УДК 004.312 DOI: 10.31114/2078-7707-2020-2-35-42

Использование SAT-решателей и ROBDD-графов для построения схем, маскирующих логические неисправности и вредоносные подсхемы

А.Ю. Матросова, В.А. Провкин, С.А. Останин

Институт прикладной математики и компьютерных наук, Национальный исследовательский Томский государственный университет, г. Томск

mau11@yandex.ru, prowkan@mail.ru, sergeiostanin@yandex.ru

***Аннотация* — В комбинационной схеме *C* из вентилей выделена подсхема с множеством *V* выходных полюсов и множеством *U* входных полюсов. Множество *V* состоит из выходных полюсов неисправных элементов схемы (рассматриваются логические неисправности) и входных полюсов исправных элементов, таких, что полюс является выходом линии, в которую включена вредоносная подсхема. Конкретный выбор множества *U* зависит от технической реализации схемы *C*, и здесь не обсуждается. (В частности, множество *U* может являться подмножеством входов схемы *C*). Предлагается корректировать поведение схемы *C*, используя маскирующую схему, желательно более простую, чем заданная подсхема, в рамках Engineering Change Order (ECO) технологий. Известные подходы к построению маскирующей схемы основаны на результатах моделирования поведения корректируемой схемы *C* на подмножестве входных наборов. Этот подход гарантирует корректное поведение схемы на подмножестве моделируемых наборов. Мы предлагаем при построении маскирующей схемы использовать частичные функции полюсов множества *V*, что позволяет гарантировать корректное поведение схемы *C* на множестве всех ее входных наборов. Построение частичных функций выполняется либо с использованием SAT-решателей, либо с использованием операций над ROBDD-графами, либо с совместным применением обеих технологий. Далее полученные частичные функции, зависящие от входных переменных схемы *C*, отображаются на множество *U* внутренних переменных, применяя эти же технологии, если множество *U* не является подмножеством входов схемы *C*. По системе частичных функций с помощью систем ESPRESSO и ABC строится маскирующая схема.**

***Ключевые слова* — комбинационные схемы, частичные функции, функции наблюдаемости внутренних полюсов, КНФ Цейтина, ортогональные ДНФ (ОДНФ), ROBDD-графы, SAT решатели, ECO технологии.**

I. Введение

Современные интегральные схемы характеризуются высокой сложностью в условиях относительно малого числа полюсов, доступных для наблюдения за поведением схемы. В связи с этим возможно обнаружение логических неисправностей на поздних стадиях производства схемы, что крайне нежелательно, поскольку приходится либо возвращать схему на более ранние этапы производства, либо выбрасывать ее как непригодную к использованию. В итоге снижается выход годных схем. Кроме того, в схему, изготовленную сторонними фирмами, могут быть внедрены вредоносные подсхемы (Trojan Circuits (TCs)) с целью разрушения схемы или извлечения конфиденциальной информации. Срабатывание вредоносной подсхемы (TC) можно рассматривать как проявление логической неисправности на входном полюсе элемента, являющемся выходом линии связи схемы *C*, к которой присоединён выход (payload) вредоносной подсхемы. Структура вредоносной подсхемы и способы подключения ее входов в схему *C* значения не имеют. В связи со сказанным, в дальнейшем будем говорить только о неисправных полюсах и их коррекции. Для коррекции схем применяются ECO (Engineering Change Order) технологии [1]-[3]. В рамках этих технологий неисправности полюсов в изготавливаемой схеме можно, в частности, маскировать с помощью корректирующей (маскирующей) схемы, подключаемой к внутренним полюсам схемы *C*. Для маскирующей схемы на кристалле отводится дополнительная площадь. Маскирующая схема, по возможности, минимизируется с целью сокращения используемой дополнительной площади. Известные подходы к построению маскирующей схемы основаны на результатах моделирования поведения корректируемой схемы *C* на подмножестве входных наборов [1]-[3]**.** Такая информация для сложных логических схем с несколькими десятками входов, как правило, оказывается неполной. Этот подход гарантирует корректное поведение схемы на множестве моделируемых наборов. Мы предлагаем при построении маскирующей схемы использовать частичные функции полюсов множества *V*. Это позволяет гарантировать корректное поведение схемы *C* на множестве всех ее входных наборов, причем, достаточно обнаружить искажение поведения полюса хотя бы на одном входном наборе, чтобы включить его в множество *V*. Будем иметь в виду, что каждому внутреннему полюсу комбинационной схемы *C* соответствует частично определённая булева функция [4], зависящая от входных переменных этой схемы. При проявлении неисправности, а также при активации вредоносной под-

