

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
им. М.В.ЛОМОНОСОВА
Факультет вычислительной математики и кибернетики

Ф - 758

11-81-759

ФОЛТЕНИ
Вилмош

На правах рукописи

РЕАЛИЗАЦИЯ ПРОМЕЖУТОЧНОГО ЯЗЫКА
ВЫСОКОГО УРОВНЯ АЛМО НА МИНИ-ЭВМ ЕС-1010

Специальность - 01.01.10 - математическое обеспечение
вычислительных машин и систем

Автореферат диссертации на соискание ученой степени
кандидата физико-математических наук

Москва 1981

Работа выполнена на кафедре системного программирования
факультета вычислительной математики и кибернетики МГУ.

Научный руководитель:
доктор физ.-мат. наук профессор Э.З. ЛЮБИМСКИЙ.

Официальные оппоненты:
доктор физ.-мат. наук профессор Г.Д. ФРОЛОВ,

кандидат физ.-мат. наук
ст. научный сотрудник Н.А. КОНОВАЛОВ.

Ведущая организация – Научно-исследовательский центр
электронной вычислительной техники (НИЦЭВТ), г. Москва.

Защита состоится " _____ " _____ 1982 г. в 15 часов 30 мин.
на заседании Ученого совета Д053.05.38 при Московском государ-
ственном университете по адресу: Москва, П7234, Ленинские го-
ры, МГУ, факультет вычислительной математики и кибернетики,
аудитория...

С диссертацией можно ознакомиться в библиотеке факультета.

Автореферат разослан " _____ " _____ 1982 г.

Ученый секретарь Совета
доцент Н.П. ТРИФОНОВ

I. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

I. Актуальность темы

В последние годы развитие вычислительной техники определя-
лось повсеместным распространением мини- и микромашин. Одной из
наиболее распространенных в странах СЭВ является минимашинна
ЕС-1010.

ЕС-1010 – самая маленькая машинна серии ЕС – производится в
Венгрии. Опишем кратко наиболее важные характеристики аппаратно-
го обеспечения этой машинны. Ее максимальная оперативная память
– 64 Кбайта. К машинне могут быть присоединены два мини-диска,
каждый из которых обладает максимальной емкостью в 800 Кбайтов,
и 4 магнитных ленты. Медленными внешними устройствами могут быть:
АЦПУ, устройство, считывающее и (или) перфорирующее бумажную
ленту, устройство, считывающее карты. Машинна микропрограммирова-
на и обладает 32 уровнями прерывания аппаратных средств, к
которым могут присоединяться дальнейшие внешние устройства ре-
ального масштаба времени. Особенно важно отметить, что ЕС-1010
аппаратно не совместима со старшими моделями ЕС.

Необходимость разработки транслятора с языка АЛМО для ма-
шинны ЕС-1010 вызвана следующими причинами:

- а) для старших членов семейства ЕС существуют трансляторы с
АЛМО, а т.к. ЕС-1010 отличается от них с точки зрения аппаратных
средств, то необходимо обеспечить совместимость между ЕС-1010
и старшими моделями;
- б) из трансляторов с языков высокого уровня ЕС-1010 обладает
только трансляторами с языков ФОРТРАН и КОБОЛ. Кажется целесо-
образным включить в математическое обеспечение ЕС-1010 трансля-

ОБЪЕДИНЕННЫЙ ИНСТИТУТ
ЭЛЕКТРОННЫХ ИССЛЕДОВАНИЙ
БИБЛИОТЕКА

торы, переводящие и с других языков высокого уровня, как, например, АЛГАМС или Комплекс АЛГОЛ. В ходе трансляции с вышеупомянутых языков высокого уровня в промежуточной программе на языке АЛМО появляются массивы больших размеров; из-за малогабаритности машины и ограниченности в адресации отображение массивов и структуры перекрытия модулей должно быть выполнено с особой тщательностью;

в) ЕС-1010 не имеет такого языка, на уровне которого могло бы быть осуществлено перекрытие произвольной структуры;

г) математическая библиотека ЕС-1010 отличается бедностью.

