

## AGAT – СИСТЕМА АВТОМАТИЗИРОВАННОЙ ГЕНЕРАЦИИ ПСЕВДОСЛУЧАЙНЫХ АДАПТИВНЫХ ТЕСТОВ

Зинченко Ю.Е., к.т.н., доцент каф. КИ, Корченко А.А., асп. каф. КИ, Зинченко Т.А., асс. каф. ВМ и П  
Донецкий национальный технический университет

Диагностика цифровых устройств сталкивается все с новыми и новыми задачами, которые появляются в результате развития индустрии. В то же время, появляются новые возможности решения этих задач, которые по разным причинам не были доступны ранее. Одной из таких возможностей является увеличение вычислительных мощностей современной техники, благодаря чему стали доступными ряд методов и подходов к решению классической задачи диагностики – построению теста цифрового устройства. В данной статье рассматривается программный комплекс AGAT, который предназначен для автоматизированной генерации псевдослучайных адаптивных тестов цифровых устройств.

### Структура комплекса AGAT

Программное обеспечение комплекса AGAT состоит из совокупностей модулей, решающих следующие основные задачи (рис. 1):

- 1) генерация псевдослучайных тестов константных неисправностей (КН);
- 2) адаптивное построение теста – накопление статистики о состояниях и переходах схемы для дальнейшего использования полученной информации в процессе адаптивного построения теста;
- 3) моделирование устройства в исправном и неисправном состояниях на основе модели КН;
- 4) интеграция с САПР OrCAD для определения критических состояний устройства для построенного теста;
- 5) конвертация входных и выходных форматов данных;
- 6) пакетная обработка наборов схем.

Основным режимом работы комплекса является построение теста цифрового устройства. Кроме того, поддерживаются такие функции как оценка внешнего теста, отображение активности выводов и списка константных неисправностей на схеме в OrCAD, работа с пакетом схем, сбор статистики о тесте, печать результатов. Взаимодействие с пользователем реализовано при помощи графического интерфейса (рис. 2). Рассмотрим подробнее задачи, которые решают модули AGAT.

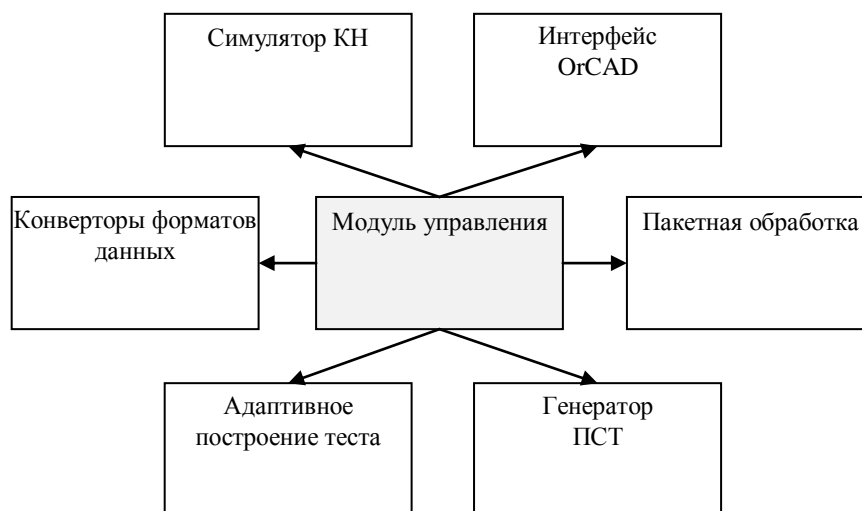


Рисунок 1 – Структура программного комплекса AGAT

*Генерация псевдослучайных тестов* – реализована на основе программной модели сдвигового регистра с линейными обратными связями [1]. Последовательность, полученная таким способом, зависит от выбранного примитивного полинома, а длина цикла этой последовательности равна  $2^n$ , где  $n$  – степень полинома.

*Адаптивный подход генерации псевдослучайных тестов* – авторская разработка [2], основывается на учете истории состояний схемы и переходов между ними. Данный подход позволяет повысить качество построения теста путем адаптации процесса построения теста к текущему графу состояний, а также за счет механизма выхода из тупиковых ветвей.

Процесс работы адаптивного алгоритма отображается при помощи набора графиков (рис. 3).

*Моделирование константных неисправностей* – обнаружение неисправностей тестом базируется на модели константных неисправностей, что позволяет определять наиболее часто встречающиеся неисправности типа обрыв и короткое замыкание.

