МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

# МАТЕРИАЛЫ

## VII Международной молодежной научной конференции

# «МАТЕМАТИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ИНФОРМАЦИОННЫХ, ТЕХНИЧЕСКИХ И ЭКОНОМИЧЕСКИХ СИСТЕМ»

**Томск, 23–25 мая 2019 г.**

*Под общей редакцией*
*кандидата технических наук И.С. Шмырина*

Томск
Издательский Дом Томского государственного университета
2019

# СИНТЕЗ ЧАСТИЧНО ПРОГРАММИРУЕМЫХ СХЕМ, ОРИЕНТИРОВАННЫЙ НА МАСКИРОВАНИЕ ТРУДНО ОБНАРУЖИМЫХ НЕИСПРАВНОСТЕЙ

**В.А. Провкин**
*Томский государственный университет*

## Введение

В процессе производства интегральных схем возможно изменение спецификации изготавливаемой схемы, возможно также обнаружение неисправностей на поздних стадиях производства схемы. В таких случаях приходится либо возвращать схемы на более ранние стадии производства, либо выбрасывать их как непригодные к использованию. Подобные ситуации снижают выход годных схем, что весьма нежелательно. Кроме того, в схемы, изготовленных сторонними фирмами, могут быть внедрены вредоносные подсхемы (Trojan Circuits (TCs)) с целью разрушения системы или извлечения конфиденциальной информации. Срабатывание вредоносной подсхемы также может рассматриваться как проявление неисправности на линии связи, к которой присоединён выход вредоносной подсхемы.

Для снижения вероятности возникновения трудно обнаружимой неисправности или внедрения вредоносной подсхемы предлагается использовать частично-программируемые логические схемы [1,2]. Такие схемы представляют собой схемы, в которой подсхемы из традиционных вентилей соединены с программируемыми блоками памяти (LUT). Сначала проектируется схема из вентилей. Затем некоторые её подсхемы покрываются программируемыми блоками. Это даёт возможность либо полностью исключить возникновение неисправности (в случае, если покрываемая линия оказывается внутри программируемого блока), либо замаскировать эту неисправность в случае, если эта линия соединена со входом в блок, и имеется возможность сделать эту линию связи несущественной.

## 1. Постановка задачи

Рассмотрим комбинационную схему $C$ (комбинационную составляющую последовательностной схемы), состоящую из вентилей с одним или двумя входами. Каждому внутреннему полюсу в общем случае соответствует частичная булева функция, которая определяется множествами единичных и нулевых наборов $M_1(f_v)$ и $M_0(f_v)$. Частичность функции возникает из-за того, что на некоторых наборах значений входных переменных (входных наборах) схемы соответствующее значение внутреннего полюса может не влиять на значения выходов схемы. Частичная функция, сопоставляемая внутреннему полюсу, определяется только на тех входных наборах, на которых смена соответствующего значения в рассматриваемом полюсе меняет значение на выходах схемы. Проявление неисправности внутреннего полюса схемы означает искажение частичной булевой функции этого полюса в области её определения. Предполагается, что включение вредоносной подсхемы наиболее вероятно для линий связи с низкой наблюдаемостью.

Рассмотрим частичную функцию $f_1$ и полностью определенную функцию $f_2$, представленные парами множеств единичных и нулевых наборов значений своих переменных: $M_1(f_1)$, $M_0(f_1)$; $M_1(f_2)$, $M_0(f_2)$. Будем говорить, что полностью определенная функция $f_2$ реализует частичную функцию $f_1$, если выполняются условие: пересечения множеств $M_1(f_1)$, $M_0(f_2)$ и множеств $M_0(f_1)$, $M_1(f_2)$ пусты. Это значит, что $M_1(f_2)$ содержит $M_1(f_1)$ и $M_0(f_2)$ содержит M0($f_1$).

Пусть задана комбинационная схема и множество линий связи, в которые наиболее вероятно внедрение вредоносной подсхемы. Требуется покрыть схему программируемыми блоками таким образом, чтобы, по возможности, как можно больше линий из

этого множества оказалось внутри программируемого блока. При этом имеется ограничение на число входов в ПЛБ. Оставшиеся линии связи маскируются путём перепрограммирования программируемого блока, если у этого блока имеется свободный вход.

