

58400  
Г-15

23/10-7

СООБЩЕНИЯ  
ОБЪЕДИНЕННОГО  
ИНСТИТУТА  
ЯДЕРНЫХ  
ИССЛЕДОВАНИЙ  
ДУБНА



2721/2-73

10 - 7195

В.В. Галактионов

АВТОКОД SLANG 3 ДЛЯ ТРА  
(Дополнения)

**1973**

ЛАБОРАТОРИЯ ВЫЧИСЛИТЕЛЬНОЙ  
ТЕХНИКИ И АВТОМАТИЗАЦИИ

10 - 7195

**В.В. Галактионов**

**АВТОКОД SLANG 3 ДЛЯ ТРА**  
**(Дополнения)**

## I. Основные символы

Буквы:

A, B, C, D, E, F, G, H, I, J, K,  
L, M, N, O, P, Q, R, S, T, U, V, X, Y, Z  
Б, Г, Д, Ж, З, И, Л, П, Ф, Ц, Ч, Ш, Щ, Ъ,  
Ы, Э, Ю, Я.

## 2. Псевдокоманды

В SLANG3 вводится понятие псевдокоманд, которые по своей сущности являются командами для самого транслятора и в ряде случаев не переходят в двоичную программу. В автокоде имеются две псевдокоманды для описания переменных COMMON и ARRAY.

### 2.1. Описание переменных.

Псевдокоманда ARRAY резервирует указанное количество ячеек памяти, а в таблицу идентификаторов заносятся соответственно символические обозначения этих ячеек.

Псевдокоманда ARRAY A,B,C

эквивалентна группе помеченных констант:

A, 0

B, 0

C, 0

### 2.2. Общие переменные.

Для независимых программ, образующих единый пакет, транслируемый за один вызов транслятора (см. пункт 8), вводится понятие "общие переменные", описываемые псевдокомандой COMMON.

Переменные A, B, C, описанные псевдокомандой COMMON в разных программах, имеют одинаковый смысл, а именно: сохраняют первоначально установленные в какой-либо программе числовые значения.

Если к моменту описания в COMMON какая-либо из переменных не была определена, ей присваивается значение адреса текущей ячейки программы и резервируется одна ячейка памяти.

Пример:

```
      *I000
      ALPHA = I00
      BETA  = 200
      ARRAY A,B,C
      COMMON ALPHA, BETA, C, DEL
```

Псевдокоманда ARRAY A,B,C резервирует три ячейки памяти I000, I001, I002 с символическими названиями A, B, C.

Псевдокоманда COMMON установит признаки общих ячеек для переменных ALPHA, BETA, C и для новой зарезервированной ячейки с именем DEL.

Если в какой-нибудь программе встретится псевдокоманда COMMON с упоминанием какой-либо из переменных ALPHA, BETA, C или DEL, то эта переменная сохранит свое числовое значение, установленное в указанном выше кусочке программы, т.е. I00, 200, I002, I003 соответственно для ALPHA, BETA, C и DEL.

### 3. Макрокоманды

В SLANG3 существуют две макрокоманды, которые транслируются в некоторую последовательность команд в двоичной программе.

#### 3.1. Обращение к подпрограмме CALL.

Макрокоманда обращения к подпрограмме с заданием параметров

```
      CALL SUBP A,B,C
```

эквивалентна следующей группе команд:

JMS SUBP

A

B

C

Параметры A, B, C – определяемые каким-либо образом в программе идентификаторы либо метки. Они могут отсутствовать, но число их не должно превышать 5.

### 3.2. Заголовок подпрограммы NAME.

Макрокоманда SUBP, NAME X, Y, Z транслируется в группу команд, потребляющих параметры, заданные макрокомандой обращения к подпрограмме CALL SUBP A, B, C:

```
SUBP, 0
CLA
TAD I SUBP
DCA X
ISZ SUBP
TAD I SUBP
DCA Y
ISZ SUBP
TAD I SUBP
DCA Z
ISZ SUBP
```

В этом случае в ячейках с метками X, Y, Z будут размещены числовые значения идентификаторов A, B, C.

Выход из подпрограммы осуществляется обычным образом – командой JMR I SUBP

### 4. Эквивалентность.

Эквивалентность позволяет приписывать идентификаторам некоторые числовые значения.

Общий вид эквивалентности:

$$\langle \text{идентификатор} \rangle = A \pm B, \text{ где}$$

A и B — определенные к моменту трансляции такой конструкции идентификаторы либо восьмеричные числа.

