Уральский государственный архитектурно-художественный университет
Национальный исследовательский Томский государственный университет
Уральский федеральный университет имени первого Президента России Б.Н. Ельцина

НОВЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ В ИССЛЕДОВАНИИ СЛОЖНЫХ СТРУКТУР

МАТЕРИАЛЫ
ОДИННАДЦАТОЙ МЕЖДУНАРОДНОЙ КОНФЕРЕНЦИИ
6–10 июня 2016 г.

Томск
Издательский Дом Томского государственного университета
2016

В качестве сравнительной оценки энергоемкости теста на стадии выбора наилучшего теста принимается сумма чисел переменных (входных, выходных, внутренних), сменивших свои значения в результате соответствующего перехода автомата, которая вычисляется как $D = d(\boldsymbol{x}_{i1}, \boldsymbol{x}_{i2}) + d(\boldsymbol{z}_{i2}, \boldsymbol{z}_{i3}) + d(\boldsymbol{y}_{i1}, \boldsymbol{y}_{i2})$, где $d(\boldsymbol{x}_{i1}, \boldsymbol{x}_{i2})$, $d(\boldsymbol{z}_{i2}, \boldsymbol{z}_{i3})$, $d(\boldsymbol{y}_{i1}, \boldsymbol{y}_{i2})$ – расстояния Хэмминга между кодами входных, внутренних и выходных состояний автомата. Так как сделанное предположение носит вероятностный характер, то имеет смысл находить несколько (заданное число или все) тестов $(\boldsymbol{z}_{i1}, \boldsymbol{x}_{i1}, \boldsymbol{x}_{i2})$ с наибольшими оценками энергоемкости D, которые будут использоваться для более точного подсчета рассеивания мощности схемой путем логического или схемотехнического моделирования.

В случае, когда имеется только структурное описание исследуемой схемы, делается предположение, что максимум рассеиваемой мощности, при заданном состоянии $\boldsymbol{z}_{i1}$ последовательностной схемы наблюдается при смене значений максимального числа входных сигналов, т.е. в искомой тестовой тройке $(\boldsymbol{z}_{i1}, \boldsymbol{x}_{i1}, \boldsymbol{x}_{i2})$ расстояние Хэмминга между векторами $\boldsymbol{x}_{i1}$ и $\boldsymbol{x}_{i2}$ должно иметь максимальное значение. Соответственно предлагается искать достаточно длинную тестовую последовательность $(\boldsymbol{z}_{i1}, \boldsymbol{x}_{i1}, \boldsymbol{x}_{i2}, \ldots, \boldsymbol{x}_{ik})$, включающую среди прочих пары $\boldsymbol{x}_{ij}, \boldsymbol{x}_{i(j+1)}$ взаимно ортогональных векторов. Реализованы два метода генерации заданного числа k тестов на основе вышеприведенной идеи. Первый простой метод генерации такой тестовой последовательности состоит в генерации $k/2$ случайных наборов $\boldsymbol{x}_{i1}$ и последующем удваивании их числа за счет добавления после каждого набора инверсного ему.

Второй, более перспективный метод заключается в генерации неслучайной последовательности максимально удаленных (по Хэммингу) друг от друга наборов, как это делается для задач функционального тестирования схем [1]. В этом случае искомый тест будет обеспечивать также и равномерное покрытие булева пространства входных векторов, что важно для случая ограниченного числа тестовых наборов. Предлагается итеративный метод увеличения числа компонент тестовых наборов. Предположим, что уже имеется 2^n упорядоченных векторов длины n. При увеличении числа компонент векторов на 1 массив из 2^n имеющихся векторов удваивается. Введенной $(n+1)$-й компоненте каждого вектора из числа первых 2^n векторов присваивается 1, если он имеет четный номер, и значение 0 в противном случае, а $(n+1)$-й компоненте каждого вектора из добавленных 2^n векторов присваивается значение 0, если он имеет четный номер, и значение 1 в противном случае. Построение векторов гарантирует, что последовательность будет состоять из взаимно инверсных пар векторов и не будет содержать одинаковых векторов. Например, начав с векторов 0 и 1, получаются векторы 00, 11, 01 и 10 длины 2, 000, 111, 010, 101, 001, 110, 011, 100 длины 3 и т.д.

Литература

1. *Wu Sh.H.* A Distance-Based Approach // VLSI Design. 2008. Article ID 165709. 9 p.