## 2. Вероятностные оценки наблюдаемости линий связи логической схемы

Под наблюдаемостью линии связи схемы будем понимать возможность наблюдения смены единичного (нулевого) значения линии связи схемы на одном из ее выходов. Соответственно, чем ниже наблюдаемость полюса, тем менее определена частичная функция. Предполагается, что включение вредоносной подсхемы наиболее вероятно к линии с низкой наблюдаемостью. В множество линий связи, которые требуется покрыть ПЛБ, включаются те линии, наблюдаемость которых ниже некоторого порогового значения.

Найдем вероятностную оценку наблюдаемости, используя ROBDD-графы. Вычисление оценки наблюдаемости линии $l$ комбинационной схемы $C$ (комбинационной составляющей последовательностной схемы) относительно $i$-го выхода сводится к построению ROBDD-графа $R\left(C_l^i\right)$ подсхемы $C_v^i$, сопоставляемой этому выходу, в предположении, что линия $l$ является входом соответствующей одно выходной подсхемы. Граф реализует функцию $f_i\left(l,x_1,\ldots,x_n\right)$, причем, переменная $l$ используется в качестве первой в разложении Шеннона при построении графа. Далее строится функция наблюдаемости относительно $i$-го выхода:

$$D_l f_i = f_i\left(0,x_1,\ldots,x_n\right) \oplus f_i\left(1,x_1,\ldots,x_n\right).$$

Эта функция зависит от входов схемы и принимает единичное значение на тех наборах, на которых изменения значений переменной $l$ меняет $i$-й выход схемы.

Введём обозначения: $f_i^{l=0} = f_i\left(0,x_1,\ldots,x_n\right)$, $f_i^{l=1} = f_i\left(1,x_1,\ldots,x_n\right)$. В силу тождества $a \oplus b = a\bar{b} \vee \bar{a}b$ функцию $D_v f_i$ можно записать в следующем виде:

$$D_l f_i = f_i^{l=0}\,\overline{f_i^{l=1}} \vee \overline{f_i^{l=0}}\, f_i^{l=1}.$$

Построение этой функции сводится к операциям полиномиальной сложности над ROBDD-графами, представляющими функции $f_i^{l=0} = f_i\left(0,x_1,\ldots,x_n\right)$, $f_i^{l=1} = f_i\left(1,x_1,\ldots,x_n\right)$ и их инверсиями.

Для построения функции, представляющей наблюдаемость линии связи схемы относительно всех выходов, выполним операцию логического сложения для всех функций наблюдаемости относительно каждого выхода:

$$f^{obs} = \bigvee_{i=1}^{m}\left(D_l f_i\right) = \bigvee_{i=1}^{m}\left(f_i^{l=0}\,\overline{f_i^{l=1}} \vee \overline{f_i^{l=0}}\, f_i^{l=1}\right),$$

и представим ее в виде ROBDD графа $R^{obs}$.

Считая единичные сигналы входных переменных равновероятными, вычисляем оценку наблюдаемости линии $l$ по графу $R^{obs}$.

Вероятность $p(\eta)$ единичного значения булевой функции η, сопоставляемой некоторой нетерминальной вершине μ ROBDD-графа, вычисляется через вероятности $p\left(\eta_\mu^{x_i=0}\right)$, $p\left(\eta_\mu^{x_i=1}\right)$ единичных значений функций $\eta_\mu^{x_i=0}$ и $\eta_\mu^{x_i=1}$, сопоставляемых дочерним вершинам нетерминальной вершины μ по следующему правилу (нетерминальная вершина μ отмечена переменной $x_i$)

$$p(\eta) = p(x_i)\,p\left(\eta_\mu^{x_i=1}\right) + p\left(\overline{x_i}\right)p\left(\eta_\mu^{x_i=0}\right).$$

### 3. Алгоритм покрытия заданного множества линий связи схемы программируемыми блоками

Сначала опишем вспомогательный алгоритм построения подсхемы с максимальным числом входов $d$, выход которой совпадает с заданной линией связи. Выполним следующие шаги алгоритма.