схемы происходит искажение этой функции. В работе [5] корректное функционирование корректируемой схемы *C* в условиях доступности множеств *V, U* ее полюсов обеспечивается отображением частичных функций полюсов из *V* от входных переменных схемы C на множество *U* внутренних полюсов После этого полученное отображение реализуется в виде логической схемы. (Если множество *U* является подмножеством входов схемы *C*, то для получения маскирующей схемы используются построенные частичные функции полюсов из *V,* зависящие от входных переменных схемы *C)*. Маскирующие схемы разработаны в условиях применения операций над ROBDD-графами [5]. Эти операции, как известно, характеризуются полиномиальной сложностью. Однако ROBDD-графы могут быть получены не для всяких корректируемых логических схем. Дело в том, что для некоторых схем количество внутренних вершин ROBDD-графа растёт экспоненциально по отношению к числу переменных графа (числу входов корректируемой схемы) независимо от выбора порядка разложения по переменным, что исключает возможность использования ROBDD-графов, если число входов таких схем достигает нескольких десятков. В этих ситуациях предлагается альтернативный подход, основанный на определении выполнимости соответствующих специальным схемам булевых формул, записанных в виде КНФ. Для каждого из неисправных полюсов множества *V* строятся две логические схемы, представляющие множества единичных и нулевых наборов соответствующей ему частичной функции от входных переменных схемы. Затем для получения КНФ к этим схемам применяется преобразование Цейтина [6], будем называть их в дальнейшем КНФ Цейтина. Длина построенных КНФ линейно зависит от числа элементов схем, порождающих эти КНФ. С помощью SAT решателя из КНФ Цейтина извлекаются множества единичных и нулевых наборов частичной функции полюса. Множества представляются в виде ортогональных ДНФ (ОДНФ) в пространстве входных переменных схемы. Будем иметь в виду, что частичные функции полюсов из множества *V,* как правило, являются слабо определенными, как для выходных, так и для входных полюсов множества *V*. Это связано с тем, что множество единичных (нулевых) наборов частичной функции есть множество тестовых наборов для константной неисправности 0 (1) этого полюса. Если тестовые наборы составляют существенную долю в пространстве всех входных наборов корректируемой схемы, то логическая неисправность такого полюса обычно обнаруживается на ранних этапах производства схемы. Чтобы затруднить обнаружение вредоносных подсхем, последние подключаются в линии, заходящие в полюсы, характеризующиеся слабо определенными частичными функциями. Проведенные нами предварительные эксперименты показали, что для корректируемых схем, для которых не удается построить ROBDD-графы из-за их размеров, и для выделенных в них полюсов из *V,* характеризующихся слабо определенными частичными функциями, получаются достаточно простые ОДНФ (из нескольких десятков конъюнкций). Они могут непосредственно использоваться для получения маскирующих схем, если множество *U* является подмножеством входных полюсов схемы *C*. Иначе для полученных ОДНФ, зависящих от входных переменных схемы *C*, находится отображение на множество *U* ее внутренних полюсов, также с использованием SAT-решателя. Затем по найденному отображению с помощью систем ESPRESSO [7] и ABC [8] строится маскирующая схема. Заметим, что SAT-решатели становятся все более эффективными. Их использование позволяет продвигаться в область маскирования неисправностей все более сложных логических схем. В то же время следует иметь в виду, что, если ту или иную задачу в рамках рассматриваемой проблемы можно решать с использованием операций над ROBDD-графами, то надо применять эти операции, если задача решается быстрее, чем с применением SAT-решателя. Иными словами, совместное использование обеих технологий, предлагаемое в данной работе, является, на наш взгляд, перспективным подходом при решении данной проблемы.

II. Постановка задачи

Задана комбинационная логическая схема *C* (комбинационный эквивалент схемы с памятью), построенная из вентилей AND, OR, NAND, NOR, NOT, XOR. Выделена подсхема с множеством *V* выходных полюсов и множеством *U* входных полюсов. На полюсах множества V зафиксировано некорректное поведение схемы C. Множество *U*, в частности, может быть подмножеством входов схемы *C*. Необходимо корректировать поведение схемы *C*, используя маскирующую схему, желательно более простую, чем заданная подсхема.

В работе [5] показано, что получение более простых маскирующих схем, основанное на применении операций над ROBDD-графами, возможно, в частности, когда неисправные полюсы удалены как от входов, так и от выходов корректируемой схемы. Именно в таких полюсах соответствующие частичные функции оказываются, как правило, слабо определенными, а неисправности, сопоставляемые этим полюсам, — трудно обнаружимыми. Такие неисправности обычно удается локализовать лишь на последних этапах создания схемы. Заметим также, что вредоносные подсхемы (TCs) обычно включаются в линии, константные неисправности которых трудно обнаружимы.

Поставленная задача решается в работе [5] с использованием ROBDD-графов и операций над ними. Однако, как уже упоминалось выше, ROBDD-графы нельзя построить для некоторых схем. В этой ситуации нами предлагается использовать SAT-решатели, предварительно получив две схемы из вентилей, представляющие множество единичных и нулевых наборов частичной функции для полюса *v* из *V,* а затем соответствующие им КНФ Цейтина. Следует иметь в виду, что как при использовании ROBDD-графов, так и при применении SAT-решателей приходится искать компактное представление частичной функции полюса

v. В первом случае частичная функция полюса представляется в виде двух ROBDD-графов, во втором — в виде двух ОДНФ, извлеченных из вышеупомянутых КНФ с помощью SAT-решателя. В случае, если множество *U* не совпадает с множеством входных полюсов, в рамках обеих технологий приходится выполнять «точное» троичное моделирование, впервые предложенное в нашем коллективе в конце 90-х годов. В данной работе более детально представляется подход, основанный на использовании SAT-решателей, поскольку подход на основе операций над ROBDD-графами нами разработан ранее и представлен в работе [5].

