

УДК 681.32

В. Ю. СКОБЦОВ*Институт прикладной математики и механики НАН Украины, Украина*

ХАРАКТЕРИСТИЧЕСКИЕ ПОСЛЕДОВАТЕЛЬНОСТИ В КОНТЕКСТЕ ТЕСТОВОЙ ДИАГНОСТИКИ УСТРОЙСТВ С ПАМЯТЬЮ

В работе рассматривается задача построения характеристических последовательностей цифровых устройств с памятью в контексте их использования при генерации проверяющих тестов таких устройств на структурном уровне. Предложены генетические алгоритмы построения характеристических последовательностей установка/сброса триггера и различающих последовательностей, которые позволяют устанавливать/сбрасывать в исправной или неисправной схемах некоторые элементы памяти или различать их. Построенные характеристические последовательности используются в генетическом алгоритме построения тестовых последовательностей верхнего уровня. Предложенные алгоритмы имеют самостоятельную ценность как эффективный инструмент анализа поведения цифровых устройств с памятью на структурном уровне.

Ключевые слова: последовательностные схемы, характеристические последовательности, построение тестов, генетические алгоритмы

Актуальность

Проблема построения тестов для цифровых схем с памятью является актуальной и требует значительных вычислительных ресурсов. Применение традиционных детерминированных алгоритмов, в данном случае зачастую терпит фиаско в силу их высокой вычислительной сложности. В последние декады в основе многих успешных решений в области тестовой диагностики цифровых схем с памятью лежит применение методов эволюционных вычислений. В настоящее время при генерации тестов также широко применяются параллельные генетические алгоритмы (ГА), которые позволяют существенно повысить быстродействие [1, 2].

При построении тестовых последовательностей для схем с памятью большое значение имеют свойства схемы устанавливаться в некоторые определенные состояния. Для схем, обладающих этими свойствами даже частично, задача построения тестов решается гораздо проще. Причина этого заключается в том, что в этом случае нам необходимо различить не все возможные пары состояний исправной и неисправной схем, а только их небольшую часть. Поэтому имеет смысл заранее исследовать возможности схемы устанавливаться в определенные состояния, найти соответствующие входные последовательности, позволяющие это сделать, и далее использовать их в процессе построения тестов. Построению таких входных последовательностей всегда уделялось большое внимание [1, 3-5].

Для сильно последовательностных схем, кото-

рые не имеют специальных схем сброса, целесообразно применять 2-уровневый ГА. При этом ГА первого нижнего уровня строит характеристические последовательности, которые позволяют устанавливать в исправной или неисправной схемах некоторые элементы памяти (переменные состояния) в определенные состояния или различать их. Генетический алгоритм верхнего уровня при генерации тестов как структурные блоки использует характеристические последовательности, построенные на нижнем уровне ГА, что делает эволюционный поиск более "направленным" и повышает его эффективность. Такой 2-уровневый генетический алгоритм, использующий 16-значное моделирование и стратегию кратного наблюдения, был предложен авторами в работе [6].

Целью данной работы является дальнейшее развитие эволюционных алгоритмов построения характеристических и идентифицирующих последовательностей для схем с памятью.

1. Характеристические последовательности

Построение характеристических последовательностей может быть выполнено на разных уровнях представления цифровых устройств:

- 1) на функциональном уровне на основе автоматной модели;
- 2) на структурном уровне представления устройств с применением структурной модели в виде логической схемы.

В обоих случаях для их построения можно использовать эволюционные алгоритмы. В первом случае используются непосредственно таблицы переходов и выходов автоматов и фитнес-функции, в которых подсчитывается число различаемых состояний. Во втором случае характеристические последовательности генерируются на основании многозначного логического моделирования и фитнес-функций, в которых подсчитывается число отличных сигналов на выходах вентилей и триггеров на основании данных логического моделирования. Мы будем применять второй подход.