1. Формируем подсхему из заданной линии и двух соединенных с ней вентилей. Переходим к шагу 2.
2. Если число входов в текущую подсхему равно максимальному количеству входов $d$, то переходим к шагу 5. Иначе – на шаг 3.
3. Если есть линия с низкой наблюдаемостью, являющаяся входом в текущую подсхему, то определяется возможность включения её в эту подсхему в качестве внутренней линии. Пусть рассматриваемая подсхема имеет $m$ входов, а элемент, выходом которого является текущая линия, имеет $n$ входов. После подключения линии к подсхеме в ней будет $m+n-1$ входов. Следовательно, присоединение возможно только при условии $m+n-1 \le d$. Если данное ограничение выполнено, то эта линия вместе с вентилем, из которого она исходит, включается в эту подсхему и становится её частью. Далее переходим к шагу 2. Если же больше нет линий с низкой наблюдаемостью, которые являются входами в подсхему, то выполняем шаг 4.
4. Рассматривается линия, являющаяся выходом текущей подсхемы. Если её наблюдаемость выше порогового значения, то переходим на шаг 5. Иначе, если её наблюдаемость низка, то также пытаемся сделать эту линию частью подсхемы. Если это невозможно по причине того, что число входов в текущей подсхеме уже равно $d$, то переходим к шагу 5. Иначе, эта линя вместе с элементом, в который она входит, становится частью текущей подсхемы. Далее возвращаемся на шаг 2.
5. Работа алгоритма завершена, получена подсхема, которая будет покрыта программируемым блоком.

Теперь приведём алгоритм покрытия заданного множества линий связи схемы программируемыми блоками. Разделим элементы комбинационной схемы на уровни следующим образом: к 1-му уровню отнесём элементы, входы которых соединены непосредственно со входами схемы, к ($i$+1)-му уровню отнесём элементы, входы которых соединены с выходами элементов $i$-го и меньших уровней. Имеется список линий связи, значение оценки наблюдаемости которых ниже некоторой пороговой величины. Упорядочим этот список по неубыванию уровней вентилей, хотя бы один из входов которых соединен с концом линии связи из заданного списка, а при равных условиях упорядочиваем линии связи по неубыванию уровней вентилей выходы которых соединены с линиями связей из заданного списка. Пусть имеется упорядоченный список линии связи. Алгоритм покрытия:

1. Просматривается список линий связи. Ищется первая линия, которая не является входом или выходом программируемого блока. Если таких линий нет, то переходим к шагу 4. Иначе выбираем эту линию и переходим на шаг 2.
2. К выбранной линии применяется алгоритм построения подсхемы из этой линии с максимальным числом входов $d$. Из списка удаляются все линии, ставшие частью подсхемы. Переходим к шагу 3.
3. Рассматривается линия связи, выходящая из полученной подсхемы. Если она имеет низкую наблюдаемость, не является входом в ПЛБ, и другая линия связи, являющаяся выходом элемента, входом в который является рассматриваемая линия, также имеет наблюдаемость ниже порогового значения, то ко второй линии применяется алгоритм построения подсхемы из этой линии, но уже с максимальным числом входов $d-1$. Оставшийся вход будет использован для маскирования

неисправности первой линии в случае её обнаружения. После построения подсхемы шаг 3 повторяется, при этом рассматривается линия уже из новой подсхемы. Если линия, выходящая из подсхемы, имеет достаточно высокую наблюдаемость, то возвращаемся к шагу 1.

4. Работа алгоритма завершена, получена частично-программируемая комбинационная схема, состоящая из вентилей и программируемых блоков.

В случае, если некоторые линии заданного множества принадлежат точкам ветвления, то предлагается поступить следующим образом: элемент, выход которого соединён с несколькими линиями, дублируется. При этом рассматриваемая отделяемая линия связи соединяется с концом одного из элементов, а все оставшиеся линии, которые образовывали ветвление – с концом другого. Также дублируются линии, соединённые с входами вентиля, выход которого являлся точкой ветвления. Необходимо вычислить значения наблюдаемости для новых линий, а также заново вычислить наблюдаемости линий, которые были дублированы, т.к. значения наблюдаемости этих линий могли измениться. Во множество покрываемых линий могут добавиться новые линии связи. Этот процесс повторяется до тех пор, пока все линии с низкой наблюдаемостью не перестанут быть точками ветвления. Далее применяется описанный выше алгоритм.

## 4. Маскирование неисправности линии связи с использованием свободного входа

В [3] возможность маскирования неисправности линии сводится к задаче выполнимости квантифицированной конъюнктивной нормальной формы. Эта задача относится к классу PSPACE-полных задач (в отличие от задачи выполнимости пропозициональной конъюнктивной нормальной формы, которая относится к классу NP-полных задач). Отмечается, что проверка выполнимости квантифицированной конъюнктивной нормальной формы реализуется значительно медленнее, чем проверка выполнимости обычной конъюнктивной нормальной формы. В данной работе предлагается способ маскирования, не требующий решения задачи выполнимости. После покрытия схемы программируемыми блоками могут остаться линии, которые являются выходами ПЛБ, и одновременно входами в вентиль. Кроме того, могут остаться линии связи, которые не находятся внутри ПЛБ, но при этом они соединены с входом программируемого блока. В обоих случаях, если у ПЛБ имеется свободный вход, то возможно маскирование неисправности линии связи. В первом случае ПЛБ со свободным входом реализует функцию упомянутого выше вентиля. Маскирование выполняется следующим образом. Рассмотрим способ маскирования, представленный на рис. 1.