III. Необходимые понятия и общий подход к решению задачи

Обозначим через f_v частичную функцию, сопоставляемую полюсу *v* схемы *C*, зависящую от входных переменных схемы. Пусть $M_1(f_v)$ — множество единичных наборов этой функции, а $M_0(f_v)$ — множество ее нулевых наборов.

Рассмотрим частичную булеву функцию f_1 и полностью определенную булеву функцию f_2, зависящие от одного и того же множества переменных и заданные множествами их единичных и нулевых наборов: $M_1(f_1)$, $M_0(f_1)$; $M_1(f_2)$, $M_0(f_2)$.

Определение. Полностью определенная функция f_2 является реализацией частичной функции f_1, если $M_1(f_1)\cap M_0(f_2)=\varnothing$ и $M_0(f_1)\cap M_1(f_2)=\varnothing$.

В таком случае $M_1(f_2)\supseteq M_1(f_1)$ и $M_0(f_2)\supseteq M_0(f_1)$.

Рассмотрим функцию $f_i(v,x_1,\ldots,x_n)$, реализуемую на *i*-м выходе схемы *C*, предполагая, что переменная *v* является входной наряду с переменными $x_1,\ldots,x_n$. Эта функция представляется подсхемой, получаемой при движении от *i*-го выхода к входам схемы *C* и подстановке функций от входных переменных соответствующих вентилей (без раскрытия скобок) вместо внутренних переменных (кроме переменной *v*) схемы *C*, пока это возможно. Выход полученной подсхемы есть *i*-й выход схемы *C*, а полюс *v* является входом полученной схемы наряду с входами $x_1,\ldots,x_n$ схемы *C*.

Представим булеву разность $D_v f_i$ функции f_i по переменной *v*:

$$D_v f_i = f_i^{v=0} \oplus f_i^{v=1}. \qquad (1)$$

Здесь $f_i^{v=0} = f_i(0,x_1,\ldots,x_n)$, $f_i^{v=1} = f_i(1,x_1,\ldots,x_n)$.

Заметим, что булева разность $D_v f_i$ задает также функцию наблюдаемости полюса *v* на *i*-м выходе схемы *C*. Действительно, функция $D_v f_i$ принимает единичное значение на всяком наборе значений переменных $x_1,\ldots,x_n$, при подстановке которого в схему *C* изменение значения переменной *v* приводит к изменению значения *i*-го выхода схемы *C*.

Функция наблюдаемости f^{obs} полюса *v* для схемы *C* в целом представляется выражением:

$$f^{obs} = \bigvee_{i=1}^{m_v} (D_v f_i), \qquad (2)$$

где m_v есть число выходов схемы *C*, связанных с полюсом *v*.

Для полюса *v* схемы *C* рассмотрим функцию $\phi(x_1,x_2,\ldots,x_n)$, представляемую подсхемой C_v схемы *C*. Выходом подсхемы C_v является полюс *v*, а ее входами — входы схемы *C*.

Представим множество $M_1(f_v)$ единичных наборов частичной функции f_v, которое в то же время является множеством всех тестовых наборов для константной неисправности 0 на полюсе *v*, выражением

$$\phi(x_1,x_2,\ldots,x_n)\,\&\, f^{obs}(x_1,\ldots,x_n), \qquad (3)$$

а множество $M_0(f_v)$ нулевых наборов частичной функции f_v, которое, в свою очередь, является множеством всех тестовых наборов для константной неисправности 1 на полюсе *v*, выражением:

$$\overline{\phi(x_1,x_2,\ldots,x_n)}\,\&\, f^{obs}(x_1,\ldots,x_n). \qquad (4)$$

Напомним, что конъюнкции ортогональны, если они содержат взаимно инверсные литеры. Под литерой здесь и далее будем понимать переменную вместе с ее знаком инверсии.

ДНФ ортогональна, если ее конъюнкции попарно ортогональны.

Опишем более детально общий подход к решению поставленной выше задачи с использованием SAT-решателя.

1) Представляем формулы 3, 4 двумя логическими схемами из вентилей.

2) Упрощаем эти схемы с тем, чтобы сократить число внутренних переменных в соответствующих схемам КНФ Цейтина.

3) Строим КНФ для упрощенных схем методом Цейтина. Эти КНФ представляют множества единичных и нулевых наборов частичной функции f_v полюса *v*.

4) Воспользовавшись SAT-решателем, из имеющихся КНФ Цейтина получаем представление множеств единичных и нулевых наборов частичной функции f_v в виде двух ортогональных ДНФ на множестве входных переменных $x_1,\ldots,x_n$ схемы *C*.

5) Если множество *U* не является подмножеством входов схемы *C*, находим отображение частичной функции f_v на множество внутренних переменных $u_1,\ldots,u_m$ схемы *C*, представляя отображение также в виде двух ДНФ. Внутренние переменные $u_1,\ldots,u_m$ в то же время являются внутренними переменными

подсхемы C_v. Множество $u_1,\ldots,u_m$ переменных выбирается из различных соображений, учитывающих техническую реализацию схемы C [3], [6]. При этом подсхема с выходным полюсом v и входами $u_1,\ldots,u_m$ вместе с подсхемами, имеющими в качестве выходов полюсы $u_1,\ldots,u_m$ и входы $x_1,x_2,\ldots,x_n$, образуют подсхему C_v. Обозначим символом $f_v(u_1,\ldots,u_m)$ частичную функцию полюса v в пространстве переменных $u_1,\ldots,u_m$. Заметим, что подсхема $C_v(u_1,\ldots,u_m)$ с выходом v и входами $u_1,\ldots,u_m$ является реализацией частичной функции $f_v(u_1,\ldots,u_m)$. Наша цель — найти, по возможности, более простую реализацию и затем использовать ее в качестве маскирующей схемы. В работе [5] показано, что маскирующая схема может быть в несколько раз проще (по числу составляющих ее вентилей), чем упомянутая выше реализация. Будем иметь в виду, что множество $u_1,\ldots,u_m$, как правило, не велико и его мощность достигает одного-двух десятков, в то время как число входных переменных схемы C может достигать многих десятков.