При этом используются генетические алгоритмы уровня, где в качестве особей выступают входные последовательности, задаваемые двоичными таблицами [1,2]. Применение многозначного моделирования в процессе их генерации позволяет повысить эффективность.

Для построения тестов (на верхнем уровне) полезными являются следующие входные характеристические последовательности [1, 2, 7].

1) S-последовательности. Назовем входную последовательность S_i -последовательностью i -го триггера, если она устанавливает i -й триггер в единичное состояние из полностью неопределенного состояния.

2) R-последовательности. Назовем входную последовательность R_i -последовательностью i -го триггера, если она сбрасывает i -й триггер в нулевое состояние из полностью неопределенного состояния.

3) Различающие D-последовательности. Различающей D_i -последовательностью для i -го триггера назовем входную последовательность, которая производит различные выходные реакции исправной схемы для двух различных состояний, отличающихся значением сигнала на i -ом триггере при неопределенных значениях остальных триггеров.

Обычно длина этих входных последовательностей ограничивается величиной $4d$, где d – структурная последовательностная глубина [1].

Следует отметить, что условие неопределенных значений для остальных триггеров является достаточно жестким, и не для всех триггеров могут быть найдены различающие входные последовательности.

Например, входная последовательность $x_{11} = 0$, $x_{21} = 0$ является R-последовательностью для схемы рис. 1, поскольку переводит схемы из полностью неопределенного состояния $y = u$ в состояние $y = 0$.

Кроме этого, последовательность $x_{11} = 0$, $x_{21} = 0$; $x_{12} = 0$, $x_{22} = 1$ является S-последовательностью для этой же схемы, поскольку устанавливает

ее из произвольного начального состояния $y = u$ в единичное состояние $y = 1$.

Аналогично входная последовательность $x_{11} = 1$, $x_{21} = 0$ является различающей D-последовательностью переменной состояния y для схемы рис. 1.

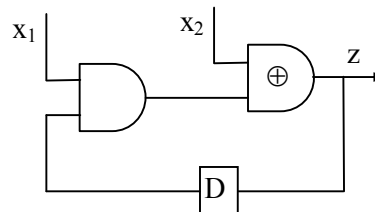


Рис. 1. Пример последовательностной схемы

Очевидно, что при включении характеристических последовательностей в популяцию вероятность построения необходимой входной последовательности из характеристических последовательностей выше, чем из случайных. Рассмотрим ситуацию, представленную на рис.2, где необходимо построить входную последовательность, устанавливающую триггеры в состояние $S_3 = (10uuuuuu)$. Допустим, что существует S-последовательность T_1 , которая как показано на рис.2 устанавливает первый триггер в единичное состояние $S_1 = (1uuuuuuu)$. С другой стороны, есть R-последовательность T_2 , позволяющая из текущего состояния сбросить в нулевое состояние второй триггер $S_2 = (u0uuuuuu)$. Поэтому последовательности T_1 и T_2 включаются в популяцию потенциальных решений и могут в результате применения генетических операторов существенно ускорить построение последовательности T_3 , которая устанавливает триггеры в необходимое состояние из текущего состояния, как это показано на рис. 2. Отметим, что наличие неисправности может помешать установить триггеры в необходимое состояние, поэтому в ГА построения характеристических последовательностей следует использовать логическое моделирование с неисправностями, решающее эту проблему.

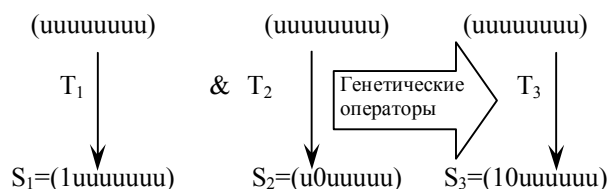


Рис. 2. Использование характеристических последовательностей для установки требуемого состояния

Аналогично, следует использовать различающие D-последовательности, что показано на рис.3,

где необходимо построить входную последовательность T , различающую пару состояний $s_3=(uuDuuD'uu)$. Допустим, что существует D_3 -последовательность T_1 , которая как показано на рис.3, различает состояния 3-го триггера $T_1=(uuDuuuuu)$. С другой стороны, есть D_6 -последовательность T_2 , различающая состояния 6-го триггера $T_2=(uuuuuD'uu)$. Поэтому последовательности T_1 и T_2 желательно включить в популяцию потенциальных решений, где они могут в результате применения генетических операторов существенно ускорить построение последовательности T_3 , которая устанавливает триггера в необходимое состояние из текущего состояния, как это показано на рис.3.

