УДК 519.714

Верификация логических описаний комбинационных устройств

Л.Д. Черемисинова

Объединенный институт проблем информатики Национальной академии наук Беларуси,

cld@newman.bas-net.by

***Аннотация* — Описываются методы и программные средства, предназначенные для верификации логических описаний проектируемых комбинационных устройств и позволяющие обнаруживать ошибки проектирования на ранних этапах. Реализованы два подхода к решению задачи верификации: на основе моделирования комбинационной схемы и на основе сведения задачи верификации к проверке выполнимости конъюнктивной нормальной формы.**

***Ключевые слова* — автоматизация проектирования, верификация, моделирование, выполнимость КНФ.**

I. Введение

Одним из наиболее эффективных методов повышения качества и надежности оборудования является создание средств автоматизации, гарантирующих корректность проектируемых устройств. Это может быть обеспечено путем верификации проектных решений на всех стадиях проектирования. Задача верификации заключается в доказательстве поведенческого соответствия двух описаний одного и того же устройства – проверке, находятся ли они в отношении эквивалентности (если оба описания полностью определены) или реализации (если исходное описание не полностью определено).

По мере возрастания сложности проектируемых устройств функциональная верификация становится все более необходимым и дорогим этапом процесса проектирования. Официальные издания проектных компаний утверждают, что коллектив разработчиков цифровой аппаратуры тратит до 70% всего времени проектирования и ресурсов на функциональную верификацию, и, если сложность проекта аппаратуры удваивается, то усилия, затрачиваемые на верификацию, учетверяются [1]. В связи с этим верификация, позволяющая достаточно рано выявить ошибки проектирования, становится все более и более ответственным этапом.

Задача верификации в традиционной постановке состоит в проверке функциональной эквивалентности пары описаний одного и того же устройства, которые получаются в ходе проектирования (оптимизации или структурной реализации). В литературе традиционно рассматривается случай, когда оба сравниваемых описания функционально полностью определены, решению этой задачи посвящены многочисленные научные публикации, ссылки на которые можно найти в [1–3].

В данной работе, наряду с традиционно рассматриваемым случаем, предлагаются также средства для решения задачи верификации для более общего случая, когда заданная функциональность проектируемого устройства не полностью определена. Такая ситуация обычно возникает, когда существуют такие комбинации значений входных сигналов проектируемого устройства, которые никогда не появятся при его нормальной работе. В этом случае при решении задачи верификации достаточно рассмотреть только возможные сценарии поведения верифицируемого устройства и проверить, имеют ли его выходные реакции специфицированные значения.

Элементный базис сравниваемых схем может содержать элементы библиотеки КМОП СБИС, а также элементы, реализующие простые функции (типа И, ИЛИ, И-НЕ и т.д.). Сравниваемые описания задаются на языке SF функционально-структурных описаний [4]. Описываемые программные средства ориентированы на верификацию логических описаний сложных устройств.

Представляемые в работе алгоритмические и программные средства верификации входят в качестве составной части в программный комплекс [5] для автоматизации проектирования интегральных микросхем с пониженным энергопотреблением, выполняемых по КМОП технологии.

II. Функциональные возможности комплекса верификации и входная информация

Комплекс верификации включает в себя алгоритмические и программные средства для решения задачи верификации, когда сравниваются:

1) два полностью определенных описания, т.е. по сути, две структурные реализации схемы в некоторых, может быть и разных, базисах;

2) два описания, одно из которых, исходное, не полностью определено (область определения не покрывает все булево пространство), а второе, порожденное в процессе проектирования (оптимизации, синтеза), полностью определено, т.е. задает, например, логическую схему.

В первом случае верификация заключается в проверке эквивалентности между двумя схемными реализациями одного и того же устройства. Во втором случае верификация заключается в проверке отношения реализуемости между описаниями (функциональными или структурными), т.е. эквивалентности описаний на области определения "наименее определенного" из двух описаний – исходного описания.

Исходное описание верифицируемого устройства может представляться в виде:

– системы полностью определенных булевых функций, заданных в виде дизъюнктивных нормальных форм (ДНФ);

– системы частично определенных булевых функций (ЧБФ) (к этому виду может быть приведено любое функционально полностью или частично определенное описание);

– многоуровневой схемы из простых вентилей типа И, ИЛИ;

– многоуровневой схемы из элементов библиотеки КМОП СБИС;

– многоблочной структуры, поведение каждого блока которой задается системой ДНФ.

ЧБФ $f(X)$ ($X = \{x_1, x_2, \ldots, x_n\}$) задается множествами U_f^0, U_f^1 и U_f^{dc} интервалов, на которых она принимает соответственно нулевое, единичное или неопределенное значения. Интервал из n-мерного булева пространства задает некоторое множество наборов значений булевых переменных из X и представляется n-компонентным троичным вектором [8].