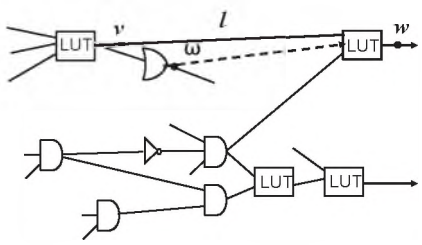


Рис. 1. Маскирование линии $l$, подключенной к входу LUT

Полюс $v$ соединен линией $l$ с входом $u_i$ программируемого блока LUT, покрывающего подсхему $C_{\mathrm{LUT}}$ из вентилей. На рис. 1 линия $l$ выделена жирным шрифтом. Входы подсхемы $C_{\mathrm{LUT}}$ являются либо входными, либо внутренними переменными схемы $C$. Число входных переменных LUT фиксировано, одна из переменных свободна, а функция от оставшихся переменных реализует подсхему из вентилей $C_{\mathrm{LUT}}$, покрывающую соответствующий фрагмент из вентилей схемы $C$. Пусть в линию $l$ включена вредоносная подсхема. Будем маскировать ее, используя свободный вход LUT. В дальнейшем этот LUT будем называть корректирующим. Обозначим его выход символом $w$. Изна-

чально идея маскирования заключалась в следующем: позиция пунктирной линии на рис. 2 заранее не определена, она может соединять свободный вход LUT с выходом другого элемента схемы, при выполнении определенных условий, а именно: функция этого элемента является реализацией частичной функции линии *l*. Однако предварительные исследования показали, что подходящая реализация частичной функции существует очень редко. Поэтому предложено использовать ту же самую реализацию путём добавления резервной линии связи (рис. 2).

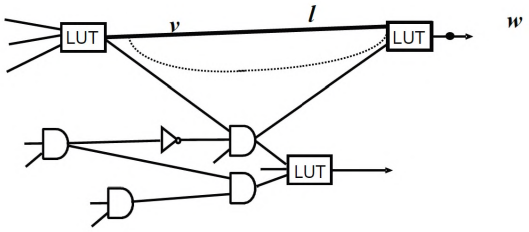


Рис. 2. Свободный вход LUT соединен с линией, дублирующей *l*

Для способа, предложенного на рис. 3, будем записывать в корректирующий программируемый блок единичные наборы полностью определенной функции, реализуемой подсхемой из вентилей, покрытой этим блоком и зависящей от его входных переменных, а именно от его $(m-1)$ переменных. Каждому единичному набору сопоставляется два набора в пространстве $m$ переменных: один с единичным значением переменной $u_m$, другой – с нулевым значением этой переменной. Другими словами, если покрываемая блоком схема реализует функцию $f(u_1,\ldots,u_{m-1})$, то в программируемом блоке формируется функция, в которой переменная $u_m$ является фиктивной (эта переменная сопоставляется пунктирной линии на рис. 1,2):

$$f_{LUT}(u_1,\ldots,u_{m-1},u_m=0)=f(u_1,\ldots,u_{m-1}),\ f_{LUT}(u_1,\ldots,u_{m-1},u_m=1)=f(u_1,\ldots,u_{m-1}).$$

В ходе работы алгоритма, описанного в разделе 3, могут также возникнуть линии связи с низкой наблюдаемостью, которые являются входами ПЛБ с более чем одним свободным входом. В этом случае возможно маскирование нескольких линии связи (а именно – стольких линий, сколько имеется свободных входов). Тогда при программировании этого блока в нём формируется функция с несколькими фиктивными переменными.