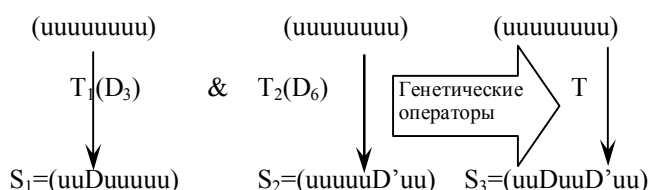


Рис. 3. Использование различающих D-последовательностей для различения пар состояний

Следует отметить, что при построении тестовой последовательности в алгоритме верхнего уровня, во-первых, важную роль играет выбор следующей пары состояний исправной и неисправной схем в процессе построения тестовой последовательности. Во-вторых, в процессе построения различающей последовательности для данной пары состояний с помощью генетического алгоритма следует внести в популяцию следующие характеристические последовательности:

- для переменных состояний, имеющих различные значения в исправной и неисправной схемах (D или D') – различающие последовательности;
- для переменных состояний, которые имеют одинаковые значения 0 или 1 в исправной и неисправной схемах – R-последовательности и S-последовательности соответственно.

2. Генетический алгоритм построения характеристических S- и R-последовательностей

Построение $S_i(R_i)$ -последовательности, устанавливающей i -й триггер в единичное состояние $y_i = 1(0)$, длины T сводится к поиску таких значений x^1, x^2, \dots, x^T при которых переменная состояния принимает значение $y_i^{T+1}(Y^1, X^1, X^2, \dots, X^T) = 1(0)$ независимо от

значений Y^1 начального состояния. Здесь и далее в скобках приводится вариант значения для случая (R_i)-последовательности. Условием независимости булевой функции $f(X, Y)$ от переменных $Y = (y_1, y_2, \dots, y_m)$ является равенство $\frac{\Delta f}{\Delta Y} = 0$.

Поэтому построение S_i -последовательности длины T сводится к решению системы уравнений:

$$\begin{cases} y_i = 1(0), \\ \frac{\Delta y_i^{T+1}}{\Delta Y^1} = 0, \end{cases} \quad (1)$$

где $\frac{\Delta f}{\Delta Y} = \bigvee_{E_1=1}^{2^p} \frac{d|E_1|f}{dX_1(E_1)}$ – булев дифференциальный

Δ -оператор [1]. Например, для двух переменных он имеет следующий вид $\frac{\Delta f}{\Delta x_1 \Delta x_2} = \frac{\Delta f}{\Delta x_1} \vee \frac{\Delta f}{\Delta x_2} \vee \frac{\Delta^2 f}{\Delta x_1 \Delta x_2}$.

Здесь $\frac{df}{dx_i} = f(x_i = 0) \oplus f(x_i = 1)$ – булева производная булевой функции $f(X)$ по переменной x_i [1, 10].

Поскольку длина S-последовательности заранее не известна, то необходимо последовательно искать решения системы (1) для $T=1, 2, \dots, N$, где верхняя граница $N = 2^m(2^m - 1) / 2$ [1]. При этом на каждом i -м шаге используются результаты $(i-1)$ -го шага. В [1] изложен алгоритм построения минимальной инициализирующей входной последовательности, которая позволяет установить все триггеры в определенные состояния независимо от начального значения переменных состояний. Мы используем этот подход к поиску S-последовательностей и R-последовательностей с помощью генетического алгоритма с использованием многозначного логического моделирования.