6) С помощью системы ESPRESSO [7] из полученной частичной функции извлекаем безызбыточную ДНФ, ее реализующую.

7) С помощью системы ABC [8] по безызбыточной ДНФ в пространстве переменных $u_1,\ldots,u_m$ получаем маскирующую схему.

Будем иметь в виду, что при построении по схеме КНФ методом Цейтина следует использовать различные внутренние переменные для различных подсхем схемы C, даже если подсхемы реализуют взаимно инверсные функции.

При неисправности множества V полюсов, $V=\{v_1,\ldots,v_q\}$, необходимо получить безызбыточную систему ДНФ для полюсов множества V в пространстве переменных $u_1,\ldots,u_m$, либо в пространстве входных переменных схемы C, если множество U является подмножеством входных полюсов этой схемы.

Остановимся на детальном описании некоторых из перечисленных шагов.

IV. Упрощение схемы, представляющей булеву разность и получение представления частичной функции полюса в виде двух КНФ Цейтина

Выполним упрощение схемы, представляющей булеву разность $D_v f_i = f_i^{v=0} \oplus f_i^{v=1}$, отыскивая в ней одинаковые подсхемы, с целью упрощения соответствующей схеме конъюнктивной нормальной формы (КНФ).

Булева разность $D_v f_i$ может быть представлена следующей схемой (рис. 1):

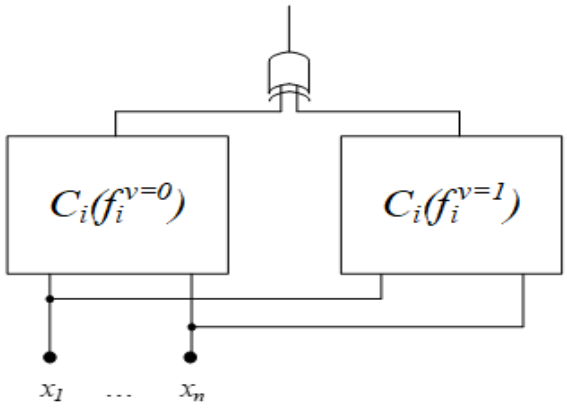

Рис. 1. Схема, реализующая булеву разность $D_v f_i$

Здесь схема $C_i(f_i^{v=0})$ реализует функцию $f_i^{v=0}$, а схема $C_i(f_i^{v=1})$ реализует функцию $f_i^{v=1}$. В этих схемах могут найтись одинаковые подсхемы, которые не зависят от переменной v. Это значит, что в схеме можно оставлять одну из одинаковых подсхем такого типа. Разработан алгоритм упрощения схемы, представляющей булеву разность $D_v f_i$.

A. *Алгоритм упрощения схемы*

Извлекаем подсхему $C_v(out)$ из схемы C.

1. Включаем в подсхему $C_v(out)$ все вентили между полюсом v и i-м выходом схемы C вместе с линиями, их соединяющими. В полученной подсхеме имеется один выход, совпадающий с i-м выходом схемы C, а ее входами являются свободные входы вентилей, вошедших в подсхему и полюс v.

2. Разделим свободные входы вентилей схемы $C_v(out)$ на s подмножеств. Каждый вход j-го подмножества связан с одним и тем же внутренним полюсом w_j схемы C.

3. Формируем множество $w_1,\ldots,w_s$ таких полюсов. Они питают вентили, имеющие входы, связанные с полюсом v. Будем рассматривать полюсы $w_1,\ldots,w_s$ в качестве входов подсхемы $C_v(out)$.

4. Подсхема $C_v(out)$ порождает две подсхемы: $C_j^1(out)$ и $C_j^0(out)$ при фиксировании полюса v константами 1, 0, соответственно.

Далее формируем подсхему $C_v(in)$ из схемы C с выходами $w_1,\ldots,w_s$, включая полюс v, если это необходимо, и входами $x_1,\ldots,x_n$.

Создадим упрощенный фрагмент схемы, представленной на рис. 1, следующим образом (рис. 2).

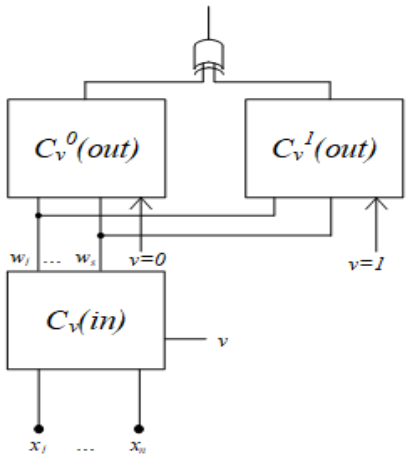

Рис. 2. Упрощенная схема, представляющая булеву разность $D_v f_i$

Проведенные нами эксперименты на контрольных примерах из базы ISCAS показали, что, как правило, число внутренних переменных удается сократить на 30% процентов и более.