2. Цель работы – создание транслятора с языка АЛМО для машины ЕС-1010, отвечающего современным требованиям; разработка методов реализации промежуточного языка высокого уровня на мини-ЭВМ с таким расчетом, чтобы как можно большую часть уже готовой работы можно было использовать при создании новых трансляторов.

3. Методика работы

При выборе метода мы руководствовались следующими аспектами:

- транслятор должен легко модифицироваться, так как определение языка АЛМО постоянно развивается;
- транслятор должен работать эффективно на малых ЭВМ, какой является ЕС-1010.

Учитывая эти аспекты, мы изучили разные методы трансляции и пришли к выводу, что перечисленным требованиям может соответствовать синтаксически управляемый транслятор, в котором используется детерминированный способ анализа. При этом имеется возможность производить семантическую обработку вызовом семантических подпрограмм под управлением анализатора. Детерминирован-

ность анализа обеспечивает быстроту трансляции. Из детерминированных способов выбран особенно быстрый – нисходящий метод. Правда, для этого потребовалось существенное преобразование исходного синтаксиса языка АЛМО.

4. Научная новизна и практическая ценность

1) Изготовлен транслятор с языка АЛМО для ЕС-1010, который является первым транслятором с языка АЛМО для мини-ЭВМ и первым синтаксически ориентированным транслятором с языка АЛМО.

2) разработан новый вариант алгоритма анализа, названный управляемым LF –анализом, который отличается от оригинального способом проверки альтернативы, достаточно прост и вполне эффективен, но не накладывает настолько сильных ограничений на обрабатываемую грамматику (пользоваться первоначальным LF –анализом в случае АЛМО было бы невозможно).

3) разработан такой алгоритм расчета структуры перекрытия, которым имеет смысл пользоваться в тех случаях, когда вызываемый модуль неоднократно передает управление обратно не непосредственно вызывавшему его модулю.

В связи с тем, что транслятор создавался для мини-ЭВМ, автору пришлось преодолеть следующие трудности:

- а) во время разработки транслятора для машины ЕС-1010 можно было программировать только на языке ассемблер, и нам предстояло разбить транслятор на возможно более короткие и простые части, что заставило нас особенно тщательно подойти к проектированию транслятора;
- б) употребление традиционной структуры перекрытия, строящейся соответственно логической последовательности модулей, привело бы

к тому, что из-за недостаточной ёмкости оперативной памяти для таблиц транслятора осталось бы слишком мало места. При сборке и отладке транслятора, когда переводятся только небольшие программы, достаточно и небольших таблиц. На последнем этапе создания транслятора нужно построить оптимальную структуру перекрытия с учетом действительной структуры вызовов, возвратов и длины модулей.

Из-за малых размеров машины возникла необходимость решения следующих проблем генерирования объектного кода.

- Адресная часть команд занимает всего 8 битов, что затрудняет для переменных АЛМО распределение оперативной памяти.
- На машине имеется всего 3 (доступных для программы пользователя) регистра: I индексный, I сумматор и I вспомогательный. Это сделало необходимой организацию экономного использования регистров в переведённой программе. Например, вместо $a < b$ рассматривается $b > a$, если при этом уменьшается количество генерируемых команд.
- К переведённой программе в структуре перекрытия присоединяется интерпретирующая система, имеющая, в свою очередь, оптимальную структуру перекрытия.