В случае, если на линии связи *l* имеет место неисправность или в неё включена вредоносная подсхема, то этот LUT необходимо перепрограммировать таким образом, чтобы переменная $u_i$ стала фиктивной, а переменная $u_m$ – существенной. При перепрограммировании LUT формируем функцию

$$f_{LUT}(u_i=1)\wedge u_m \vee f_{LUT}(u_i=0)\wedge \overline{u_m}$$

из единичных наборов функции корректирующего LUT. Эти наборы получены непосредственно по структуре подсхемы, покрытой корректирующим LUT, и теперь представляют функцию этого LUT в пространстве $u_1,\ldots,u_{i-1},u_{i+1},\ldots,u_m$ его входных переменных. Она отличается от исходной функции LUT тем, что в ней переменная $u_i$ заменена переменной $u_m$, и переменная $u_m$, (как прежде переменная $u_i$) теперь является существенной. С целью маскирования ТС делаем переменную $u_i$ несущественной. Это значит, что каждому единичному набору функции

$$f_{LUT}(u_i=1)\wedge u_m \vee f_{LUT}(u_i=0)\wedge \overline{u_m}$$

необходимо сопоставить два набора в пространстве $m$ переменных: один с единичным значением переменной $u_i$, а другой – с нулевым значением этой переменной. Тогда ТС, подключенный к входу $u_i$, не может изменить корректного поведения схемы $C$.

## 5. Экспериментальные результаты

Алгоритм покрытия линий заданного подмножества программируемыми блоками без свободного входа реализован программно. Выполнены эксперименты на контрольных примерах схем из набора MCNC. При этом число входов в ПЛБ ограничено 8. Для этих схем выбрано пороговое значение, линии связи с оценками наблюдаемости ниже этого значения рассматривались как наиболее вероятные для включения вредоносной подсхемы. К этим схемам применён алгоритм покрытия множества линий с наблюдаемостью ниже заданного значения. На вход программы подаётся схема в формате BLIF, на выходе формируется схема, в которой линии с низкой наблюдаемостью покрыты ПЛБ, также в формате BLIF. Корректность работы программы и эквивалентность исходной и полученной схем проверена с помощью системы логического проектирования ABC применением команды cec, предназначенной для верификации комбинационных схем. В табл. 1,2 приведены данные о количестве вентилей и линий связи в исходной схеме, в схеме, полученной путём разделения точек ветвления, и в схеме, к которой применён алгоритм покрытия программируемыми блоками. В табл. 3 показано, сколько ПЛБ и с каким количеством входов использовано для покрытия линий с низкой наблюдаемостью.

Таблица 1

**Сравнение количества элементов и линий связи до и после разделения ветвящихся линий**

| Название схемы | Пороговое значение | В исходной схеме | | | После расщепления | | |
|---|---|---|---|---|---|---|---|
| | | Вентилей | Линий связи | Линий связи с низкой наблюдаемостью | Вентилей | Линий связи | Линий связи с низкой наблюдаемостью |
| 9symml_rs | 0.01 | 309 | 503 | 38 | 363 | 586 | 73 |
| apex6_rs | 0.01 | 856 | 1573 | 103 | 934 | 1709 | 170 |
| cordic_rs | 0.01 | 78 | 135 | 40 | 96 | 165 | 54 |
| x3 | 0.01 | 1015 | 1752 | 58 | 1035 | 1790 | 68 |
| vda_rs | 0.01 | 801 | 1497 | 41 | 839 | 1560 | 61 |
| term1_rs | 0.01 | 252 | 433 | 132 | 404 | 704 | 230 |
| i4 | 0.01 | 294 | 542 | 36 | 294 | 542 | 36 |
| k2_rs | 0.01 | 1680 | 3078 | 191 | 1900 | 3446 | 301 |
| too_large_rs | 0.001 | 1220 | 2047 | 341 | 1462 | 2463 | 519 |
| t481_rs | 0.001 | 2392 | 4267 | 180 | 2734 | 4852 | 375 |
| frg1_rs | 0.01 | 174 | 303 | 100 | 193 | 339 | 108 |

Таблица 2

**Сравнение количества элементов и линий связи до и после покрытия линий с наблюдаемостью ниже порога**

| Название схемы | Пороговое значение | После расщепления | | | После покрытия | | | |
|---|---|---|---|---|---|---|---|---|
| | | Вентилей | Линий связи | Линий связи с низкой наблюдаемостью | Вентилей | Линий связи | Линий связи с низкой наблюдаемостью | Линий связи, для которых возможно маскирование |
| 9symml_rs | 0.01 | 363 | 586 | 73 | 259 | 513 | 0 | 0 |
| apex6_rs | 0.01 | 934 | 1709 | 170 | 749 | 1556 | 17 | 7 |
| cordic_rs | 0.01 | 96 | 165 | 54 | 41 | 116 | 5 | 4 |
| x3 | 0.01 | 1035 | 1790 | 68 | 946 | 1725 | 3 | 0 |
| vda_rs | 0.01 | 839 | 1560 | 61 | 751 | 1499 | 0 | 0 |
| term1_rs | 0.01 | 404 | 704 | 230 | 147 | 594 | 20 | 8 |
| i4 | 0.01 | 294 | 542 | 36 | 234 | 506 | 0 | 0 |
| k2_rs | 0.01 | 1900 | 3446 | 301 | 1491 | 3145 | 0 | 0 |
| too_large_rs | 0.001 | 1462 | 2463 | 519 | 906 | 1997 | 53 | 30 |
| t481_rs | 0.001 | 2734 | 4852 | 375 | 2194 | 4477 | 0 | 0 |
| frg1_rs | 0.01 | 193 | 339 | 108 | 75 | 240 | 9 | 6 |