Разработан генетический алгоритм построения характеристических последовательностей S-и R-типа. Ниже приводится псевдокод ГА построения S(R)-последовательности.

ГА построения S_i -последовательности (схема, номер триггера i)

```
{
  // подготовка начальной популяции
  Формирование начальной популяции  $S_i$ -последовательностей (схема);
  16-значное логическое моделирование в  $B_{16}$  (схема, начальная популяция);
  Оценка особей (начальная популяция);
  Номер популяции  $N=0$ ;
  While (( $y_i \neq 1(0)$ )  $\wedge$  ( $N \leq T$ ))
  // не достигнут критерий остановки
  {
    for ( $i=0$ ;  $i <$  мощность_популяции;  $i++$ )
    {
      Операция Выбора (родитель A, родитель B);
```

```

if(rand() < Pc)
    Операція Скрещивання(родитель А, роди-
    тель В, потомокА, потомок В);
if(rand() < Pm)
    {Операція Мутації (потомокА);
    Операція Мутації (потомокВ);}
Добавить_в_промежуточную
популяцію(потомокА, потомок В);
}
Логическое моделирование в В16(схема,
промежуточная популяция);
Оценка особей(промежуточная популяция);
Генерация новой популяцію(мощность
популяції);
Номер Популяції ++;
}
Формирование Si- последовательностей;
}

```

В приведённом алгоритме используется пред-
ставление особей – входных последовательностей в
виде двоичных таблиц [1,2], и представленные в [1,2]
генетические операторы кроссинговера и мутации.

В предложенном алгоритме вычисляется зна-
чение фитнес-функции каждой особи в популяции,
которое основано на результатах исправного 16-
значного логического моделирования в многознач-
ном алфавите В₁₆[1]. Естественно, свойства входных
последовательностей, которые генерируются гене-
тическим алгоритмом, зависят в первую очередь от
вида используемой фитнес-функции [1,2]. По-
скольку генетический алгоритм генерирует S(R)-
последовательности, которые устанавливают неко-
торые триггеры в единичные (нулевые) состояния,
то в фитнес-функции используются следующие
данные результатов логического моделирования в
многозначном алфавите:

- n_1 – число установленных в 1(0) триггеров –
наша основная цель;
- активность схемы или число событий моде-
лирования: n_2 – активность на выходах триггеров, n_3
– активность на выходах вентилях; чем выше число
событий при моделировании схемы на заданной
последовательности, тем выше вероятность того,
что ещё неустановившиеся триггеры перейдут в
определённое состояние;
- L_T – длина последовательности – из двух
последовательностей при прочих равных условиях
необходимо предпочесть более короткую.

Таким образом, фитнес- функция имеет сле-
дующий вид:

$$f(s) = c_L L_T \sum_{i=1}^3 c_i n_i, \quad (2)$$

где n_i – описанные выше параметры, c_1, c_2, c_3, c_4 –
нормализующие константы, которые имеют сле-
дующий смысл:

- c_1, c_2, c_3 уравнивают влияние на фит-
несс-функцию числа триггеров, перешедших в из-
вестное состояние, и активность вентилях в схеме;

- c_L позволяет искать более короткие входные
последовательности; если данную константу вы-
брать близкой к единице $c_L < 1$, то из двух входных
последовательностей при прочих равных условиях
лучшее значение фитнес-функции будет иметь та
последовательность, длина которой меньше.

Псевдокод процедуры вычисления фитнес-
функции одной особи (входной последовательно-
сти) приведён ниже.

```

Оценка особи(входная последовательность, длина)
{
    16-значное моделирование в В16(схема,входная
    последовательность);
    n1=Число установленных в 1(0)триггеров;
    n2=Число событий на выходах триггеров;
    n3= Число событий на выходах вентилях;
    LT//Длина последовательности;
    Вычисление_фитнесс_функции f по формуле (2);
}

```

3. Построение различающих входных D-последовательностей

Описанный в предыдущем разделе алгоритм с
небольшими модификациями можно использовать и
при построении различающих входных D-
последовательностей, которые производят различ-
ные выходные реакции исправной схемы для двух
различных состояний, отличающихся значением
сигнала на i-ом триггере при неопределённых зна-
чениях остальных триггеров, как это показано ниже
на следующем примере

uuuuDuuuuuuD'uuu .