Примеры:

$$\begin{aligned} \text{ALPHA} &= 120 \\ \text{BETA} &= \text{ALPHA} - 20 \\ \text{GAMMA} &= \text{ALPHA} - \text{BETA} \\ \text{DELTA} &= 1000 - \text{GAMMA} \end{aligned}$$

## 5. Команды.

### 5.1. Адрес типа "литерал".

Адресом типа "литерал" называется следующая конструкция в адресной части команды:

$$" \langle \text{восьмеричное число} \rangle "$$

Знак "-" является признаком адреса типа "литерал". При появлении указанной конструкции в адресе команды транслятор формирует определяемую ею восьмеричную константу и записывает ее в одну из ячеек буфера ссылок, указанного псевдокомандой `BASE#`, и заменяет исходный адрес типа "литерал" адресом этой ячейки.

### 5.2. Полный адрес имеет структуру:

$$\pm A \pm B, \text{ где } A \text{ и } B - \text{метки, либо определенные каким-либо другим образом идентификаторы, либо восьмеричные числа.}$$

Примеры:

$$\begin{aligned} \text{JMS RET} &+11 \\ \text{TAD} &- \text{ALPHA} + \text{BETA} \end{aligned}$$

## 6. Микропрограммирование.

В ZLMOZ допускается микропрограммирование только двух операций.

Примеры: CLR CBL  
SZA CLA  
RAR CLL

## 7. Диагностика транслятора

Транслятор в автокоде обнаруживает некоторые формальные несоответствия синтаксису языка, ошибки в структуре операторов и констант и печатает диагностические тексты. В зависимости от вида ошибки игнорируется либо вся команда или константа, либо некоторые их части. В случае обнаружения ошибки при трансляции, выдача двоичной программы на перфоленгу не производится, и в конце листинга программы печатается сообщение о количестве обнаруженных ошибок.

Диагностические тексты:

1) Неописанный идентификатор xxxxxx

2) Дважды описанный идентификатор xxxxxx

3) Неправильная константа xxxxxx

Диагностика выдается, если xxxxxx не является восьмеричной константой.

4) xxxxxx Переполнена таблица идентификаторов.

Число идентификаторов больше, чем 250.

5) Несоответствие страниц.

Диагностика выдается при появлении в адресе команды ссылки на адрес не из текущей страницы.

6) Неправильный идентификатор xxxxxxxxxxxx

Указанный идентификатор содержит более 6 символов.

- 7) Неправильная конструкция холерической константы.  
При обнаружении недопустимых символов в текстовой константе.

#### 8. Подготовка программы для трансляции.

Транслятор может обрабатывать пакет, состоящий из нескольких независимых программ.

Понятие программы: последовательность операторов автокода, заканчивающаяся знаком конца - "\$".

Формирование пакета:

```
*NAME _ SLANG
*ASSIGN _ LTAPE
*PERSONAL LIBRARY
*NO LOAD LIST
*MAIN SLANG
*EXECUTE
┌───┐
├───┤ 1-ая программа
├───┤ $
├───┤ 2-ая программа
├───┤ $
├───┤ .
├───┤ .
├───┤ .
├───┤
├───┤ N-ая программа
├───┤ $
└───┘
*END FILE
```

#### 9. Управляющие карты

Пользователь может устанавливать или переопределять режим трансляции каждой программы или всех программ пакета с помощью управляющих карт:

- 1) \*NAME# - карта идентификации программы. 18 символов, следующие после \* NAME# печатаются на АЦПУ большими буквами. Ставится перед программой.



- 2) \* PUNCH - выдача на перфоленту двоичной программы.
- 3) \* NO PUNCH - отключение перфорации двоичной программы.
- 4) \* NO LIST - отключение выдачи листинга программы на АЦПУ.
- 5) \* TEXT - выдача на перфоленту текстовой программы в коде ASCII.
- 6) \* NO TEXT - отключение перфорации текста программы.
- 7) \* STANDARD - установление стандартного режима трансляции с выдачей листинга программы и перфорации двоичной программы при отсутствии ошибок трансляции.

При отсутствии управляющих карт, устанавливающих режим трансляции, автоматически выбирается стандартный режим трансляции.

#### ЛИТЕРАТУРА

Г. В. В. Галактионов.

Сообщение ОИЯИ, IO-59II, Дубна, 1971 г.

Рукопись поступила в издательский отдел  
25 мая 1973 года.