

Обзор языка программирования COLAMO

Петров А.В., 20.11.2010

Вводная часть

Язык COLAMO, а точнее система программирования, в которую входит данный язык, предназначен для реализации алгоритмов на реконфигурируемых вычислительных системах (к которым авторы языка, судя по ряду публикаций [1], относят прежде всего платы-сопроцессоры с множеством используемых на них ПЛИС).

Основные результаты получены в течение 20-ти лет разработки системы в Научно-исследовательском институте многопроцессорных вычислительных систем имени академика А.В.Каляева Южного федерального университета (г. Таганрог) - <http://mvs.sfedu.ru/>. Лидерами данной научной группы по всей видимости являются А.В.Каляев и И.И.Левин - <http://mvs.sfedu.ru/persons/person.html>.

Список целей, которые преследуют авторы COLAMO [1]:

1. Программирование как структурной, так и процедурной составляющей на ЯВУ;
2. Реконфигурация прикладных задач без участия высококвалифицированного схемотехника;
3. Обеспечение совместимости и переносимости проектов между PBC разных архитектур;
4. Масштабирование прикладной задачи при увеличении ресурса;
5. Удаленное использование вычислительных ресурсов PBC.

В составе разработанной системы заявлено наличие следующих компонентов:

1. Транслятор ассемблера Argus 3.0. Язык структурно-процедурного программирования Argus представляет собой низкоуровневый язык для описания процедурной составляющей прикладной параллельной программы. Программа на Argus организует потоки данных на уровне команд контроллеров распределенной памяти, обеспечивая их синхронизацию;
2. Транслятор языка программирования PBC высокого уровня COLAMO 2.0 в ассемблер Argus. Транслятор также ответственен за создание структурной составляющей в объектном представлении, которая автоматически передается в среду синтеза Fire!Constructor ;
3. IDE Argus IDE 3.0;
4. Среда Fire!Constructor – обеспечивающую синтез масштабируемых параллельно-конвейерных процедур на основе библиотеки IP-ядер и интерфейсов – синтез происходит в язык VHDL.

Язык COLAMO

В языке COLAMO отсутствуют явные формы описания параллелизма. Распараллеливание достигается с помощью объявления типов переменных и индексации элементов массивов (язык ориентирован на потоки данных). Естественным способом реализации неявного описания параллелизма в программе является правило единственной подстановки, которое широко используется в языках потока данных и заключается в следующем: переменная может получить значение в программе только один раз (детально значение правила будет описано для каждого типа переменных в отдельности). Данное правило приводит к противоречию с традиционными принципами программирования. В связи с этим делается следующее ограничение: это правило действует в специальных

конструкциях предлагаемого языка, описывающих вычислительные структуры, задаваемые пользователями.

Операторы COLAMO делятся на три группы:

1. операторы описания вычислительных структур;
2. операторы управления вычислительным процессом;
3. операторы описания данных и программно-аппаратных ресурсов.

Фундаментальным типом вычислительной структуры в языке COLAMO является конструкция «кадр». Кадр языка соответствует совокупности арифметико-логических команд, выполняемых на различных ЭП и контроллерах распределенной памяти, соединенных между собой в соответствии с информационной структурой алгоритма таким образом, что вычисления производятся с максимально возможным параллелизмом и асинхронностью. Кадр фактически определяет вычислительную структуру и потоки данных в РВС в данный момент времени. При этом все операции в теле кадра выполняются асинхронно с максимальным параллелизмом, а последовательность смены кадров однозначно определяется программистом.

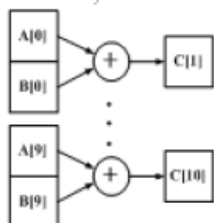
Подробное описание всех конструкций языка приведено в работе [2].

Пример программы

```

VAR A,B,C: Integer [10 : Vector] Mem;
VAR I : Number;
CADR SummaVector;
  For I := 0 to 9 do
    C[I] :=A[I]+B[I];
  ENDCADR;

```

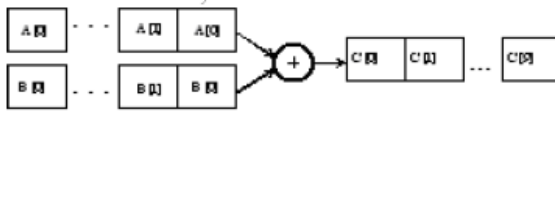


а)

```

VAR A,B,C : Integer [10 : Stream] Mem;
VAR I : Number;
CADR SummaStream;
  For I := 0 to 9 do
    C[I] :=A[I]+B[I];
  ENDCADR;

```



б)

Рис.2. Параллельное и последовательное сложение массивов.

Транслятор COLAMO достаточно сложен и описан в работе [3, 4]. Одним из фундаментальных понятий, используемых при построении транслятора является понятие многоярусного информационного графа программы (см. работы [5, 6]).

Производительность

Производительность можно оценить по работе [7]: “Реализация модулей «Вычитание прямого сигнала» и «Обращение интегрального уравнения» на РВС с использованием языка COLAMO показала достаточно высокую производительность РВС, которая составила порядка 50 Гфлопс. Выигрыш по времени решения обеих подзадач по сравнению с обычным персональным компьютером составил около 100 раз” (параметры РВС в статье не раскрываются).

Список литературы

1. А.И. Дордопуло, И.И. Левин, В.А. Гудков, А.А. Гуленок [Комплекс средств разработки прикладных программ на реконфигурируемых вычислительных системах](#) // «Искусственный интеллект» №3, 2004
2. Подробное описание языка COLAMO - <http://colamo.parallel.ru/>
3. Лексический и синтаксический анализ для языка COLAMO
4. Синхронизация
5. В.В.Воеводин, Вл.В.Воеводин. Параллельные вычисления – С.-Петербург. Изд-во «БХВ-Петербург», 2002.
6. И.А. Каляев, И.И. Левин, Е.А. Семерников, В.И. Шмойлов [Реконфигурируемые мультиконвейерные вычислительные структуры](#), Издательство ЮНЦ РАН, 2008
7. А.Г. Коваленко [Решение задачи диагностики дорожных покрытий на реконфигурируемой вычислительной системе с применением языка COLAMO](#) // Журнал «Искусственный интеллект» №3, 2009
8. [Обсуждение проекта на electronix.ru](#)
9. [Статья в журнале Суперкомпьютеры](#)
10. В.Б. Коваленко, М.С. Кочерга, Е.А. Семерников [Реконфигурируемые вычислительные системы для цифровой обработки сигналов](#) // Журнал «Искусственный интеллект» №3, 2009
11. И.И. Левин, А.И. Дордопуло, О.М. Омаров, [Унифицированное представление алгоритма задачи для произвольного варианта распараллеливания](#) // «Искусственный интеллект» №3, 2004