При этом, в начальном состоянии одна пере-
менная состояния $y_i = D$ или $y_i = D'$, а остальные
имеют неопределённые значения $y_j = u$ для $j \neq i$.
Фактически, поиск различающей входной последо-
вательности для i-ой переменной состояния сводит-
ся к поиску таких значений x^1, x^2, \dots, x^T , при
которых некоторая k-я переменная состояния
 $y_k^{T+1}(Y^1, X^1, X^2, \dots, X^T)$ зависит от значения y_i на-
чального состояния.

Известно, что условием зависимости булевой
функции $f(x, y)$ от переменной y_i является равен-
ство булевой производной $\frac{df}{dy_i} = 1$ [1, 10]. Поэтому

построение различающей последовательности для
переменной состояния длины T сводится к решению
уравнения

$$\frac{dy_k^{T+1}(Y^1, X^1, X^2, \dots, X^T)}{dy_i} = 1. \quad (3)$$

Для решения этого уравнения можно использовать генетический алгоритм, приведенный в предыдущем разделе со следующими модификациями. Основное отличие данного ГА от выше представленного в процедуре вычисления значений фитнес-функции каждой особи в популяции. Как и в предыдущем ГА, для оценки значений фитнес-функции используются данные логического моделирования в многозначном алфавите B_{16} . Логическое моделирование в B_{16} для особи (потенциального решения) использует неопределенные значения переменных состояний $y_i = u$ для всех $j \neq i$ и $y_i = D$ или $y_i = D'$ для исследуемой переменной. При оценке качества последовательности в фитнес-функции используются следующие данные результатов логического моделирования в многозначном алфавите:

- n_4, n_5 – число значений D или D' на выходах вентилях и триггерах схемы в процессе моделирования – чем больше таких значений, тем выше вероятность их распространения на внешние выходы схемы, что требуется для различающей последовательности;

- L_T – длина последовательности – выбирается более короткая особь.

Таким образом, фитнес- функция для одного набора последовательности имеет следующий вид:

$$f(s) = c_L L_T \sum_{i=4}^5 c_i n_i, \quad (4)$$

где c_4 и c_5 – как и ранее константы нормирования.

Значение фитнес-функции для всей последовательности вычисляется из оценочных функций всех её входных наборов s_i (строк двоичной таблицы) по формуле:

$$H(\tilde{T}, s) = \sum_{i=1}^{\text{длина посл.}} L^i f(s_i), \quad (5)$$

где \tilde{T} – анализируемая последовательность;

s_i – набор из последовательности,

i – позиция набора в последовательности,

f – заданная неисправность,

L – предварительно заданная константа в диапазоне $(0; 1]$, благодаря которой предпочтение отдаётся более коротким последовательностям.

Построенные на нижнем уровне характеристические последовательности используются далее на верхнем уровне для различения пар состояний при

генерации проверяющей последовательности для данной неисправности [7].

Выводы

В работе предложены генетические алгоритмы построения характеристических последовательностей: установка/сброса триггера – S/R-последовательностей, различающих D-последовательности, которые позволяют устанавливать/сбрасывать в исправной или неисправной схемах некоторые элементы памяти (переменные состояния) в определенные состояния или различать их. Построенные характеристические последовательности используются в ГА построения тестовых последовательностей верхнего уровня [7] для последовательностных схем.

Предложенные алгоритмы имеют самостоятельную ценность как эффективный инструмент анализа поведения цифровых устройств с памятью на структурном уровне.