Система ЧБФ $F(X) = \{f_1(X), f_2(X), \ldots, f_m(X)\}$ представляется множеством многовыходных интервалов $(\boldsymbol{u}_k, \boldsymbol{t}_k)$, каждый из которых задается парой троичных векторов длины n и m. Троичный вектор $\boldsymbol{u}_k$ представляет собой интервал из n-мерного булева пространства наборов значений переменных множества X, троичный вектор $\boldsymbol{t}_i$ задает значения функций $f_i \in F$ на интервале $\boldsymbol{u}_k$. Для каждой функции $f_i \in F(X)$ справедливо: если i-я компонента t_k^i вектора $\boldsymbol{t}_k$ равна 1 или 0, то на всех наборах из интервала $\boldsymbol{u}_k$ функция f_i принимает соответствующее значение; если же $t_k^i =$ "–", то либо f_i принимает разные значения (из {1, 0, –}) по крайней мере на двух наборах из интервала $\boldsymbol{u}_k$, либо ее значение не определено на всем интервале.

Система ЧБФ $F(X)$, заданная множеством I_F многовыходных интервалов $(\boldsymbol{u}_i, \boldsymbol{t}_i)$, может быть представлена парой троичных матриц $\boldsymbol{U}$ и $\boldsymbol{T}$, задающих своими строками многовыходные интервалы. Например, интервалу $(\boldsymbol{u}_i, \boldsymbol{t}_i) = (x_1 x_3 x_5, f_1 f_3)$ соответствуют строки $(0-1-0)$ и $(0-1)$ матриц $\boldsymbol{U}$ и $\boldsymbol{T}$.

Условие реализуемости системы F схемой S, реализующей на своих выходах функции $y_i(X)$, заключается в том, что на области определения каждой функции $f_i(X) \in F$ значения функций f_i и y_i должны совпадать, т.е. для всех $f_i(X) \in F$ и $y_i(X)$ должно выполняться

$$M_{fi}^1 \subseteq M_{yi}^1;\ M_{fi}^0 \subseteq M_{yi}^0,$$

где M_{fi}^1 и M_{yi}^1, M_{fi}^0 и M_{yi}^0 – множества наборов значений булевых переменных из X, на которых функции $f_i(X)$ и $y_i(X)$ принимают единичные и нулевые значения.

При решении задачи проверки реализуемости системы ЧБФ $F(X)$ комбинационной схемой S можно ограничиться частичным анализом сравниваемых описаний – не на всем булевом пространстве переменных из X, а только на области определения функций системы $F(X)$.

III. Представление входной информации

Предполагается, что сравниваемые в процессе верификации описания задают функциональные зависимости между одноименными входными и выходными переменными. Каждое из сравниваемых описаний, принимаемых комплексом, задается в одном из форматов на языке SF функционально-структурных описаний [4]:

– SDF – система ДНФ, задающая систему полностью определенных булевых функций в матричном виде;

– SBF – система ЧБФ в матричном виде;

– LOG – система логических уравнений, задающая двухуровневую или многоуровневую схему (в частном случае и систему ДНФ) из вентилей типа НЕ, И, ИЛИ, исключающее ИЛИ и др.;

– 2-connect – двухуровневое иерархическое описание структурной реализации, блоки которого имеют функциональное описание типа SDF, SBF или LOG, а второй уровень задает связи между ними.

Иерархическое описание может задавать схему из библиотечных элементов КМОП СБИС или многоблочную структуру, каждый из блоков которой реализует систему полностью или частично определенных булевых функций. В том случае, когда производится оптимизация на структурном уровне, исходным описанием может быть схема, заданная в форматах LOG и 2-connect.

Второе из сравниваемых описаний представляется в формате LOG или 2-connect и задает комбинационную схему в базисе многовходовых логических элементов или многоблочную структуру. В обоих случаях порожденное описание можно трактовать как структурную реализацию исходного описания – схему (двухуровневую или многоуровневую).

IV. Алгоритмическая поддержка комплекса

Ранее были предложены два класса методов верификации, эффективных для разных типов сравниваемых описаний и способов их задания.

Методы первого класса основаны на моделировании порожденного описания верифицируемого устройства на области задания исходного описания. Этот подход заключается в подаче двоичных сигналов на входы моделируемой схемы, продвижении их по схеме, активации ее выходов и проверки результатов моделирования с ожидаемыми значениями. Однако при моделировании практически возможен только частичный анализ сравниваемых описаний – на некотором подмножестве входных воздействий (так как их число последних может достигать 2^n, где n – число входных переменных), и моделирование в общем случае не может обеспечить полноту верификации.

В работе [6] предложено обобщение метода двоичного моделирования, позволяющее моделировать поведение схемы на интервалах значений входных переменных. Это значительно повышает порог применимости метода моделирования (по сложности моделируемых описаний), но не решает проблему полноты верификации в силу того, что некоторые интервалы приходится иногда раскрывать до наборов значений переменных.

