

УДК 004.4'416

Битнер В.П.^{1,2}

¹ Московский физико-технический институт (государственный университет)

² ЗАО «МЦСТ»

Система интерпретации промежуточного представления программы в оптимизирующем компиляторе

Начальным этапом трансляции в оптимизирующем компиляторе, созданном и развиваемом для архитектуры «Эльбрус» [1], является преобразование исходного кода программы в промежуточное представление, предназначенное для удобства генерации кода и/или проведения различных оптимизаций [2]. Промежуточное представление исходной программы можно интерпретировать как программный код, который приводится к машинному коду и исполняется (оно основано на ассемблере целевой архитектуры, но при этом не является программой на языке ассемблера).

Принято различать два типа представления исходной программы: высокоуровневый, максимально абстрагированный от особенностей конкретной вычислительной архитектуры и предназначенный главным образом для межпроцедурных оптимизаций, и низкоуровневый, приближенный к архитектуре некоторой конкретной вычислительной системе и предназначенный для оптимизаций внутри процедуры [3]. Данная работа посвящена разработке системы интерпретации промежуточного представления низкоуровневого типа посредством трансляции промежуточного кода в рабочую программную среду, которая обладает необходимыми свойствами надежности, доступности, удобства использования. Техническая реализация трансляции строится на моделировании элементов промежуточного представления Си-кодом, выполняемом с сохранением функциональной эквивалентности. Моделируются не только элементы промежуточного представления, но и аппаратная поддержка исполнения программы на процессоре с архитектурой «Эльбрус», например, моделирование асинхронной предподкачки массивов данных [4] или процесса

использования стека данных. Результатом трансляции промежуточного представления является программный код на языке Си, исполняемый на архитектуре x86 и в точности повторяющий функциональность промежуточного кода исходной программы.

Особое внимание в работе уделяется моделированию архитектурных особенностей, при котором достигаются наиболее простые и эффективные решения: реальный регистр моделируется как переменная типа `int`, реальный стек – как массив элементов типа `char`, вызов процедуры – как инициализация «станка» – указатель типа `void` – и вызов по косвенности с использованием средств языка Си и так далее. Общий принцип моделирования основывается на замене всех непереносимых объектов архитектуры «Эльбрус» эквивалентными (в смысле функциональности) и операбельными объектами архитектуры x86.

В процессе разработки оптимизирующего компилятора архитектуры «Эльбрус» система интерпретации промежуточного представления программы актуальна как необходимая основа для создания нового эффективного отладочного средства. Такое отладочное средство будет использоваться для верификации оптимизаций в оптимизирующем компиляторе, особенно в случаях, когда ошибка в оптимизации не приводит к явному ее обнаружению, то есть проявляется только на этапе исполнения программы.

Литература

1. *Волконский В.Ю.* Оптимизирующие компиляторы для архитектуры с явным параллелизмом команд и аппаратной поддержкой двоичной совместимости // Информационные технологии и вычислительные системы. – 2004. – №3. – С. 4-26.
2. *Ахо А.В., Сети Р., Ульман Д.Д.* Компиляторы: принципы, технологии и инструменты. – М.: Вильямс. – 2003 с. 768.
3. *Дроздов А.Ю., Степаненков А.М.* Методы комбинирования алгоритмов анализа и оптимизаций в современных оптимизирующих компиляторах // Компьютеры в учебном процессе. – 2004. – №11. – С. 3-12.

4. Galazin A.B. [et al.]. A Software Instruction Prefetching Method in Architectures with Static Scheduling // Programming and Computer Software. – 2008. V. 34, N. 23. – P. 49-53.