

- ◆ определены функции и предложены режимы функционирования контроллера удаленной диагностики;
- ◆ разработаны алгоритмы функционирования контроллера удаленной диагностики;
- ◆ выполнены необходимые расчеты, доказавшие эффективность принятых решений.

**В.Ф. Гузик, А.О. Пьявченко, В.А. Переверзев**

### **ОБЗОР МЕТОДОЛОГИЙ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ ПРОБЛЕМНО-ОРИЕНТИРОВАННЫХ ВЫЧИСЛИТЕЛЬНЫХ УСТРОЙСТВ НА БАЗЕ ПЛИС**

Для решения задач различной степени сложности существуют различные методики построения и проектирования проблемно-ориентированных систем. Традиционный подход к проектированию, основанный на использовании интегральных схем (ИС), широко расширился с появлением программируемых логических интегральных схем (ПЛИС). При этом, если при проектировании традиционным способом необходимы знания типовых функциональных блоков, их свойств и возможностей, то при проектировании проблемно-ориентированных систем с использованием ПЛИС появилась возможность компоновать аппаратную часть из типовых модулей – IP-блоков, мегафункций или других проверенных проектов. Это дало возможность разрабатывать и изготавливать специализированные и высокопроизводительные устройства с высокими эксплуатационными параметрами в относительно сжатые сроки.

Как известно, сложные алгоритмы и системы перед их реализацией в аппаратуре проходят этапы проверки и моделирования на универсальных комплексах моделирования. Однако до последнего времени системные инженеры и программисты опирались в своей работе на языки C/C++, в то время как их партнеры по разработке аппаратуры предпочитали схемотехническое представление проектов и лишь в последнее время стали описывать свои разработки на языках описания аппаратуры (VHDL, Verilog и т.п.). Разница в способах описания проекта наносит явный ущерб не только успешности взаимного общения специалистов (разрабатывающих начинку одного кристалла), но и, что более существенно, замедляет процесс проектирования и ухудшает его качество.

В большинстве случаев процесс проектирования системы, содержащей БИС программируемой логики, носит многоуровневый, пошаговый характер с возвратами и корректировкой принятых решений. Каждый из этих этапов носит неоднозначный характер. В современных условиях в процессе проектирования участвует большое число разработчиков, причем, как правило, на разных этапах используют как нисходящий, так и восходящий подходы к проектированию, пытаясь использовать их лучшие черты. Процесс проектирования реализуется одновременно по четырем направлениям разработок: ПО, аппаратуры, конструкторской документации и протокола взаимодействия отдельных элементов системы.

Техническая сторона проектирования определяется тремя основными составляющими: элементной базой (ПЛИС), инструментарием (САПР) и методикой (маршрутом проектирования). При этом порядок этапов проектирования может носить абсолютно произвольный характер, т.е. выполняться последовательно, параллельно или в различных комбинациях.

Современное требование уменьшения времени, необходимого для выпуска конечного продукта, заставляет искать пути сокращения как времени выполнения любых отдельных этапов проектирования, так и устранения причин, вызывающих необходимость итерационных возвратов к предыдущим этапам проектирования.

Рассмотренные проблемы перевода программно реализованных алгоритмов в аппаратуру и современные требования к проектированию ПОВС привели к появлению таких новых подходов к проектированию, как сопряженное проектирование, верификация и моделирование.

Со-проектирование (сопряженное проектирование) – процесс параллельного проектирования аппаратных и программных средств, при котором осуществляется оценка целесообразности выбора аппаратной или программной реализации определенного фрагмента проекта. Этот процесс также дает проектировщикам возможность, увидев (на абстрактном уровне), как система могла бы работать с данным разделением аппаратных средств ЭВМ и программного обеспечения.

Уже существуют наборы программ, содержащие мощные инструментальные средства проектирования (например, Celoxica DK) и представляющие из себя системную среду разработки алгоритмических акселераторов и комплекса конструирования электронных устройств.

Предлагаемый маршрут проектировщика предоставляет разработчику комплексную интегрированную среду разработки, основанную на языках высокого уровня (ЯВУ), таких как C, C++, Pascal, Object Pascal, Java, и включает в себя полный процесс проектирования устройства от этапа ввода схемы, моделирования и заканчивая синтезом. Основной особенностью является совместимость с Transaction Level Model (TLM), что позволяет смешанное абстрактное моделирование и симуляцию программно-аппаратного окружения, которая позволяет итерационно уточнять уровень моделирования устройства.

Преимуществами использования данного подхода являются: увеличение производительности при конструировании аппаратно-программных комплексов, сокращение времени на разработку и улучшение общего качества проектирования. Все участники процесса конструирования устройства от проектировщиков системы и разработчиков аппаратных средств до разработчиков программного обеспечения и тестирующих могут четко разделить задачи от уровня системной спецификации до реализации.

**К.П. Цурко**

### **РЕАЛИЗАЦИЯ АЛГОРИТМА КОМПОНОВКИ НА ОСНОВЕ МЕТОДА ДИХОТОМИИ И МЕТОДА ВЕТВЕЙ И ГРАНИЦ**

Цель данной работы состоит в реализации и исследовании алгоритма дихотомического деления схемы, то есть разбиения графа на две равные части, с помощью метода ветвей и границ, который гарантирует нахождение оптимального решения, уходя от полного перебора всех вариантов решений.

Достижение указанной цели предполагает решение следующих основных задач:

- ◆ Построение графовых моделей схем ЭВА для этапа конструкторского проектирования.
- ◆ Исследование алгоритма дихотомического деления с помощью метода ветвей и границ.
- ◆ Реализация алгоритма на языке программирования высокого уровня.

Результатом проделанной работы является реализованный на языке программирования высокого уровня программный продукт, который позволяет решать