

П-58

ОБЪЕДИНЕННЫЙ
ИНСТИТУТ
ЯДЕРНЫХ
ИССЛЕДОВАНИЙ

ДУБНА



21/11-77

4590/2-77

11 - 10813

М.Ю.Попов, Е.Д.Федюнькин

МОДЕРНИЗАЦИЯ АВТОКОДА MADLEN

(методы оптимизации транслятора)

1977

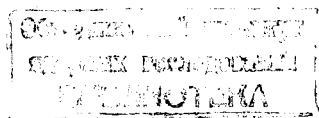
11 - 10813

М.Ю.Попов, Е.Д.Федюнькин

МОДЕЛИЗАЦИЯ АВТОКОДА MADLEN

(методы оптимизации транслятора)

Направлено на совещание по программированию и математическим методам решения физических задач, Дубна, 1977.



Попов М.Ю., Федюнькин Е.Д.

11 - 10813

Модернизация автокода MADLEN (методы
оптимизации транслятора)

Создана очередная версия автокода MADLEN для БЭСМ-6.
Обсуждаются методы оптимизации алгоритмов, используемых в
автокодных трансляторах. Описаны новые возможности языка.

Работа выполнена в Лаборатории вычислительной техники
и автоматизации ОИЯИ.

Препринт Объединенного института ядерных исследований. Дубна 1977

© 1977 Объединенный институт ядерных исследований Дубна

1. Особенности новой версии автокода.

Новая версия автокода MADLEN^{/1-3/} вызвана к жизни проблемами, возникшими при дальнейшем развитии операционной системы "Дубна". Широкий спектр конфигураций существующих ныне вычислительных комплексов, базирующихся на машине БЭСМ-6, требует от операционной системы способности к автонастройке на конкретную конфигурацию. Эта способность может быть реализована при наличии в языке программирования аппарата условной трансляции.

В автокод MADLEN-2 встроен бесстекový аппарат условной трансляции, допускающий произвольную глубину вложения операторов условной трансляции. При посредстве этого аппарата автокодная программа может управлять работой транслятора примерно так же, как программа в машинном коде управляет работой машины. Реализовано три типа условной трансляции: по сравнению двух полных адресов, по спецификациям и по атрибутам. В основном мы следовали синтаксическим принципам, разработанным фирмой CDC для автокода COMPASS^{/4/}.

Созданы средства генерации и использования системных текстов. Системный текст - перемещаемая программа, содержащая информацию об идентификаторах (системные символы) и макросах (системные макросы). Имя системного текста является одним из параметров, сообщаемых транслятору в момент вызова:

*ASSEMBLER (P1, P2, STEXT = ИМЯ, P4, ...),

после чего эти идентификаторы и макросы оказываются заранее определенными в каждой автокодной программе. Системный текст, используемый при трансляции диспетчера, содержит в себе информацию о конфигурации вычислительного комплекса.

Появились идентификаторы нового типа – переменные. Значение этих идентификаторов может быть переопределено в любом месте подпрограммы.

Расширились возможности для представления констант, главным образом путем введения суффиксов-модификаторов различного вида. Теперь допускается использование холеритов в адресных выражениях.

Расширены синтаксические рамки инструкций BLOCK, LOG, ISO, GOST, TEXT. Появился ряд новых инструкций, облегчающих программирование и создающих новые возможности.

Введены некоторые типы макросов (генерация полного текстового макроса будет реализована в следующей версии автокода).

Для целей автодокументации создан аппарат динамического управления форматом листинга, снабженный магазинным стеком на 15 позиций.

2. Некоторые соображения об оптимизации.

Ниже изложены некоторые принципы оптимизации, использованные при создании транслятора MADLEN-2. На наш взгляд, эти принципы следует учитывать при программировании трансляторов или аналогичных больших систем (особенно, для машины БЭСМ-6).

