

УДК 004.4'23

Программа РЕКУРРЕНТ – инструмент анализа и синтеза целочисленных преобразователей

Степченко Д. Ю.

1. Причины разработки программы РЕКУРРЕНТ

При разработке параллельных вычислительных архитектур достаточно перспективным подходом является использование принципа рекуррентных потоковых вычислений на базе рекуррентно-динамической вычислительной парадигмы [1, 2]. Один из перспективных механизмов для организации рекуррентных потоковых вычислений – реализация концепции самодостаточности данных [1, 3], когда функциональные поля (теги), сопровождающие данные, содержат всю информацию по их обработке. Использование рекуррентной развертки дает возможность определить дальнейшую процедуру обработки данных на ряд шагов. В частности, такой механизм является одной из особенностей рекуррентного обработчика сигналов (РОС) [4].

Успешная реализация указанного механизма предполагает, в частности, возможность поиска рекуррентных последовательностей (значений) тегов, которые обеспечили бы реализацию вычислительных процессов за минимальное число вычислительных шагов и с минимальными накладными расходами (с минимумом вспомогательных процедур). Одновременно должен быть обеспечен поиск настроечных параметров для некоего комбинационно-регистрового устройства, обеспечивающего рекуррентную развертку тегов. В РОС таким устройством является преобразователь тегов (ПТ).

Известно, что работы в данном направлении велись в Киевском институте кибернетики (Бариновой И. А. и Махибородой А. В.), а также и в ИПИ РАН. Однако функционально-законченное программное средство, в полной мере отвечающее поставленным задачам по уровню функциональных и сервисных возможностей, так и не было создано. В связи с этим было принято решение о создании программы РЕКУРРЕНТ. О существовании в настоящее время программ со схожими возможностями, к сожалению, неизвестно.

2. Назначение программы РЕКУРРЕНТ и требования к ней

Программа РЕКУРРЕНТ предназначена для решения задач анализа и синтеза целочисленных преобразователей, работающих в режиме генерации рекуррентных последовательностей и поддерживающих параллельные рекуррентные процессы в параллельных структурах. Порождаемые преобразователями последовательности чисел используются как динамически воспроизводимые эквиваленты программ, а сами преобразователи при этом рассматриваются как средства компрессии последовательностей инструкций.

В частности, РЕКУРРЕНТ служит для определения параметров настройки для ПТ в системе РОС, которые обеспечивали бы правильную смену тегов. Другое важное предназначение программы РЕКУРРЕНТ – анализ структуры графа функционального преобразователя.

Программа РЕКУРРЕНТ подлежит интеграции в систему параллельного программирования РОС (СПРОС) [4], которая является инструментальным средством автоматизации программирования РОС.

Программа РЕКУРРЕНТ должна:

– диагностировать рекуррентные последовательности для заданной функции при

фиксированной разрядности и константах (фиксированном функциональном преобразователе);

- вводить булевы функции (функциональные преобразователи);
- выполнять поиск заданной рекуррентной последовательности для заданной функции;
- использовать механизмы маскирования и конкатенации;
- обладать интерфейсом, дружественным пользователю;
- иметь стыковку с подсистемой абстрактного синтеза;
- функционировать на современных ПК;
- функционировать в операционных системах семейства Windows.

Отсутствие функционально-законченного программного средства, отвечающего перечисленным задачам, послужило стимулом к разработке программы РЕКУРРЕНТ 3.0/DOS (Хлебниковым С. А. и Степченковым Д. Ю.).

3. История создания программы РЕКУРРЕНТ. Выбор языка программирования. Особенности функционирования программы

В качестве языка программирования был выбран Smalltalk, на время создания программы обладавший наибольшими возможностями для реализации поставленных задач. Программа функционировала в операционной системе MS-DOS.

Однако выбор языка оказался не слишком удачным. Выяснилось, что ни РЕКУРРЕНТ 3.0, ни среда интерпретатора – выбранной версии языка Smalltalk не функционируют ни на современных ПК, ни в современных операционных системах. В связи с этим работы над РЕКУРРЕНТ 3.0/DOS были остановлены, и возникла необходимость в написании новой версии программы. Для создания РЕКУРРЕНТа для Windows был выбран язык Visual C++ 6.0. Новые версии языка Smalltalk – крайне дорогостоящие, в то время как язык C++, на котором написаны основные компоненты системы СПРОС, отвечает всем условиям для успешного создания программы. Кроме того, одинаковый язык программирования даст возможность интеграции РЕКУРРЕНТа и других компонентов СПРОС. Одновременно было принято решение о расширении функциональных возможностей программы.