Напомним, что булева разность $D_v f_i$ для одного выхода схемы C есть функция наблюдаемости полюса v на этом выходе. Для схемы в целом функция наблюдаемости f^{obs} этого полюса представляется схемой на рис. 3.

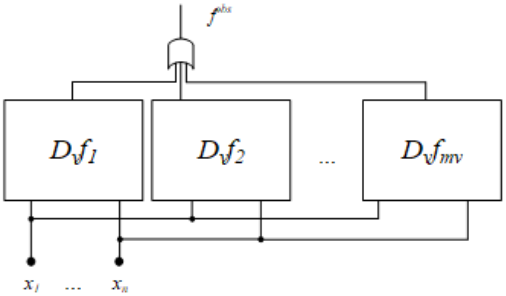

Рис. 3. Представление функции наблюдаемости для многовыходной схемы C

Представив в виде схемы функцию наблюдаемости для многовыходной схемы C, мы можем получить схемы, представляющие множества единичных и нулевых наборов частичной функции полюса v, показанные на рис. 4, 5, соответственно. Полюс v используется как дополнительный выход.

По схемам, представляющим множества единичных и нулевых наборов частичной функции, формируются КНФ Цейтина: $КНФ_1$ и $КНФ_0$.

Далее с помощью SAT-решателей получаем из этих КНФ ортогональные ДНФ.

V. Извлечение ОДНФ из КНФ Цейтина

1. Для $КНФ_1$ ($КНФ_0$) находим набор, обращающий эту КНФ в единицу, используя SAT-решатель. Если такого набора не существует, переходим к п.4 алгоритма. В найденном наборе выделяем его часть, представленную входными переменными схемы C.

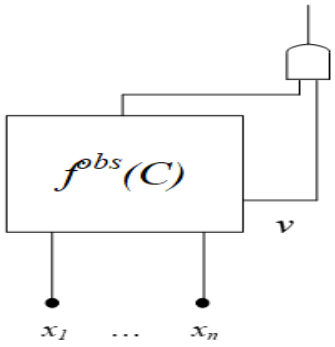

Рис. 4. Схема, представляющая множество единичных наборов частичной функции полюса v

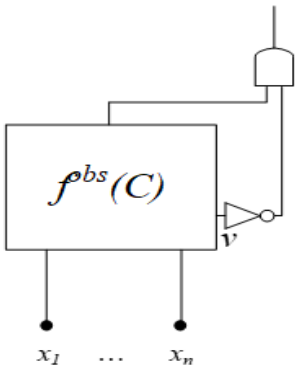

Рис. 5. Схема, представляющая множество нулевых наборов частичной функции полюса v

2. Расширяем этот набор, удаляя одну за другой входные переменные, насколько это возможно. В результате получаем конъюнкцию от входных переменных. При удалении каждой переменной обращаемся к SAT-решателю с целью проверки, является ли соответствующая конъюнкция импликантой функции, представляемой рассматриваемой КНФ. (Алгоритм проверки будет описан далее). Если имеем первый из найденных наборов, то расширение происходит в области единичных значений исходной КНФ. Иначе расширяем очередной набор в оставшейся от исходной КНФ области единичных значений, полученной исключением из исходной области уже найденных конъюнкций. Очередная полученная конъюнкция исключается из функции, представляемой КНФ, путем добавления ее инверсии (клаузы) к имеющейся КНФ.

3. Очередная полученная конъюнкция включается в формируемую ортогональную ДНФ. Переходим к п.1 алгоритма, рассматривая оставшуюся КНФ.

4. Работа алгоритма закончена.

Применив предложенный алгоритм к $КНФ_1$ и $КНФ_0$, получим представление частичной функции f_v в виде двух ортогональных ДНФ на множестве входных переменных $x_1,\ldots,x_n$ схемы C: D_v^1, D_v^0, соответственно.

VI. Отображение частичной функции от входных переменных схемы на множество её внутренних переменных

В условиях, когда множество U не является подмножеством входных переменных схемы C, необходимо далее найти отображение частичной функции на внутренние переменные $u_1,\ldots,u_m$, то есть найти множества единичных и нулевых наборов частичной функции $f_v(u_1,\ldots,u_m)$ полюса v схемы C на заданном подмножестве U внутренних переменных этой схемы. Для функций большого (возможно, многих десятков) числа переменных нельзя воспользоваться двоичным моделированием с целью получения такого отображения. Поэтому предлагается специальная процедура, основанная на использовании сочетания троичного моделирования с двоичным и применении SAT-решателя, возможно, с использованием операций над ROBDD - графами.