1. Самое быстрое сканирование входного языка реализуется с помощью многоуровневого динамического переключателя. На первом уровне все символы языка сортируются на имеющие и не имеющие синтаксического значения. Пусть, например, очередной символ – на 14 регистре. Вход в динамический переключатель:

14. UJ, LEVEL1

Таблица LEVEL1 имеет столько же входов, сколько символов в языке. В каждом слове левая команда ис-

пользуется для попутных семантических операций (у нас – для перекодировки: на сумматор читается соответствующий символ в коде TEXT). Сортировка осуществляется правой командой. Для символов, не имеющих синтаксического значения, она имеет вид:

UJ, НАКОПИТЕЛЬ СИМВОЛОВ

Для символов, имеющих синтаксическое значение, она имеет вид:

10, UJ, N

На 10 регистре – базовый адрес сортирующей таблицы второго уровня, число N – номер слова в таблице. Таблиц второго уровня – несколько. Контекстная зависимость в процессе сканирования реализуется попросту изменением базового адреса на 10 регистре. Таблица второго уровня устроена аналогично таблице LEVEL1. Из таблицы второго уровня происходит рассылка на конкретные процессоры или дополнительные переключатели третьего уровня.

Заметим специально, что в случае языков с сильной контекстной зависимостью такого рода переключатель позволяет заметно сэкономить память, поскольку нужное изменение программы сканирования достигается единственной командой засылки базового адреса на индекс-регистр.

2. Поиск в постоянной таблице (например, таблице кодов операций) всегда следует реализовать в виде мультипликативного HASH-процесса с ассоциативной адресацией омонимов. Дело в том, что для постоянной таблицы HASH-константу всегда можно вычислить на ЭВМ так, чтобы базисная таблица использовалась оптимально, а длина омонимических цепочек была минимальной.

3. Любые таблицы, для поиска которых используется HASH-процесс, следует организовывать с ассоциативной адресацией омонимов. В этом случае экономится память, поскольку возникает единое поле расширений для объектов разного рода. Кроме того, отпадает необ-

ходимость завышать размеры переменных таблиц и исчезает понятие максимального числа объектов.

4. Следует активно использовать маскообразующие свойства операции циклического сложения. В некоторых случаях при этом удается получить выигрыш во времени в десятки раз. Пример 1. Непустой сумматор содержит неизвестное число символов в коде ISO, нормализованных влево. Остаток сумматора заполнен кодом 0B. Требуется заполнить остаток сумматора кодом пробелов. Реализация:

```
15, ATX,  
  , ARX, = 7777777777777776  
15, AOX, -1  
15, AEX, -1  
  , AAX, = 6H  
15, AEX,
```

Пример 2. Требуется обнулить содержимое сумматора, если 41 бит не нуль:

```
15, ATX,  
  , AAX, = :002  
15, ATX,  
  , ARX, = 7777777777777777  
15, AEX,  
15, AAX,
```

5. Целое деление восьмеричного числа без порядка на фиксированную константу можно организовать посредством целого умножения на другую константу. Пусть делимое - на сумматоре. Деление на 10:

```
, NTR, 3  
  , A * X, = 4001463146314632 .
```

Деление на 6:

```
, NTR, 3  
  , A * X, = 4002525252525253 .
```

Подобного рода операции широко используются программистами фирмы CDC, в СССР их применяли Веретеннов и Гуревич^{15/}.

6. Если вновь написанная система программ начала работать с первого запуска, в системе с необходимостью есть ошибка.

Транслятор MADLEN-2 находится в опытной эксплуатации.

Литература

1. Волков А.И. ОИЯИ, Б4-11-4654, Дубна, 1969.
2. Волков А.И. ОИЯИ, 11-5427, Дубна, 1970.
3. Волков А.И. ОИЯИ, 11-5426, Дубна, 1970.
4. COMPASS version 3 reference manual. Pub.60492600, Control Data Corporation. Sunnyvale, California, 1976.
5. Веретеннов В.Ю. и др. ИАЭ-2486, Москва, 1975.

Рукопись поступила в издательский отдел
5 июля 1977 года.