Литература

1. Скобцов, Ю. А. Моделирование, тестирование и диагностика цифровых устройств [Текст] / Ю. А. Скобцов, Д. В. Сперанский, В. Ю. Скобцов. – Москва : Национальный открытый университет «ИНТУИТ», 2012. – 439 с.
2. Skobtsov, Y. A. Evolutionary test generation methods for digital devices [Text] / Y. A. Skobtsov, V. Y. Skobtsov // In Design of Digital Systems and Devices (Eds. Marian Adamski, Alexander Barkalov, and Marek Wegrzyn). – Springer-Verlag Berlin Heidelberg, 2011. – Lecture Notes in Electrical Engineering. – 2011. – Vol. 79. – P. 331 – 361.
3. Pixley, C. Exact Calculation of Synchronization Sequences based on Binary Decision Diagrams [Text] / C. Pixley, S. Jeong, G. Hatchel // IEEE Trans. on CAD. – 1994. – Vol. 13. – P. 1024 – 1034.
4. Rho, J. –K. Minimum Length Synchronizing Sequences of Finite State Machine [Text] / J. –K. Rho, F. Somenzi, C. Pixley // Proc. ACM Design Automation Conf., 1993. – P. 463 – 468.
5. Wehbeh, J. A. Initialization of Sequential Circuits and its Application to ATPG [Text] / J. A. Wehbeh, D. G. Saab // Proc. IEEE VLSI Test Symp., 1996. – P. 246 – 251.
6. Скобцов, Ю. О. Двухуровневый алгоритм генерации проверяющих тестов для схем с памятью [Текст] / Ю. О. Скобцов, В. Ю. Скобцов, Ш. Н. Хинди // Радиоелектронні і комп'ютерні системи. – 2009. – № 7. – С. 136 – 140.

Поступила в редакцию 26.02.2014, рассмотрена на редколлегии 24.03.2014

Рецензент: д-р техн. наук, проф. Н. И. Чичикало, Донецкий национальный технический университет, Донецк, Украина.

ХАРАКТЕРИСТИЧНІ ПОСЛІДОВНОСТІ В КОНТЕКСТІ ТЕСТОВОЇ ДІАГНОСТИКИ ПРИБОРІВ З ПАМ'ЯТТЮ

В. Ю. Скобцов

У роботі розглядається задача побудови характеристичних послідовностей цифрових пристроїв з пам'яттю в контексті їх використання при генерації перевіряючих тестів таких пристроїв на структурному рівні. Запропоновано генетичні алгоритми побудови характеристичних послідовностей устанавлення/ скидання тригера і розрізняючих послідовностей, які дозволяють встановлювати/скидати у справній або несправній схемах деякі елементи пам'яті або розрізнити їх. Побудовані характеристичні послідовності використовуються в генетичному алгоритмі побудови тестових послідовностей верхнього рівня. Запропоновані алгоритми мають самостійну цінність як ефективний інструмент аналізу поведінки цифрових пристроїв з пам'яттю на структурному рівні.

Ключові слова: послідовнісні схеми, характеристичні послідовності, побудова тестів, генетичні алгоритми.

CHARACTERISTIC SEQUENCES IN TEST DIAGNOSTICS CONTEXT OF DEVICES WITH MEMORY

V. Y. Skobtsov

In this paper we consider the problem of the characteristic sequences generation for digital devices with memory in the context of their use in checking tests generating of such devices at the structural level. There were proposed genetic algorithms for generating characteristic sequences of set/reset of flip-flop and distinguishing sequences that allow you to set/reset of some memory elements in fault-free or faulty circuits or to distinguish them. Generated characteristic sequences are used in the genetic test generation algorithm of top-level. The proposed algorithms have an intrinsic value as an effective tool for analyzing the behavior of digital devices with memory at the structural level.

Key words: sequential circuits, characteristic sequences, test generation, genetic algorithms.

Скобцов Вадим Юрьевич – канд. техн. наук, доцент, зав. лабораторией дискретной математики и прикладной алгебры Института прикладной математики и механики НАН Украины, Донецк, Украина.