Все имевшиеся до сих пор трансляторы с АЛМО не являлись синтаксически ориентированными. Тем не менее синтаксически управляемый транслятор обладает следующими преимуществами:

- а) по сравнению с традиционными трансляторами его структура проще и поэтому
 - легко разбивается на модули для программистов;
 - соединение готовых модулей выполняется быстрее и легче;
 - возможные изменения в языке могут быть сравнительно легко вписаны в транслятор;

б) с помощью подробной синтаксической таблицы можно добиться быстрого создания каркаса транслятора (его части, выполняющей синтаксический анализ). На этот каркас можно наложить семантические программы, выполняющие непосредственно сам перевод. Составленный таким образом транслятор обычно оказывается медленным по сравнению с традиционным, его можно сделать более быстрым путем локальных переработок, замены отдельных ветвей синтаксического дерева на более сложные семантические программы. Итак, мы всегда можем добиваться соответствующего равновесия между сложностью транслятора и временем перевода;

в) в случае необходимости создания транслятора для новой машины можно заново использовать большинство частей уже готового синтаксически управляемого транслятора.

5. Апробация работы

Результаты диссертационной работы докладывались на конференции "Системное программирование '75", организованной Обществом научной вычислительной техники им. Яноша Неймана и Физико-математическим отделом АН ВНР, а также на городском семинаре по программированию при факультете вычислительной математики и кибернетики МГУ под руководством проф. М.Р.Щура-Бура и на заседании кафедры системного программирования того же факультета.

6. Публикации. Структура и объём работы

По теме диссертации имеются публикации:

- Транслятор с языка АЛМО на ЕС-1010, внутренняя спецификация. Работа, предшествовавшая написанию трансляторов, которая включает в себя описание общих принципов и алгоритмов наиболее важ-

- ных частей транслятора. Научно-исследовательский институт прикладной вычислительной техники. Книга (60 стр.), Будапешт, 1974г.
- Алгоритмический машинно-ориентированный язык АЛМО. В сущности представляет собой перевод книги, содержащей определение языка с добавлением понятий, необходимых при пользовании транслятором, написанном для ЕС-1010. Научно-исследовательский институт прикладной вычислительной техники. Книга (105 стр.), Будапешт, 1975 г.
 - Транслятор с АЛМО для ЕС-1010. Доклад на конференции "Системное программирование '75", напечатанный в сборнике докладов конференции, стр.306-317, Будапешт, 1975 г.
 - ЕС-1010. Транслятор с языка АЛМО. Краткое описание для пользователей ЕС-1010, выполненное по просьбе ВИДЕОТОНА в виде проспекта (на русском языке, стр.5), Будапешт, 1976 г.

Диссертация состоит из введения, пяти глав и заключения, содержит 22 рисунка, включая блок-схемы, одну таблицу, библиографию и 3 приложения. Объем диссертации составляет 119 страниц (без приложений).

II. СОДЕРЖАНИЕ РАБОТЫ

Во введении к диссертации обсуждаются исторические аспекты темы и приводится краткое описание всех разделов работы.

В первой главе дается описание концепции АЛМО, обосновывается актуальность реализации АЛМО на мини-ЭВМ ЕС-1010 и показывается специфика и новизна реализации промежуточного языка высокого уровня на мини-ЭВМ.

Введена система обозначений, с помощью которой концепцию АЛМО можно описать сжато и наглядно. Коротко приводятся важней-

шие характеристики ЕС-1010 и объясняются причины выбора синтаксически управляемого транслятора.

Во второй главе формулируется постановка задачи и дается точное описание необходимых для применения транслятора (состава) технических средств и программного обеспечения, а также краткое описание всего процесса: от перевода программы до ее выполнения.