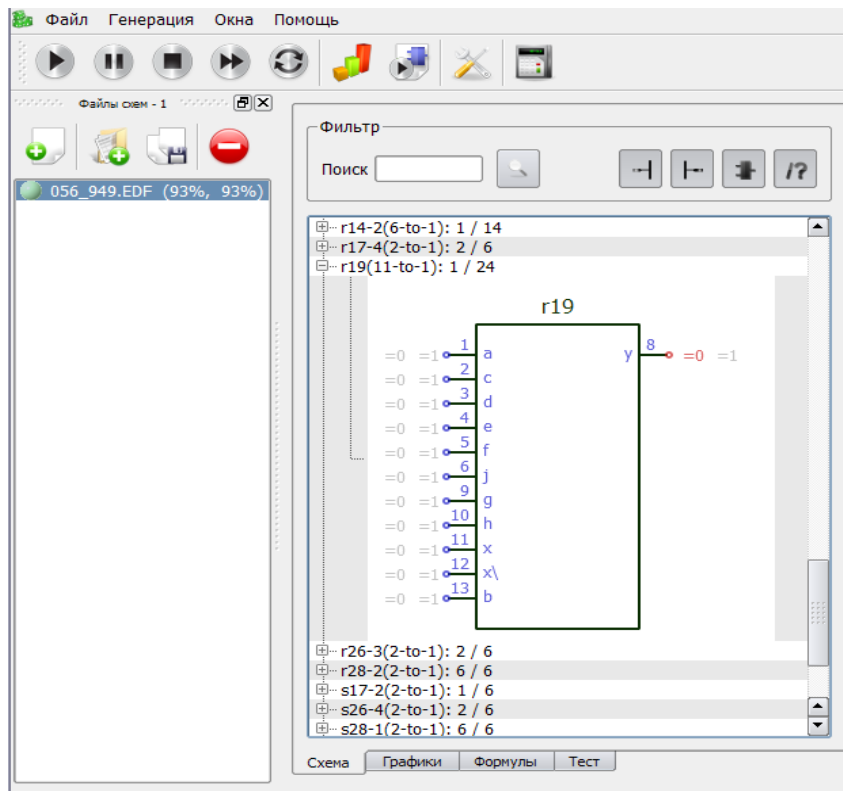


Рисунок 2 – Интерфейс программного комплекса AGAT

*Интеграция с САПР OrCAD* – служит для проверки построенного теста на наличие критических состязаний в схеме. Кроме того, с помощью OrCAD на схеме отображается активность элементов и список неисправностей для текущего теста.

*Конвертация входных и выходных форматов* – выполняет функцию преобразования форматов данных, используемых как внутри комплекса между модулями системы, так и внешних, входных и выходных потоков.

*Пакетная обработка наборов схем* – включает в себя возможность задания индивидуальных или общих параметров, режимы сравнения результатов при различных входных параметрах и другое.

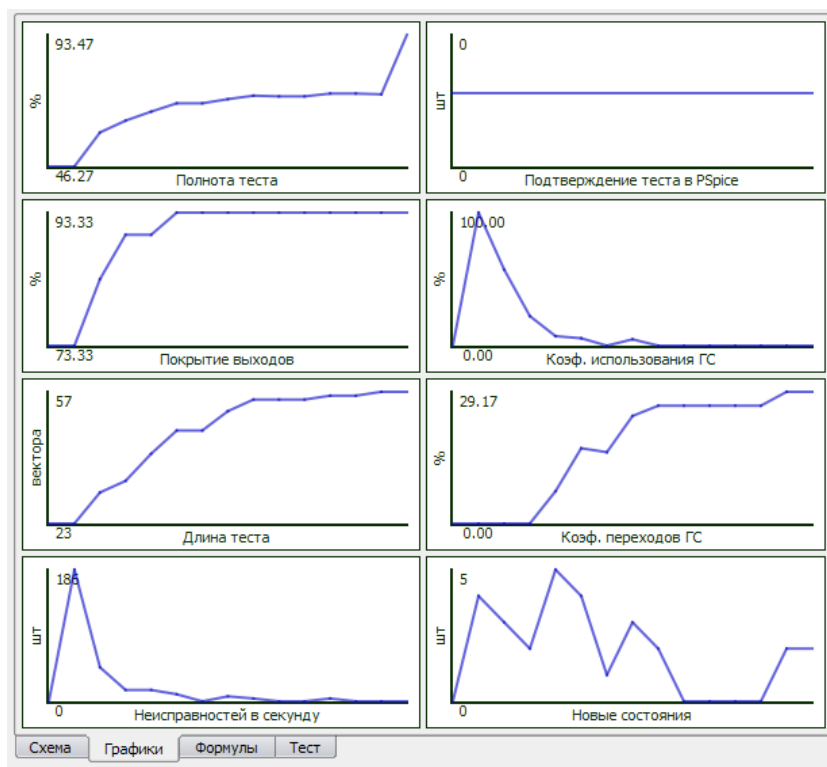


Рисунок 3 – Визуальное отображение хода генерации теста

Кроме того, система ведет статистику процесса построения тестов с возможностью просмотра как в самой программе (рис. 4), так и возможностью экспорта во внешний файл.