В результате была создана программа, функционирующая на современных ПК, под операционными системами Windows 9x – Windows 2000 – Windows XP.

Минимальные требования к аппаратуре:

Операционная система: Windows 95

Процессор: Pentium 2 с тактовой частотой 350 МГц или аналог

RAM: 128 Мб.

Видео: стандартная видеокарта SVGA

Манипулятор: мышь.

Рекомендуемые требования:

Операционная система: Windows 2000

Процессор: Pentium 3

RAM: 256 Мб

Требования к объему жесткого диска будут уточнены позднее, после реализации оконного интерфейса.

4. Базовая терминология, используемая в рамках РЕКУРРЕНТ. Структура и функциональные возможности программы

Программа РЕКУРРЕНТ базируются на следующих понятиях: функция, преобразователь, конкатенация, синтез-преобразователь, граф, маскирование.

Функции определяются как побитовые преобразования над строками битов произвольной длины. Над аргументами и константами допускаются такие преобразования, как инверсия, операция логического И, операция логического ИЛИ, сложение по модулю 2, битовая инверсия, сдвиги вправо и влево: с умножением бита, циклические, с заполнением разряда единицей либо нулем.

Функциональный преобразователь формируется на базе функции путем фиксации ее разрядности и констант (при их наличии). Фиксация разрядности и констант позволяет произвести диагностику функционального преобразователя с целью анализа его структуры и получения графа.

Следует различать простой функциональный преобразователь, конкатенируемый преобразователь и синтез-преобразователь.

Конкатенируемый преобразователь получается в результате выполнения операции конкатенации – формирования составных преобразователей на основе простых. При конкатенации происходит совмещение разрядов в заданном порядке без функциональной связи между преобразователями.

Синтез-преобразователь получается в результате работы подсистемы абстрактного синтеза по заданной структуре.

Граф – представление структуры полной таблицы соответствий входов и выходов преобразователя. Может быть изображен графически или представлен в текстовом виде, как информация о результатах анализа структуры преобразователя.

Компоненты, из которых может состоять граф:

- полюс – структурное образование в графе, элемент которого переходит сам в себя;
- кольцо – структурное образование графа в виде циклически повторяющихся элементов, число которых определяет длину кольца;
- дерево – структурное образование графа, не включающее в себя циклически повторяющихся элементов (колец, за исключением одного); дерево всегда сходится или к кольцу или к полюсу.

Все битовые строки диапазона представимости преобразователя могут иметь только однократную входимость в рекуррентную последовательность. Для преодоления этого служит маскирование: внесение избыточности в разрядность преобразователя и наложение маски для отбора значащих разрядов. Структура графа по форме не зависит от наложения маски.

Маскирование – процесс смены, наложения и удаления маски. Маскирование предназначено для преодоления того факта, что все битовые строки диапазона представимости преобразователя могут иметь только однократную входимость в рекуррентную последовательность. Маска – процесс внесения избыточности в разрядность преобразователя и отбора значащих разрядов.

Логически систему РЕКУРРЕНТ под Windows можно разбить на 3 модуля: "Работа с функциями", "Работа с преобразователями", «Поиск» (см. рис. 1).

В рамках модуля "Работа с функциями" выполняется построение функций в базе логических операций над битовыми строками, и обеспечиваются следующие сервисные операции:

- редактирование,
- удаление,
- сохранение в библиотеке,
- переименование,
- листинг,
- переименование.