Сначала к троичным векторам, представляющим конъюнкции ДНФ D_v^1, D_v^0, применяется «точное» троичное моделирование, позволяющее получить области, содержащие единичные и нулевые наборы частичной функции $f_v(u_1,\ldots,u_m)$. Это моделирование выполняется, либо с использованием операций над ROBDD-графами [5], либо с использованием SAT-решателя. Полученные области представляются троичными векторами, как правило, зависящими от небольшого числа переменных (в пределах одного, двух десятков). Затем используется двоичное моделирование на подсхеме $C_v(u_1,\ldots,u_m)$ с целью выделения двоичных векторов, относящихся к области единичных и нулевых наборов этой подсхемы. Далее с помощью SAT-решателя или операций над ROBDD-графами выполняется проверка каждого выделенного набора на его достижимость в процессе функционирования схемы C. Достижимые наборы (булевы векторы) включаются в соответствующие им области единичных (нулевых) наборов частичной функции $f_v(u_1,\ldots,u_m)$.

Под «точным» троичным моделированием одно выходной логической схемы будем понимать моделирование, приводящее к следующим результатам.

Результатом «точного» троичного моделирования некоторой схемы, обозначим ее C_{u_j}, является значение 1 (0), если интервал, сопоставляемый троичному вектору, целиком содержится в области единичных (нулевых) наборов значений функции, представляемой схемой C_{u_j}. Иначе результатом троичного моделирования этой схемы является неопределенное значение «-». Известно, что непосредственная подстановка троичного вектора в логическую схему (традиционное троичное моделирование) может давать значение «-» в ситуации, когда соответствующий вектору интервал содержится в области единичных (нулевых) наборов значений переменных функции, представляемой схемой, что нежелательно [9].

В работах [5] было предложено выполнять точное троичное моделирование с использованием операций над ROBDD-графами.

В данной работе мы предлагаем использовать для этой цели SAT-решатель с целью расширения класса схем, для которых маскирование оказывается возможным. Пусть $к_\alpha$ – конъюнкция, сопоставляемая троичному вектору α. Строим две схемы для получения по ним КНФ Цейтина. Для проверки поглощаемости конъюнкции $к_\alpha$ областью единичных (нулевых) наборов значений входных переменных схемы C_{u_i} используется схема, представленная на рис. 6 (рис. 7). Обозначим полученные схемы $C_{u_j}^1$, $C_{u_j}^0$, соответственно. Строим для них КНФ Цейтина КНФ ($C_{u_j}^1$) и КНФ ($C_{u_j}^0$). Здесь f — функция, реализуемая схемой C_{u_i} на выходе u_i.

A. *Алгоритм точного троичного моделирования по КНФ Цейтина*

1) Находим выполняющий набор для КНФ ($C_{u_j}^1$). Если его не существует, то результатом троичного моделирования является значение 1. Иначе переходим к следующему пункту алгоритма.

2) Находим выполняющий набор для КНФ ($C_{u_j}^0$). Если его не существует, то результатом троичного моделирования является значение 0.

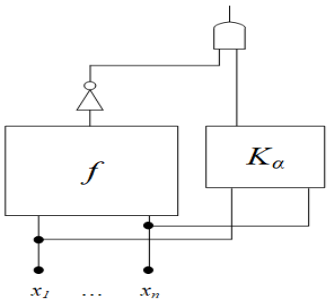

Рис. 6. Схема $C_{u_j}^1$

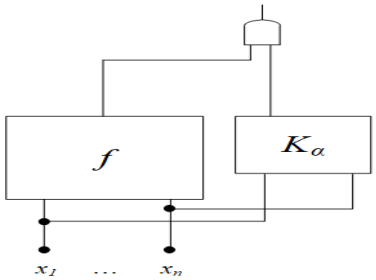

Рис. 7. Схема $C_{u_j}^0$

3) Иначе результатом троичного моделирования является значение «-».

B. Алгоритм точного троичного моделирования по КНФ Цейтина

4) Находим выполняющий набор для КНФ ($C^1_{u_j}$). Если его не существует, то результатом троичного моделирования является значение 1. Иначе переходим к следующему пункту алгоритма.

5) Находим выполняющий набор для КНФ ($C^0_{u_j}$). Если его не существует, то результатом троичного моделирования является значение 0.

6) Иначе результатом троичного моделирования является значение «-».

Выполнив «точное» троичное моделирование для каждой из подсхем схемы $C_v(u_1,\ldots,u_m)$ с выходами $u_1,\ldots,u_m$ и входами $x_1,\ldots,x_n$, получим троичный вектор β. Вектор β является результатом «точного» троичного моделирования вектора α для множества полюсов $u_1,\ldots,u_m$ схемы C. Зная результаты точного троичного моделирования для конъюнкций (троичных векторов) ДНФ D^1_v, D^0_v, получим множества троичных векторов M^*_1, M^*_0, которые содержат булевы векторы, представляющие множества единичных и нулевых наборов искомой частичной функции $f_v(u_1,\ldots,u_m)$, соответственно. Получая значения «-» в троичном векторе β мы не знаем, на каких именно, булевых векторах достигается значение 1, а на каких – значение 0 в полюсе v, и какие из булевых векторов достижимы на полюсах $u_1,\ldots,u_m$ схемы C. Поэтому далее предлагается выполнить следующие процедуры.