К ПОСТРОЕНИЮ ТЕСТОВ ДЛЯ ПОСЛЕДОВАТЕЛЬНОСТНЫХ СХЕМ С ИСПОЛЬЗОВАНИЕМ ABC

Г.В. Кидярова

Национальный исследовательский Томский государственный университет, Томск, Россия
gvkidyarova@mail.ru

При построении тестовых наборов для логических схем можно использовать так называемое мутационное тестирование [1]. В этом случае возможные неисправности явно перечисляются, и для построения входной последовательности, различающей поведение эталонной и проверяемой (возможно неисправной) схем, можно использовать системы проектирования и верификации, например систему ABC [2]. При таком построении проверяющего теста достаточно часто используется метод, основанный на построении майтера: отождествляются соответствующие входы эталонной схемы и схемы с неисправностями, и соответствующие выходы схем отождествляются с входами вентилей XOR. Таким образом, на i-м выходе майтера реализуется функция $f_i = g_i \oplus h_i$, где g_i – функция, реализуемая эталонной схемой на i-м выходе, h_i – функция, реализуемая на i-м выходе проверяемой схемой с неисправностями. Проверяемая и эталонная схемы эквивалентны тогда и только тогда, когда функция, реализуемая на любом выходе майтера, тождественно равна 0. Ряд методов проверки тождественности равенства 0 булевой функции основан на проверке выполнимости конъюнктивной нормальной формы (КНФ) функции и используется в различных системах верификации логических схем. Одной из таких систем является система синтеза и верификации ABC, разработанная в университете Беркли, США. Система ABC позволяет пользователю решать задачи, связанные с синтезом, оптимизацией, исследованием и верификацией как комбинационных, так и последовательностных схем.

После построения майтера проверяется, являются ли эталонная и проверяемая схема эквивалентными, и если нет, то система выдает различающую входную последовательность, которая для комбинационных схем (схем без элементов задержки) состоит из одного входного набора. В системе ABC различающая последовательность строится с использованием SAT [3]. В случае последовательностных схем длина различающей последовательности обычно больше единицы.

Поскольку при построении проверяющих тестов инженеры стараются минимизировать количество тестовых наборов, мы сравнили два способа построения тестовых последовательностей с использованием системы ABC. Первый способ заключается в построении различающей последовательности для исходных проверяемой и эталонной последовательностных схем с помощью команды *dprove*, которая проводит проверку на эквивалентность (Sequential equivalence checking) с использованием последовательностного майтера. Второй способ основан на построении комбинационных эквивалентов длины k [4] для таких схем с последующей проверкой на эквивалентность полученных комбинационных схем.

После проведения серии экспериментов со схемами из базы ISCAS было выявлено, что использование второго способа позволяет получить более короткие различающие последовательности, чем последовательности, полученные с использованием команды *dprove*. В дальнейшем планируется определение классов последовательностных схем, для которых использование метода, основанного на построении комбинационных эквивалентов, при построении тестов может оказаться более эффективным, чем, например, использование широко рекламируемого PDR метода [5].

Литература

1. *Кушик Н., Евтушенко Н., Торгаев С.* Эффективная верификация цифровых компонентов физических систем с использованием мутационного тестирования // Известия вузов. Физика. 2015. Т. 58, № 9. С. 56–60.

2. *ABC:* A System for Sequential Synthesis and Verification : официальный сайт. URL: http://www.eecs.berkeley.edu/~alanmi/abc.

3. *Mishchenko A., Chatterjee S., Jiang R., Brayton R.* FRAIGs: A unifying representation for logic synthesis and verification // ERL Technical Report, EECS Dept. UC Berkeley, 2005.

4. *Скобцов Ю.А., Скобцов В.Ю., Хинди Ш.Н.* Генерация тестов для последовательностных схем с использованием кратной стратегии наблюдения выходных сигналов // Науковий вісник Чернівецького університету. Фізика. Електроніка. 2008. Вип. 423. С. 29–36.

5. *Een N., Mishchenko A., Brayton R.* Efficient implementation of property directed reachability // Formal Methods in Computer-Aided Design (FMCAD). 2011. P. 125–134.

ТЕОРИЯ ПОСТРОЕНИЯ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ КАК ФОРМАЛЬНАЯ ОСНОВА ДЛЯ СОВМЕСТНЫХ ИССЛЕДОВАНИЙ АЛГОРИТМОВ И АРХИТЕКТУР[*]

О.В. Климова

Институт машиноведения УрО РАН, Екатеринбург, Россия
klimova@imach.uran.ru

Параллельная обработка изменила подход к проектированию вычислительных систем. Современной тенденцией при проектировании таких систем стала совместная оптимизация алгоритмов и архитектур [1, 2]. Поднимая реализацию процесса проектирования на более высокий уровень абстракции и расширяя тем самым его возможности, совместные исследования алгоритмов и архитектур позволяют сделать выбор варианта архитектуры обоснованным, базирующимся на анализе и оценке сложности реализации множества вариантов организации вычислений, синтезируемых в процессе таких исследований. Однако для обеспечения возможности их реализации необходимо построить формальные инструменты, способные описать разнообразие параллельных алгоритмов. Такими инструментами могут быть модели, представляющие параметризованное описание внутренней структуры требуемых вычислений. Для построения таких моделей необходимо изучать внутреннюю структуру последовательных алгоритмов и устанавливать законы образования эквивалентных им композиционных форм. Создание таких моделей позволит проектировать многопроцессорные вычислительные системы в едином пространстве, определяемом алгоритмическими и архитектурными параметрами, на основе алгорит-

[*] Настоящая работа выполнена при частичной поддержке программы УрО РАН «Математические модели, алгоритмы, высокопроизводительные вычислительные и информационные технологии и приложения», проект 15-7-1-20.