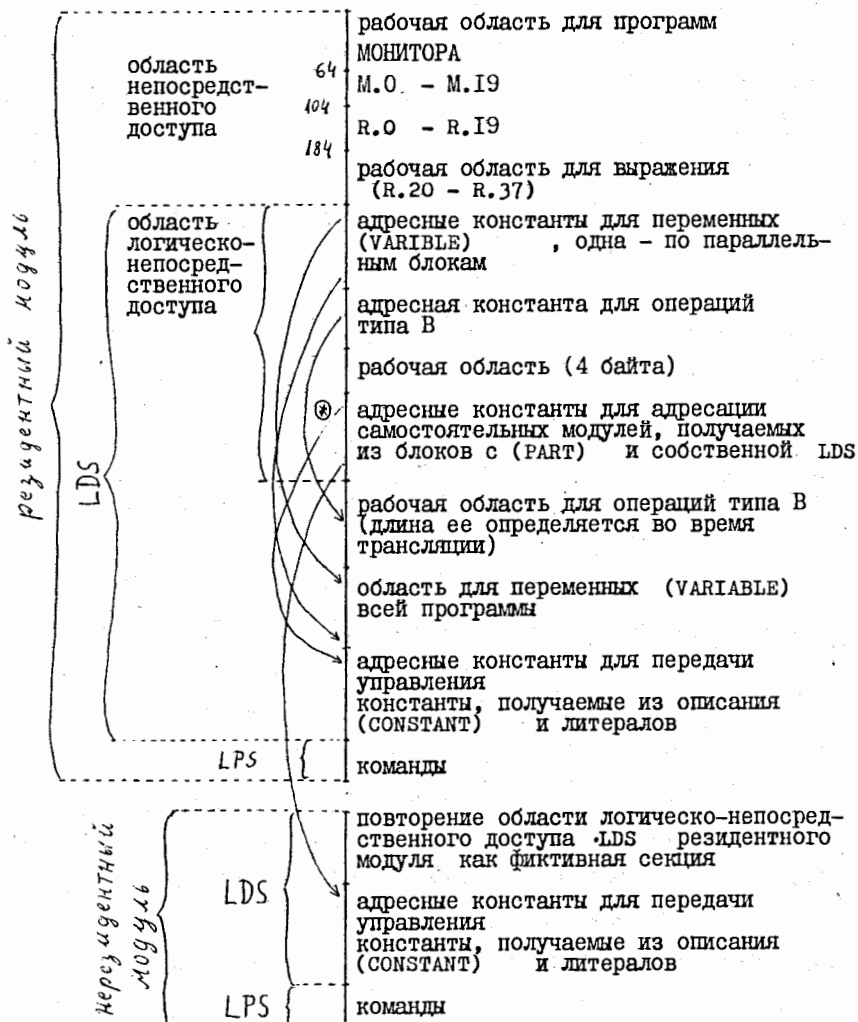
Третья глава содержит описание наиболее важных понятий АЛМО и ассемблера ЕС-1010 и показывает, как отдельные понятия АЛМО отображаются на язык ассемблера.

Довольно подробно описывается редактор связи, особенно для случая со структурой перекрытия. Это необходимо для понимания действий, связанных с блоками с (PART).

Показано, как отображается внешняя и внутренняя память, простые блоки, блоки с (PART), блоки подпрограмм (которые, в свою очередь, могут быть простыми блоками и блоками с (PART)), а также содержащиеся в них декларации.

Ниже, на стр.8, представлено распределение внутренней памяти машины для переведенной с языка АЛМО программы. Сравнительно большое число адресных констант обосновывается тем, что поскольку адресная часть машинной команды ЕС-1010 состоит всего из 8 разрядов, т.е. длина непосредственно адресуемой области составляет только 256 байтов, то адресация более длинных областей, используемых для различных целей, возможна только косвенным путем, посредством прежде всего адресных констант. Структуру переведенной простой программы наглядно поясняет программа MEMORY SCHEDULING, CORRECTED (см. в Приложении II. в), а структуру программы с (PART) показывает программа PART TEST в Приложении III.б.

Программа на языке АЛМО с (PART) отображается следующим образом:



Простая программа (без (PART)) отличается следующим:

- в области $\textcircled{\text{B}}$ имеется только одна адресная константа, указывающая на собственную LDS;
- отсутствуют нерезидентные модули.

На наглядном примере показывается отображение меток и команд передачи управления.