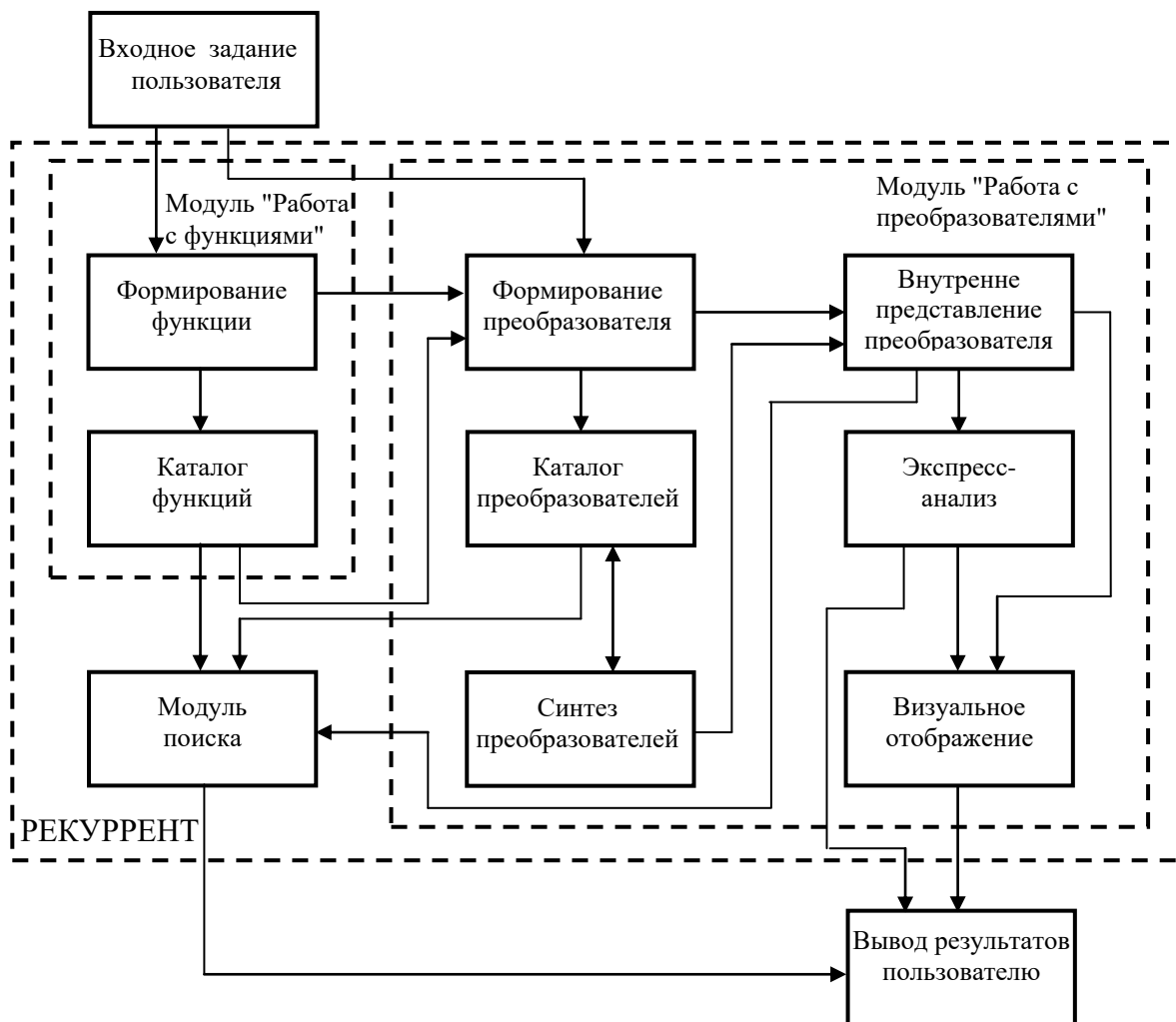


Рис. 1. Функционально-модульная структура программы РЕКУРРЕНТ

В рамках модуля "Работа с преобразователями" выполняются следующие операции:

- построение преобразователей на основе библиотеки сформированных функций и построение конкатенаций на базе простых преобразователей;
- построение синтез-преобразователей на основе заданной структуры;
- построение графа преобразователя и экспресс-анализ типа структуры, порождаемой преобразователем (информация о кольцах, узлах кольца, деревьях и проч.);
- обеспечение сервисных операций над преобразователями (аналогично модулю работы с функциями);

В рамках модуля "Поиск" должны выполняются ряд операций поиска преобразователей, содержащих искомые рекуррентные последовательности. Поиск основан на переборе всех возможных вариантов, в соответствии с заданными параметрами поиска.

В настоящее время различаются «поиск функций», «поиск конкатенации», «поиск преобразователей»; каждый из них берет за основу соответствующие объекты, занесенные пользователем в библиотеку программы.

Возможно использование маскирования с возможностью искать различные рекуррентные последовательности с различными масками в рамках одного функционального преобразователя, перебор различных вариантов разрядности преобразователя и констант, использование механизма конкатенации построенных преобразователей.

