

О СИСТЕМЕ ПРОГРАММ МОДЕЛИРОВАНИЯ ДИНАМИЧЕСКОГО ПОВЕДЕНИЯ ПЕРЕКЛЮЧАТЕЛЬНЫХ СХЕМ С ЗАДАВАЕМОЙ ТОЧНОСТЬЮ

Работа выполнена при финансовой поддержке РФФИ, грант 98-01-00288.

Рассматривается задача моделирования и анализа динамического поведения БИС на транзисторном уровне представления. Моделью БИС служит переключательная схема на полурешетках проводимости и состояний. Описываются алгоритмы и программы имитационного моделирования и анализа схемы с любой наперед заданной точностью.

1. Основные понятия

Переключательная схема служит моделью больших интегральных схем (БИС) и определяется в [1] как тройка конечных множеств (X, Z, Z_0) , где X – множество элементов схемы (которые, в свою очередь, являются схемами); $Z_0 \subseteq Z$ – множество полюсов схемы (среди которых выделяются полюсы источника питания VDD и GND и входные полюсы, через которые на схему подаются воздействия извне); Z – множество узлов схемы, представляющее собой разбиение на множестве полюсов всех ее элементов. Таким образом, схемы имеют иерархическую структуру, на самом нижнем уровне которой находятся базисные элементы – транзисторы различных типов, резистор. Транзистор представляет собой элемент с управляемой проводимостью и задается монотонной функцией проводимости $p=f(s)$; здесь s – состояние затвора транзистора; p – проводимость между его истоком и стоком, $p \in P$, $s \in S$; P и S – полурешетки проводимостей и состояний соответственно. В общем случае $P=\{0, 1, X, 0', 1', X', E\}$; здесь $0, 1, X$ – точки полурешетки, представляющие собой проводимости соответственно разомкнутой, замкнутой и резистивной цепи; остальные элементы – в подходящей степени неопределенные значения: $0'=1+X$, $1'=0+X$, $X'=0+1$, $E=0+1+X$ (полная неопределенность). Для некоторых классов схем достаточно рассматривать полурешетку $P' \subseteq P$; например, для КМОП-схем $P'=\{0, 1, X'\}$.

Состоянием узла является пара проводимостей от этого узла до полюсов источника питания, т.е. в общем случае $S=P^2$. При этом $|S|=|P|^2=49$. Однако в ряде случаев нет необходимости различать все состояния; например, те из них, для которых совпадают значения функций проводимости всех базисных элементов, физически эквивалентны. Информация о неразличимых состояниях задается в виде эквивалентности R на множестве точек полурешетки P^2 , т.е. на множестве $\{00, 0X, 01, X0, XX, X1, 10, 1X, 11\}$. Затем по эквивалентности R строится полурешетка $\langle R \rangle$, элементами которой являются интервалы, совпадающие с объединениями некоторых смежных классов эквивалентности R . $\langle R \rangle$ является адекватной моделью P^2 (по теореме 1.4 в [1]), и $|\langle R \rangle| \leq |P^2|$. Замена полурешетки P^2 на $\langle R \rangle$ позволяет понизить мощность алфавита состояний без потери адекватности моделирования; R при этом играет роль точности модели. Например, для R , заданной классами $\{00\}$, $\{X0, 10\}$, $\{11, 1X\}$, $\{01, X1\}$, $\{0X, XX\}$, получается следующая полурешетка $\langle R \rangle$: $\{00,$

$0'0, 10', 1'1, 1'X, E0, 1'0', E0', EE\}$. Резистор представляет собой элемент с двумя полюсами и постоянной проводимостью между ними, равной X .

Назовем полным состоянием схемы набор состояний всех ее узлов, а входным состоянием – набор состояний всех входных полюсов. Так как каждое полное состояние содержит в себе и входное состояние, то будем обозначать: $r=aq$; r – полное состояние, a – соответствующее ему входное состояние, q – набор состояний всех остальных (не являющихся входными полюсами) узлов. Полное состояние r называется устойчивым, если оно соответствует вычисленным по нему полным проводимостям в схеме, т.е. при неизменных состояниях входных полюсов схема остается в устойчивом состоянии сколь угодно долго. Назовем точкой монотонности схемы пару (b, r) , где b – входное состояние; $r=aq$ – полное устойчивое состояние, если $a \leq b$ или $b \leq a$; $\leq$ – отношение порядка в полурешетке S .