Перечисляются по пунктам действия, происходящие при обработке символов (BEGIN) и (PART), а в случае (END) (ввиду его небольшой сложности) эти действия показываются на блок-схеме.

Объясняется причина синтаксического объединения понятий выражения и отношения, затем подробно обсуждается их перевод, поскольку эта часть транслятора является одной из самых больших и самых важных, наиболее часто употребляемых его частей. Описываются принципы, на основе которых было решено, в каком случае (при использовании каких функций) необходимо обращение к интерпретирующей системе и в каком нет. Аналогичные рассуждения справедливы и при трансляции операторов.

В результате следующие операторы переводятся прямым путем: простая пересылка, кроме того случая, когда она принадлежит

к типу В и размер её не превышает D ;

стековая запись;

стековая выборка;

безусловные и условные операторы управления;

(COPY) - оператор обмена.

С обращением к интерпретирующей системе переводятся:

простая пересылка, если она типа В и размер её превышает D ;

перевод в значение;

перевод в текст;

операторы ввода и вывода;

(INCOPY), (OUTCOPY) - операторы обмена;

оператор останова.

Перевод отдельных операторов обсуждается следующим образом: записывается синтаксис оператора в преобразованном виде, места, важные с точки зрения обсуждения, отмечаются стрелками (если нужно, занумерованными), с помощью которых затем описывается состояние и действие. В качестве примера рассмотрим перевод команд "верхнее окончание цикла" и "нижнее окончание цикла". Эти два типа операторов при трансляции частично объединяются.

(FOR) <выражение> ↓ (GOTO) <метка>
 <прод.оконч.цикла>

<прод.оконч.цикла> ::= (ELSE STEP) <постоянный модификатор>₁ |
 (AND STEP) <постоянный модификатор>₁ ↑₂

В точке, помеченной знаком ↓, понятие "выражение" уже переведено. Последовательность генерированных операторов оканчивается оператором передачи управления, который выполняется, если отношение ложно.

- В точках ↑₁ и ↑₂ следующие состояния совпадают:
- метка находится в магазине выражения,
 - постоянный модификатор запоминается как целое число,
 - при трансляции выражения-отношения под действием (FOR) запоминается индекс M-переменной, фигурирующей в первом члене отношения.

В обоих случаях из этой информации должно генерироваться одно и то же, только в разном порядке. В случае "верхнего окончания цикла", т.е. в точке ↑₁, генерируемая последовательность должна быть такая:

...
 if goto lg
 (GOTO) <метка>
 lg: M.i + <постоянный модификатор> → M.i

тогда как в случае "нижнего окончания цикла", т.е. в точке ↑₂, должна генерироваться такая последовательность:

...
 if goto lg
 M.i + <постоянный модификатор> → M.i
 (GOTO) <метка>

lg: ...
 где lg - метка, генерируемая транслятором.

Конец третьей главы посвящен описанию размещения текстов библиотечных вставок. Табличные данные и библиотечные вставки относятся друг к другу следующим образом: в тексте табличной вставки не может быть ни табличной, ни библиотечной вставок, а в тексте библиотечной вставки может содержаться как табличная, так и библиотечная вставка, но их вложенность друг в друга не должна превышать тройной глубины.

В четвертой главе дается описание самого транслятора и генерируемой программы. Подробно описывается используемый нами метод анализа и перевода, а также сравнение с другими синтаксически управляемыми трансляторами. Вводится понятие управляемого LF-анализа, описывается, как необходимо преобразовать для него исходную грамматику. Сущность управляемого LF-анализа состоит в том, что в ходе нисходящего детерминированного анализа для выбора подходящей альтернативы иногда при просмотре вперед входного текста, ищется какое-нибудь, по отношению ко всей грамматике довольно простое, синтаксическое понятие.