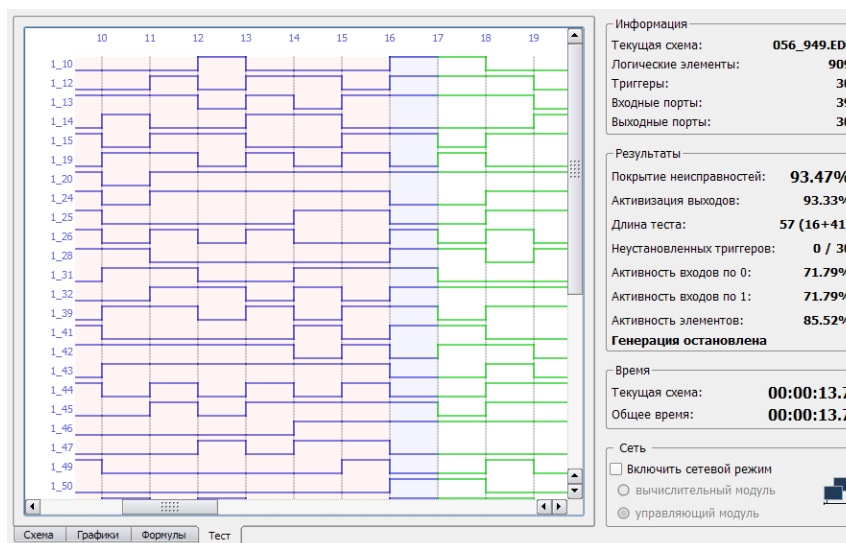


Рисунок 4 – Статистика процесса построения теста

### Эффективность комплекса

Для оценки эффективности построения теста был проведен ряд экспериментов, которые позволили сравнить результаты работы комплекса AGAT с результатами, полученными традиционным алгоритмом построения псевдослучайных тестов.

Экспериментальные исследования проводились для набора ТЭЗ, которые по своей структуре и сложности аналогичны схемам набора ISCAS'89 [3]. Полученные результаты представлены в таблице 1.

Исследования проводились для традиционного псевдослучайного метода и его адаптивной модификации в одинаковых условиях с фиксированным временем генерации теста, равным 2 минутам. Были отобраны такие объекты диагностики, для которых традиционный метод псевдослучайной генерации не давал достаточной полноты покрытия неисправностей. Результаты показали, что во многих случаях адаптивный подход построения теста дает прибавку полноты покрытия теста в пределах 10-30%. Таким образом, эксперимент подтвердил эффективность адаптивного подхода при построении тестов ЦУ.

Таблица 1 - Сравнительные результаты эксперимента

ТЭЗ	Триггеры/ лог. Элем.	Полнота теста		Длина теста	
		A*	T*	A*	T*
1	25 / 355	59%	54%	50	45
2	72 / 503	73%	43%	162	143
3	9 / 307	91%	43%	55	17
4	16 / 796	82%	34%	83	74
5	17 / 206	95%	65%	105	95
6	13 / 267	71%	55%	50	45
7	48 / 238	75%	56%	69	49
8	14 / 63	70%	45%	28	12
9	27 / 190	79%	74%	55	48
10	40 / 386	57%	48%	33	31
11	19 / 142	81%	54%	95	45
12	36 / 255	78%	65%	81	84
13	40 / 504	79%	68%	82	61
14	32 / 650	83%	85%	68	65
15	14 / 601	53%	43%	26	13

\*Примечание: T / A – данные, полученные традиционным/адаптивным ПСТ.

### **Литература:**

1. Rene David. Random testing of digital circuits. – CRC Press, 1998. – 496 p.
2. Зинченко Ю.Е., Корченко А.А. Адаптивный подход к генерации псевдослучайных тестов цифровых устройств // Наукові праці Донецького національного технічного університету. Серія «Проблеми моделювання та автоматизації проектування» (МАП-2011). Випуск 9 (179): – Донецьк: ДонНТУ. – 2011. – С. 360-365.
3. Brgles F. Combinational profiles of sequential benchmark circuits / F. Brgles, D. Bryan, K. Kozminski // International symposium of circuits and systems, ISCAS-89. – 1989.– p. 1929-1934.