Воспользуемся двоичным моделированием, а именно, для каждого троичного вектора из M^*_1, (M^*_0), формируем множество булевых векторов в пространстве $u_1,\ldots,u_m$ переменных. Подставляем булев вектор γ в подсхему $C_v(u_1,\ldots,u_m)$. Если подсхема обращается в 1 (0), формируем для вектора γ схему C_γ. Схема C_γ представляет перемножение функций одно выходных подсхем схемы C, соответствующих компонентам $u_1,\ldots,u_m$ вектора γ и зависящих от входных переменных схемы C. Если компонента вектора γ принимает значение 0, выходной элемент подсхемы заменяется инверсным элементом. Входные переменные заменены константами из троичного вектора α, порождающего вектор β, а затем вектор γ. В случае выполнимости КНФ, сопоставляемой схеме C_γ, вносим булев вектор γ в множество $M_1\left(f_{v_j}(u_1,\ldots,u_m)\right)$ единичных ($M_0\left(f_{v_j}(u_1,\ldots,u_m)\right)$ нулевых) наборов значений переменных частичной функции $f_v(u_1,\ldots,u_m)$.

Если задано множество $V=\{v_1,\ldots,v_q\}$ неисправных полюсов, то находим множества $M_1\left(f_{v_j}(u_1,\ldots,u_m)\right)$, $M_0\left(f_{v_j}(u_1,\ldots,u_m)\right)$ для частичных функций каждого полюса v_j из V.

VII. Получение маскирующей схемы

Получив множества $M_1\left(f_{v_j}(u_1,\ldots,u_m)\right)$, $M_0\left(f_{v_j}(u_1,\ldots,u_m)\right)$ для частичных функций каждого полюса v_j из V, применяем систему ESPRESSO [7] с целью представления реализации частичных функций в виде системы ДНФ, а затем используем эту систему ДНФ в ABC [8] комплексе для создания маскирующей схемы C_p. Полученная маскирующая схема связана с полюсами $u_1,\ldots,u_m$ схемы C, а каждый ее выход соединяется с полюсами, питаемыми соответствующим неисправным полюсом схемы C (рис. 8).

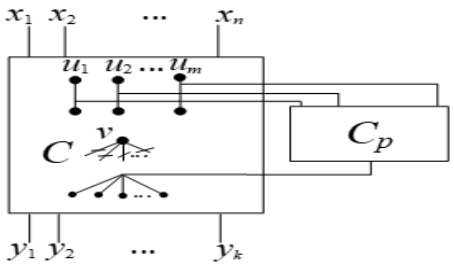

Рис. 8. Подключение одновыходной маскирующей схемы для неисправного полюса *v*

При маскировании вредоносной подсхемы, выход которой подключен к некоторой линии l схемы C, выход маскирующей схемы C_p соединяется с входом вентиля (одного), питаемого этой линией (рис. 9).

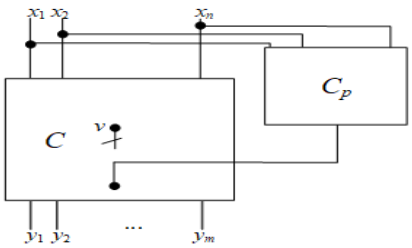

Рис. 9. Подключение одновыходной маскирующей схемы для маскирования ТС

VIII. Заключение

Предварительные эксперименты показали целесообразность применения изложенного подхода с целью получения, по возможности, более простых маскирующих схем к корректируемым схемам, для которых ROBDD-графы построить не удается. Поскольку SAT-технологии непрерывно и быстро совершенствуются, использование SAT-решателей, когда применение ROBDD графов для коррекции логических схем невозможно, и операций над ROBDD-графами, когда это сокращает вычислительные затраты, позволит продвинуться в применении предлагаемого подхода к все более сложным схемам.

Литература

[1] S. Krishnavami, H. Ren,N. Modi and Puri. DeltaSyn: an efficient logic difference optimizer for ECO synthesis // in Proc. Asia and South Pacific Design Automation Conference, 2009, pp. 789-796.

[2] A.-C. Cheng, H.-R. Jiang and J.-Y. Jou Resource-aware functional ECO patch generation // in Proc. DATE, 2016.

[3] A.Q. Dao, N.-Z. Lee, L.-C. Chen, M.P.-H. Lin, J.-H.R. Jiang, A. Mishchenko, and R. Brayton Efficient computation of ECO patch functions // in Proc. DAC, 2018.

[4] Matrosova A., Ostanin S. Trojan Circuits Masking and Debugging of Combinational Circuits with LUT Insertion // 2018 IEEE International Conference on Automation, Quality and Testing, Robotics. AQTR 2018 (THETA 21), 24-26 may 2018, Cluj-Napoca, Romania. [Cluj-Napoca], 2018. P. 462-467. 1 CD-R.

[5] Matrosova A., Provkin V., Nikolaeva E. Masking Internal Node Faults and Trojan Circuits in Logical Circuits // Proceedings of 2019 IEEE East-West Design & Test Symposium (EWDTS), 13-16 september 2019, Batumi. Kharkov: IEEE, 2019. P. 416-419.

[6] Цейтин Г.С. О сложности вывода в исчислении высказываний // Записки научных семинаров ЛОМИ АН СССР. 1968. Т. 8. С. 234-259.

[7] Logic Minimization Software (http://ramos.elo.utfsm.cl/~lsb/elo211/aplicaciones/aplicaciones/espresso/ESPRESSO Logic Minimization Software.htm) (дата обращения: 21.03.2020).