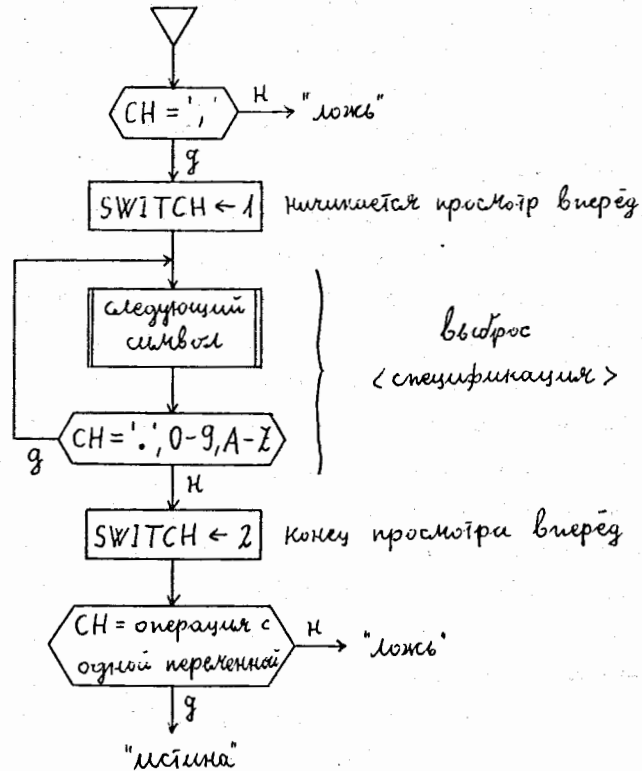
Например, выражение после синтаксического преобразования выглядит следующим образом:

обозначение: << >> управляющая программа
 { } множество левых терминалов

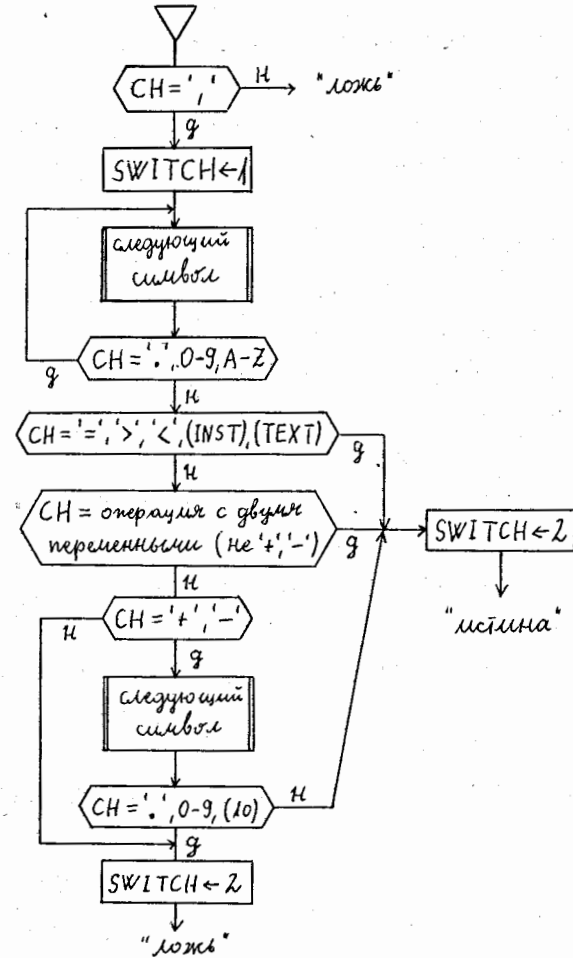
<выражение> ::= <основное выражение> <продолжение выражения>
 <основное выражение> ::= { константа } <константа> | <переменная>
 <продолжение выражения> ::= <операция 1> , <операция 1>
 <продолжение выражения> | <<конец выражения>> | , <выражение>
 , <операция 2> <продолжение выражения>
 { константа } = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, -, (10), +, (2), (4), (8), (16), "

Блок-схема «операция 1»:

Обозначение: CH = актуальный символ



Блок-схема «конец выражения»:



Просмотр вперед не имеет никакого принципиального влияния на дальнейший ход трансляции. После теоретического обоснования анализа и преобразования синтаксиса даётся несколько практических советов по реализации транслятора. Первая часть главы, кото-

рую можно назвать теоретической, заканчивается описанием стратегии обработки ошибок.