Под динамическим поведением переключательной схемы будем понимать процесс изменения ее полного состояния при асинхронном изменении состояний входных полюсов. Пусть схема находится в полном устойчивом состоянии $r=aq$, и входное состояние a меняется на b . Это изменение описывается промежуточным входным состоянием $a+b$, где $+$ – сложение в полурешетке S . В схеме при этом возникает переходный процесс, в результате которого она переходит в полное устойчивое состояние t . По завершении переходного процесса (при установившемся входном состоянии b) схема переходит в новое устойчивое состояние s . Пару состояний (t, s) будем называть динамическим переходом из состояния $r=aq$ под воздействием входного состояния b , и обозначать $\Psi(b, aq)=(t, s)$.

2. Решаемые задачи

Решаются задачи моделирования и анализа динамического поведения переключательной схемы.

Задача моделирования ставится так: для заданной переключательной схемы C , начального полного устойчивого состояния $r=aq$ и теста (последовательности входных состояний) $B=(b_1, \dots, b_m)$ найти последовательность полных устойчивых состояний $(s_1, \dots, s_m)$, в которые переводят схему наборы теста. Компоненты s_i должны принимать значение в полурешетке S , т.е. задача решается с предельной (при $S=P^2$) или с заданной ($S=\langle R \rangle$) точностью.

Задача анализа динамического поведения схемы состоит в построении таблицы ее динамических переходов.

3. Описание алгоритмов

Для адекватного моделирования поведения схемы последнюю необходимо представить в виде одноуровневой (не иерархической) структуры. Для этого выполняется операция подстановки в схему C схем всех ее элементов, после чего C представляется матрицей $G=||g_{ij}||$, строки и столбцы которой сопоставлены узлам схемы, а $g_{ij}: S^k \rightarrow P$ есть функция непосредственной проводимости от i -го к j -му узлу схемы. Здесь k — число управляющих (отождествленных с затворами транзисторов) узлов схемы, а g_{ij} строится как дизъюнкция функций проводимости всех базисных элементов, полюсы которых отождествлены с i -м и j -м узлами.

Для каждого входного состояния b теста B выполняется два шага моделирования:

(1) для исходного состояния $r=aq$ и промежуточного входного состояния $a+b$ вычисляется полное устойчивое состояние t , $r \leq t$;

(2) для полного состояния t и входного состояния b вычисляется полное устойчивое состояние s , $s \leq t$.

Полученная пара состояний является динамическим переходом схемы: $\Psi(b, aq)=(t, s)$; состояние s заносится в результат и становится исходным для моделирования следующего состояния теста.

Собственно шаг моделирования входного состояния b из полного состояния r состоит в следующем:

1) вычисляются непосредственные проводимости между всеми узлами схемы, зависящие от текущего состояния r ;

2) так как в алгебре проводимостей не выполняются законы дистрибутивности (а это необходимо для следующего шага), то схема сводится к дистрибутивной путем отождествления узлов, проводимости между которыми равны 1;

3) вычисляются полные проводимости от всех узлов схемы до полюсов источника питания (алгоритм имитационного моделирования переключательных сетей [1]) — тем самым определяется новое полное состояние r' ;

4) для входных полюсов вычисляется функция узлового соединения $a_i' = a_i \nabla b_i$, представляющая собой покомпонентную дизъюнкцию проводимостей в состояниях a_i' и b_i ;

5) если $r' = r$, то достигнуто устойчивое состояние схемы, и процесс завершается; иначе $r := r'$, переход к п.1.

Так как функции всех элементов схемы монотонны и пары $(a+b, r)$ и (b, t) являются точками монотонности схемы, то шаги (1) и (2) всегда выполнимы, т.е. приводят схему в устойчивое состояние за конечное число итераций.

В схеме могут быть выделены синхронные входы, состояния которых заменяются на b_i уже на шаге (1). В этом случае возможен (из-за нарушения монотонности) бесконечный цикл изменения состояния схемы, т.е. существование решения не гарантируется. Однако если устойчивое состояние все же достигается, то оно обладает большей определенностью, чем получаемое при асинхронных входах.

Алгоритм вычисления полной проводимости в схеме C между узлами i и j состоит в приписывании всем

узлам a схемы символа $s_a \in P$ — проводимости от i к a — при помощи следующей итеративной процедуры:

1) $s_i := 1$, $s_a := 0$ для всех $a \neq i$;

2) если для некоторой пары узлов (a, b) , где $a \neq j$, имеет место свойство

$$s_b \neq s_a \vee (s_a \wedge p_{ab}), \quad (*)$$

(здесь p_{ab} — непосредственная проводимость схемы между узлами a и b), то узлу b приписывается новый символ s ;

3) шаг 2 повторяется до тех пор, пока в схеме не найдется хотя бы одна пара узлов со свойством (*).

После этого s_j есть полная проводимость схемы между узлами i и j . Если $i=GND$, $j=VDD$, то в результате этой процедуры будут вычислены проводимости от всех узлов до полюса GND , а если $i=VDD$, $j=GND$ — от всех узлов до полюса VDD . Сходимость алгоритма доказывается в [1] на основе конечности полурешетки P .