[8] ABC: A System for Sequential Synthesis and Verification (https://people.eecs.berkeley.edu/~alanmi/abc/) (дата обращения: 21.03.2020).

[9] A. Matrosova, O. Goloubeva, S. Tsurikov On correction of the results of ternary simulation and preliminary estimation of the correction results // Proceedings of the 6-th Biennal Conference on Electronics and Microsystems Technology, Tallinn. 1998. P. 183-186.

Applying SAT Solvers and ROBDDs for Deriving Circuits Masking Logical Faults and TSs in Discrete Systems

A.Yu. Matrosova, V.A. Provkin, S.A. Ostanin

Institute of applied mathematics and computer science, National Research Tomsk State University, Tomsk, mau11@yandex.ru, prowkan@mail.ru, sergeiostanin@yandex.ru

***Abstract* — A combinational circuit *C* (the combinational part of a sequential circuit) consisting of gates and its sub-circuit C_s with a set *V* of output nodes and a set *U* of input nodes is considered. A set *V* comprises outputs of circuit *C* fault gates (logical faults are regarded) and inputs of fault free gates so that these inputs are at the same time outputs of lines in that of which Trojan Circuit (TC) may be injected. A set *U* consists of either internal nodes of circuit *C* or some its input nodes. Correction of circuit *C* behavior in the frame of Engineering Change Order (ECO) technology is suggested. It is reduced to deriving masking circuit (patch circuit C_p) that is simpler C_s. Note that additional area for C_p is necessary and we try to decrease it. There are many ways of deriving masking circuits. As a rule, the approaches are based on results of circuit *C* simulation on sub-set of its input vectors. These approaches guarantee correct behavior on these input vectors. We derive masking (patch) circuit using incompletely specified functions of nodes from *V*. This way guarantees a correct behavior of circuit *C* on the all input vectors. We form incompletely specified Boolean functions for each node *v* from *V* depending on either internal variables (a set *U*) or some input variables of the circuit *C*. Deriving of incompletely specified Boolean functions is based on either applying operations on ROBDDs or/and using SAT solvers. The situation when faults are detected on the last stage of a circuit fabrication is considered. In this case we need to return to the first stages of the fabrication or refuse applying this circuit at all. To increase yield, masking fault technology is used. In this case incompletely specified Boolean functions of fault nodes, as a rule, are poor determined ones. In this paper we consider TCs that pay load is injected into circuit *C* line. Output of this line that is input of fault free gate usually is also characterized by poor determined incompletely specified Boolean function. That is why applying information about these functions we may get simpler C_p in comparison with C_s. Experimental results represented in paper [5] confirms it. Having incompletely specified Boolean functions, we get masking circuit applying ESPRESSO and ABC systems. Note that SAT solvers become more and more effective and operations on ROBDDs are characterized by polynomial complexity. Hopefully using the both techniques we may execute masking more and more complicated circuits.**

***Keywords* — combinational circuits, incompletely specified Boolean functions, observability functions, Disjoint Sum of Products (DSoP), Tseitin CNF, ROBDDs, SAT solvers, ECO technologies.**

References

[1] S. Krishnavami, H. Ren,N. Modi and Puri. DeltaSyn: an efficient logic difference optimizer for ECO synthesis // in Proc. Asia and South Pacific Design Automation Conference, 2009, pp. 789-796.

[2] A.-C. Cheng, H.-R. Jiang and J.-Y. Jou Resource-aware functional ECO patch generation // in Proc. DATE, 2016.

[3] A.Q. Dao, N.-Z. Lee, L.-C. Chen, M.P.-H. Lin, J.-H.R. Jiang, A. Mishchenko, and R. Brayton Efficient computation of ECO patch functions // in Proc. DAC, 2018.

[4] Matrosova A., Ostanin S. Trojan Circuits Masking and Debugging of Combinational Circuits with LUT Insertion // 2018 IEEE International Conference on Automation, Quality and Testing, Robotics. AQTR 2018 (THETA 21), 24-26 may 2018, Cluj-Napoca, Romania. [Cluj-Napoca], 2018. P. 462-467. 1 CD-R.

[5] Matrosova A., Provkin V., Nikolaeva E. Masking Internal Node Faults and Trojan Circuits in Logical Circuits // Proceedings of 2019 IEEE East-West Design & Test Symposium (EWDTS), 13-16 september 2019, Batumi. Kharkov: IEEE, 2019. P. 416-419.

[6] Tseitin G.S. O slozhnosti vyvoda v ischislenii vyskazyvanij (On the Complexity of Derivation in Propositional Calculus) // Zapiski nauchnyh seminarov LOMI AN SSSR. 1968. T. 8. C. 234-259 (in Rassion).

[7] Logic Minimization Software (http://ramos.elo.utfsm.cl/~lsb/elo211/aplicaciones/aplicaciones/espresso/ESPRESSO Logic Minimization Software.htm) (access date: 21.03.2020).

[8] ABC: A System for Sequential Synthesis and Verification (https://people.eecs.berkeley.edu/~alanmi/abc/) (access date: 21.03.2020).

[9] A. Matrosova, O. Goloubeva, S. Tsurikov On correction of the results of ternary simulation and preliminary estimation of the correction results // Proceedings of the 6-th Biennal Conference on Electronics and Microsystems Technology, Tallinn. 1998. P. 183-186.