Во втором пункте главы рассмотрены части транслятора, которые с точки зрения применяемых алгоритмов могут претендовать на более широкий интерес. Алгоритмы показываются на блок-схемах или с помощью АЛГОЛА-60. Производится ознакомление со следующими алгоритмами:

- составления при втором проходе транслятора карт, описывающих систему перекрытия для редактора связи в случае программы с (PART);
- распределения памяти для переменных;
- обработки операндов;
- перевода выражения;
- генерирования команды на языке ассемблера;
- составления таблицы идентификаторов и литералов и осуществления над ними наиболее важных действий;
- оформления и выдачи документации, с их обсуждением в зависимости от вызовов транслятора с АЛМО;
- обработки символов входной программы, включая обработку вставок, а также осуществление упоминавшегося выше в связи с анализом просмотра вперёд.

В третьем пункте описывается способ расчёта системы перекрытия, которым выгодно пользоваться в тех случаях, когда вызываемый модуль неоднократно передаёт управление не непосредственно вызывавшему его модулю. Сущность метода состоит в том, что исходя из структуры вызовов и возвратов модулей строится граф, который раскрашивается, при этом одинаковыми цветами выделяется один и тот же участок памяти.

Четвёртый пункт знакомит со структурой системы интерпретации, её главнейшими группами.

В пятой главе описываются характерные особенности транслятора, приводятся данные об объёмах различных его частей, о числах сообщений ошибок во время перевода и выполнения программы. В отдельной таблице собраны значения времен трансляции с АЛМО в характерных режимах транслятора для программ, дающихся в приложении.

В заключении кратко излагаются основные результаты диссертации.

III. ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

В работе получены следующие результаты:

1. С помощью введения управляющих символов разработан удобный для практических применений вариант беступикового синтаксического анализа сверху-вниз, который сохраняет все преимущества исходного метода и при этом применим для существенно более широкого множества грамматик.

2. Разработана соответствующая эффективная методика преобразования синтаксиса языка, которая была успешно применена к синтаксису языка АЛМО. При этом получены две таблицы, которые без изменения могут быть использованы при написании других синтаксически ориентированных трансляторов с АЛМО.

3. Доказана возможность и разработана методика эффективной реализации машинно-ориентированного языка высокого уровня на мини-машинах, обеспечивающая, в частности, произвольную структуру перекрытия, взаимодействие частей PART с автокодными модулями, заданную точность вычислений.

4. Разработан и внедрен в эксплуатацию акционерным обществом ВИДЕОТОН транслятор с языка АЛМО на мини-ЭВМ ЕС-1010.

ПУБЛИКАЦИИ

по теме диссертации

1. Фолтени В. Транслятор с языка АЛМО на ЕС-1010, внутренняя спецификация. Научно-исследовательский институт прикладной вычислительной техники. Будапешт, 1974, 60 с.
2. Фолтени В. Алгоритмический машинно-ориентированный язык АЛМО. Научно-исследовательский институт прикладной вычислительной техники, Будапешт, 1975, 105 с.
3. Фолтени В. Транслятор с АЛМО для ЕС-1010. В сборнике "Системное программирование '75". Общество научной вычислительной техники им. Яноша Неймана. Будапешт; 1975, с.306-317.
4. Фолтени В. ЕС-1010. Транслятор с языка АЛМО. Проспект ВИДЕОТОНА, 1976, 5 с.

Рукопись поступила в издательский отдел
23 декабря 1981 года.