчете размеров рабочего поля ИС следует исходить из того, что на РП должен помещаться самый большой из используемых Вами операторов с учетом памяти, занимаемой строками TX<sup>1</sup> , а также теми СП, которые Вы возможно используете из  $n/n$  произвольного преобразования ( *vtzan* или *atzan* ).

(В целях сокращения таблицы ниже приводятся данные только о тех операторах, длина которых превышает минимальное допустимое РП = 300g).

Оператор: Длина : Длина : Т о ч н а я    д л и н а

	MAX	MIN	
TITLE	I025 <sup>4)</sup>	266	266 + <u>if</u> заголовки есть <u>then</u> MESS <sup>3)</sup> <u>else</u> 0
GRAPH	I247	I223	510 + MESS
GRAF	I223	II77	502 + MESS
POLAR	I24I	I2I5	464 + MESS
LGLG	I274	I250	535 + MESS
CURVE	IIIO	460	402 + ( <u>if</u> интерполяция линейная <u>then</u> 56 <u>else</u> I76) + ( <u>if</u> параметр $h \neq 0$ <u>then</u> 51 <u>else</u> 0) + ( <u>if</u> режим линии <sup>4)</sup> произв. преобр. <sup>5)</sup> <u>then</u> 42 <u>else</u> 0) + ( <u>if</u> режим линии <u>then</u> 10; <u>else</u> 0) + <u>if</u> <u>stran</u> использован <u>then</u> 45 <u>else</u> 0
MESS	537	5I3	5I3 + <u>if</u> BLANK использован <u>then</u> 24 <u>else</u> 0
FRAME	422	376	376 + <u>if</u> BLANK использован <u>then</u> 24 <u>else</u> 0
REALN	I07I	I045	332 + MESS

1. Число строк TX , относящихся к системным программам не превышает 100.
2. Все числа даны в восьмеричной системе счисления
3. MESS обозначает длину оператора MESS
4. Здесь имеются в виду режимы прерывания линии, отличные от режимов, определяемых оператором SYLIN ci-1,7
5. Преобразование, задаваемое операторами ATRAN или STRAN.