Параметры поиска и диапазон их изменения определяются пользователем.

5. Место РЕКУРРЕНТа в СПРОС. Программа РЕКУРРЕНТ и варианты РОС

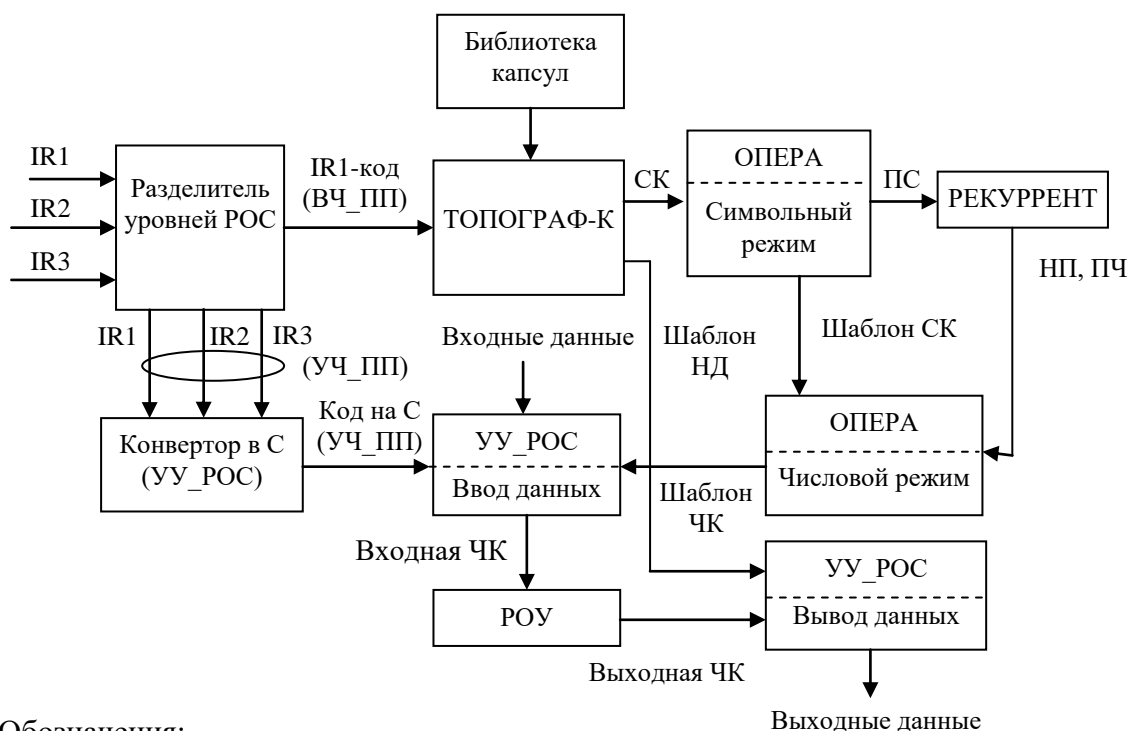
Хотя программа РЕКУРРЕНТ – логически завершенная, она будет, в основном, использоваться в составе системы СПРОС как некая ее часть. Более того, она создавалась как инструмент для работ по созданию РОС, а также для построения рекуррентного операционного устройства [5] в рамках НИР "Архитектура" [6]. Работа над РЕКУРРЕНТОм была продолжена в рамках НИР "Сигнал" [7]. Для того, что бы быть полноценной частью СПРОС, РЕКУРРЕНТ должен выполнять свою роль в проектировании всех вариантов РОС.

Существует 4 основных варианта РОС: РОС общего назначения, который подразделяется на РОС с временной и с комбинационной избыточностью; проблемно-ориентированный РОС; специализированный РОС; обработчик сигналов с автономным механизмом реализации рекуррентной развертки функциональных полей [4]. Каждый из этих вариантов оказывает влияние, в том числе, и на средства моделирования. Программа РЕКУРРЕНТ задействована в проектировании каждого из них.

Особо стоит указать на возможности использования РЕКУРРЕНТа для РОС общего назначения с временной и с комбинационной избыточностью.

Если в РОС с временной избыточностью основной задачей РЕКУРРЕНТа является поиск настроечных параметров для ПТ, то в РОС с комбинационной избыточностью используется универсальный функциональный преобразователь с 200-процентной избыточностью (определенный с помощью программы РЕКУРРЕНТ), и производится поиск всего лишь одного настроечного параметра – маски, позволяющей «отобрать» нужные разряды среди избыточных.