Таблица 3

**Сравнение количества элементов и линий связи до и после разделения ветвящихся линий**

| Название схемы | Входов | Выходов | Программируемых блоков | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | Всего | 2-входовых | 3-входовых | 4-входоых | 5-входовых | 6-входовых | 7-входовых | 8-входовых |
| 9symml_rs | 9 | 1 | 31 | 6 | 15 | 9 | 1 | 0 | 0 | 0 |
| apex6_rs | 135 | 99 | 36 | 3 | 8 | 7 | 3 | 3 | 4 | 8 |
| cordic_rs | 23 | 2 | 7 | 0 | 0 | 2 | 0 | 0 | 2 | 3 |
| x3 | 135 | 99 | 24 | 2 | 10 | 6 | 2 | 2 | 0 | 2 |
| vda_rs | 17 | 39 | 27 | 5 | 10 | 10 | 1 | 0 | 1 | 0 |
| term1_rs | 34 | 10 | 47 | 3 | 15 | 3 | 6 | 4 | 3 | 13 |
| i4_rs | 192 | 6 | 24 | 0 | 16 | 4 | 4 | 0 | 0 | 0 |
| k2_rs | 45 | 45 | 109 | 7 | 50 | 29 | 7 | 11 | 3 | 1 |
| too_large_rs | 38 | 3 | 90 | 15 | 13 | 11 | 5 | 5 | 18 | 23 |
| t481_rs | 16 | 1 | 165 | 2 | 82 | 65 | 16 | 0 | 0 | 0 |
| frg1_rs | 28 | 3 | 19 | 2 | 3 | 2 | 1 | 1 | 1 | 9 |

Из табл. 3 видно, что предложенный алгоритм способен значительно снизить число линий с низкой наблюдаемостью в схеме, и, таким образом, снизить вероятность внедрения вредоносной схемы или возникновения трудно обнаружимой неисправности. В некоторых случаях возможно полное покрытие линий программируемыми блоками. В остальных же случаях неисправности на некоторых линиях связи могут быть замаскированы путём перепрограммирования соответствующего блока. Также можно заметить, что число вентилей и линий связи в схеме уменьшается после покрытия, несмотря на то что после разделения всех точек ветвления сложность схемы возрастает.

### Заключение

Рассматриваются комбинационные схемы из вентилей. Для таких схем предложен алгоритм покрытия линий программируемыми блоками с ограниченным числом входов с целью снижения возможностей включения в эти линии вредоносных подсхем. Разработана программная реализация описанного алгоритма. Проведены эксперименты на контрольных примерах. Показано, что во многих случаях возможно значительное снижение риска внедрения вредоносной подсхемы (ТС), а также нейтрализация её действия.

### ЛИТЕРАТУРА


1. *Yamashita S., Yoshida H., Fujita M.* Increasing yield using partially-programmable circuits // Workshop on Synthesis and System Integration of Mixed Information technologies (SASIMI). – 2010. – P. 237–242.

2. *Jo S., Matsumoto T., Fujita M.* SAT-based automatic rectification and debugging of combinational circuits with LUT insertions // Proc. Of IEEE Asian Test Symposium. – 2012. P. 19–24.

3. *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.


## ПОСТРОЕНИЕ ЧАСТИЧНЫХ ФУНКЦИЙ ДЛЯ ПОДСХЕМ


**А.С. Ложкин, А.Ю. Матросова, Э.С. Фатеева**

*Томский государственный университет*


### Введение

Производство логических схем высокого уровня интеграции и высокого быстродействия связано с большими трудозатратами. К сожалению, даже на последних этапах проектирования возможно обнаружение неисправности в схеме. Возвращение к ранним этапам проектирования является дорогостоящим, а выбрасывание неисправной схемы приводит к снижению выхода годных схем, что отражается на себестоимости производимых схем. Конечно, маскировать неисправность можно, используя дублирование в