Алгоритм построения таблицы динамических переходов схемы:

1. Перечисляются всевозможные входные наборы $b_1, \dots, b_m$, $m = |S|^n$, n — число входных полюсов.

2. Определяется устойчивое состояние s_0 (например, выполнением шага моделирования для полностью неопределенных полного и входного состояний).

3. Строятся множества состояний $Q = \emptyset$, $Q' = \{s_0\}$.

4. Для всех пар (b_j, s_i) , $j=1, \dots, m$, $s_i \in Q'$, вычисляются (путем выполнения двух шагов моделирования) динамические переходы $(t_{ij}, s_{ij}) = \Psi(b_j, s_i)$.

5. $Q := Q \cup Q'$, $Q' := \emptyset$.

6. В Q' добавляются те из состояний t_{ij} , s_{ij} , которые не содержатся в Q .

7. Шаги 4 – 6 повторяются до тех пор, пока в Q' появляются новые состояния.

4. Описание программ

Алгоритмы моделирования реализованы программно на языке C++. Исходными данными для программы служат текстовые файлы, содержащие описание схемы и описание теста. Схема описывается на языке SDL [2]. Обязательно присутствие раздела INPUTS, содержащего имена входных полюсов, и хотя бы одного из разделов NETS (описание узлов схемы) и COMPS (описание связей элементов схемы). При трансляции описания схемы во внутреннюю форму производится синтаксический контроль описания. При наличии в описании разделов NETS и COMPS одновременно проверяется соответствие их друг другу. В качестве примера приведем описание схемы логического вентиля ИЛИ-НЕ с двумя входами:

```
NAME: NOR-2; INPUTS: X1,X2;OUTPUTS: Y;
PMOS: T11,T12; NMOS: T21,T22; COMPS;
T11=1*N3, 2*N5, 3*N1; T12=1*N3, 2*N5, 3*N2;
T21=1*N5, 2*N6, 3*N1; T22=1*N6, 2*N4, 3*N2;
X1=N1; X2=N2; Y=N5; VDD=N4; GND=N3;NETS;
N1=X1, T11.3, T21.3; N2=X2, T12.3, T22.3;
N3=GND, T11.1,T12.1; N4=VDD, T22.2; N5=Y,
T21.1, T12.2, T11.2; N6=T21.2, T22.1;ENDC.
```

Последовательность входных состояний (тест) задается уравнениями вида: <имя_входного_полюса>= $a, b, \dots$, где $a, b, \dots$ принимают значения из S . Возможна со-

крашенная запись теста уравнением вида: <имя_входного_полюса>=[$a, b, \dots$]/ $i, \dots$, где i – натуральное число, означающее, что группа значений $a, b, \dots$ повторяется i раз. Если значение переменной не задано, то полагается равным 00. Уравнения разделяются символом ';'. Пример задания теста для схемы NOR_2: $X1=[10]/2, [01]/2; X2=[10,01]/2$.

Может быть задано также начальное состояние, которое проверяется на устойчивость. Если начальное состояние не задано, то оно вычисляется с помощью шага моделирования для полностью неопределенных полного и входного состояний.

Начальное состояние задается уравнениями вида: <имя_узла или имя_полюса>= a ; где $a \in S$. Если уравнение для узла не задано, то его начальное состояние полагается равным EE (кроме полюсов GND и VDD , состояния которых равны 10 и 01 соответственно). Уравнения с одинаковой правой частью можно объединить в одну цепочку равенств.

Пример задания начального состояния для схемы NOR_2: $N6=N5=00; X2=N1=10$.

Кроме того, могут быть заданы также список синхронных входов и эквивалентность R на множестве точек полурешетки P^2 . Результатом работы программы является последовательность полных устойчивых состояний со значениями в полурешетке S , в которые переводят схему наборы теста.

Основные блоки программы:

- трансляция во внутреннее представление описаний схемы, теста и начального состояния с синтаксическим и семантическим контролем (программа написана при участии А. Лесных);
- проверка начального состояния на устойчивость;
- подстановка в схему схем всех ее элементов;
- построение по эквивалентности R полурешетки $\langle R \rangle$;
- собственно моделирование;
- вывод результатов – полных состояний схемы или (по желанию пользователя) состояний отдельных узлов.

ЛИТЕРАТУРА

1. Агibalов Г.П. Дискретные автоматы на полурешетках. Томск: Изд-во Том. ун-та, 1993. 227 с.
2. Киносита К., Асада К., Карацу О. Логическое проектирование СБИС: Пер. с япон. М.: Мир, 1988. 309 с.

Статья представлена кафедрой программирования факультета прикладной математики и кибернетики Томского государственного университета, поступила в научную редакцию 11 января 1999г.