На рис. 2 показана схема СПРОС, взятая из [4]. Как видно из рисунка, программа РЕКУРРЕНТ взаимодействует с программой ОПЕРА.



Обозначения:

ВЧ – вычислительная часть;

НД – набор данных;

НП – настроечные параметры ПТ;

ПП – прикладная программа;

ПС – последовательность символов;

ПЧ – последовательность чисел;

РОУ – рекуррентное операционное устройство;

СК – символьная капсула;

ТОПОГРАФ-К – трансформация ориентированного потокового графа в капсулу;

УУ – устройство управления;

УЧ – управляющая часть;

ЧК – числовая капсула

Рис. 2. Структура СПРОС

Программа ОПЕРА формирует требуемую последовательность трансформации функциональных полей, передаваемую в РЕКУРРЕНТ в виде файла входных данных. Необходимыми входными данными для РЕКУРРЕНТа являются также тип и аппаратный базис преобразователя тегов, а также символьная капсула. Последовательность символов из файла переводится в последовательность уникальных чисел, для которой находится комбинация настроечных параметров преобразователя тегов. Результат работы программы РЕКУРРЕНТ – преобразование последовательности символов в последовательность чисел и набор настроечных параметров ПТ – передается в программу ОПЕРА.

6. Выводы

Программа РЕКУРРЕНТ – необходимая часть для функционирования СПРОС. Программа РЕКУРРЕНТ 3.0/DOS – законченная и полностью работоспособная версия для работы на старых ПЭВМ под операционной системой MS DOS. Программа РЕКУРРЕНТ/Windows обладает расширенными функциональными возможностями по сравнению с программой РЕКУРРЕНТ 3.0/DOS.. В настоящее время существует полностью работоспособная версия РЕКУРРЕНТ/Windows, работающая в пакетном режиме. Эту версию можно успешно использовать для разработки рекуррентной архитектуры. В настоящее время ведутся работы по созданию оконной версии РЕКУРРЕНТ/Windows.

В качестве возможного направления дальнейшего развития РЕКУРРЕНТа можно наметить:

- 1) изменение алгоритма поиска на "более интеллектуальный", в котором поиск происходил бы с использованием анализа на предмет возможной входимости искомым последовательности в графы функциональных преобразователей, повторенных на основе той или иной функции;
- 2) более тесную интеграцию РЕКУРРЕНТа в систему СПРОС;
- 3) интеграцию с новой, более интеллектуальной системой абстрактного синтеза, разработка которой сейчас ведется в ИПИ РАН;
- 4) визуализацию графов преобразователя.

Литература

1. *А.В.Филин*. Динамический подход к выбору архитектуры вычислительных устройств обработки сигналов. // Системы и средства информатики. Вып. 11. - М.: Наука, 2001, с. 247-261.
2. *А.В.Филин*. Особенности обработки сигналов на процессоре с рекуррентно-динамической парадигмой вычислений. // Системы и средства информатики. Вып. 11. - М.: Наука, 2001, с. 262-282.
3. *А.В. Махиборода*. Проблемы реализации динамического параллелизма в естественно-надежных компьютерах. // Системы и средства информатики. Вып. 7. - М.: Наука, 1995, с. 198-207
4. *Ю.А.Степченко, В.С.Петрухин*. Перспективы развития цифровых сигнальных процессоров и возможная реализация рекуррентного обработчика сигналов. // Исследование и построение вычислительных систем. Спецвыпуск. - М.: Наука, 2004, 32 с.
5. *Ю.А.Степченко, В.С.Петрухин, А.В.Филин*. Рекуррентное операционное устройство для процессоров обработки сигналов. Вып. 11. - М.: Наука, 2001, с. 283-315.
6. Исследование проблем создания естественно- надежных ЭВМ. Отчет о НИР (заключительный), шифр "Архитектура", № г.р. 01.9.50 000083, ИПИ РАН, 1996
7. Исследование новой вычислительной парадигмы и разработка на её основе логического проекта динамического многопоточного процессора обработки сигналов. Отчет о НИР N 2 (промежуточный), книга 2, шифр "Сигнал", № г.р. 01.2.00 104927, ИПИ РАН, 2002.