Методы второго класса основаны на формальном доказательстве функциональной идентичности проектов путем сведения задачи верификации к проверке выполнимости некоторой конъюнктивной нормальной формы (КНФ) [2, 3], отражающей структуру сравниваемых описаний. В отличие от методов моделирования эти методы позволяют обеспечить полноту верификации. Развитию и практическому применению таких методов способствовал наметившийся в последнее десятилетие существенный прогресс в решении задачи выполнимости КНФ.

Методы моделирования эффективны для случая задания системы ЧБФ на наборах значений аргументов или “небольших” интервалах. Методы на основе выполнимости КНФ, наоборот, эффективны для случая задания системы ЧБФ на “крупных” интервалах, охватывающих большие подпространства булева пространства.

В предлагаемых средствах реализован также комбинированный метод, совмещающий в себе методы верификации обоих классов. Основная идея этого метода состоит в использовании сначала моделирования, пока оно эффективно и полно; далее в сложных для моделирования случаях используется метод сведения задачи верификации к проверке выполнимости КНФ.

V. Верификация на основе моделирования

Реализованные в рамках программного комплекса методы верификации посредством моделирования основаны [6, 7] на параллельном моделировании схем сразу на всех наборах или интервалах из области определения системы ЧБФ исходного описания на основе быстрых булевых вычислений над булевыми и троичными векторами большой размерности. Они включают анализ результатов моделирования и выявление интервалов области задания исходного описания, не реализуемых порожденным описанием. Выделяются три типа алгоритмов решения задач верификации на основе моделирования.

1. Двоичное моделирование [6] схемы из вентилей И, ИЛИ, НЕ на области задания исходной системы ЧБФ, представленной множеством многовыходных наборов $(\boldsymbol{b}_k, \boldsymbol{t}_k)$, где $\boldsymbol{b}_k$ – булев вектор.

Моделирование схемы на наборах значений переменных или, в общем случае, на интервалах из области определения системы ЧБФ $F(X)$ исходного описания состоит в последовательной подаче на входы схемы наборов (интервалов) значений входных переменных; вычислении значений сигналов на выходах элементов схемы; сравнении реакций схемы со значениями функций системы $F(X)$ исходного описания.

Выполняется параллельное моделирование предварительно ранжированной многовыходной логической схемы сразу на всех наборах или, в общем случае, на интервалах из области определения системы ЧБФ $F(X)$ [8]. При параллельном моделировании схемы на l интервалах (или наборах) значений переменных состояние каждого ее полюса (включая входные и выходные) представляется троичным (или булевым) вектором размерности l, задающим состояния этого полюса при подаче на входы схемы каждого из l интервалов. При этом для выходного полюса каждого i-го элемента вычисляется локальная функция $\varphi_i(\mathbf{z}_{1i}, \mathbf{z}_{2i}, \ldots, \mathbf{z}_{ki})$ путем выполнения покомпонентной операции φ_i [6] над троичными (или булевыми) векторами $\mathbf{z}_{1i}, \mathbf{z}_{2i}, \ldots, \mathbf{z}_{ki}$ длины l, приписанными входным полюсам элемента.

2. Троичное моделирование [6] схемы из вентилей И, ИЛИ, НЕ на области задания исходной системы ЧБФ, представленной множеством многовыходных интервалов $(\boldsymbol{u}_k, \boldsymbol{t}_k)$, где $\boldsymbol{u}_k$ и $\boldsymbol{t}_k$ – троичные векторы.

Особенностью троичного моделирования является то, что j-я компонента вектора-результата, вычисляемого для некоторого полюса схемы, может иметь неопределенное значение не только в том случае, когда функция, реализуемая полюсом, имеет неопределенное значение на всем j-м интервале исходной системы ЧБФ, но и в том, когда функция имеет разные значения на разных наборах этого интервала. Эта неоднозначность может проявляться на выходах моделируемой схемы, приводя к тому, что для некоторых исходных интервалов невозможно дать однозначный ответ на вопрос, реализуются ли они схемой. Проблема может быть разрешена путем повторного, но уже двоичного моделирования схемы на наборах, входящих в анализируемый интервал, или путем проверки реализуемости оставшейся части исходного описания формальными методами верификации на основе проверки выполнимости КНФ.

Преимущество троичного моделирования над двоичным состоит в том, что интервалы не требуется

расщеплять на наборы, что позволяет резко сократить длину требуемых для моделирования векторов и повысить быстродействие моделирования. Однако по результатам троичного моделирования не всегда удается однозначно ответить на вопрос, реализуется ли система функций схемой на всех интервалах, т.е. может остаться множество интервалов, для которых необходим дополнительный анализ.

3. Двоично-троичное моделирование [1] схемы из вентилей И, ИЛИ, НЕ на области задания исходной системы ЧБФ, представленной множеством многовыходных интервалов $(\boldsymbol{u}_k, \boldsymbol{t}_k)$, где $\boldsymbol{u}_k$ и $\boldsymbol{t}_k$ – троичные векторы. Сначала выполняется троичное моделирование (алгоритм 2), затем выделяется область, реализуемость которой остается под вопросом, после чего интервалы $(\boldsymbol{u}_k, \boldsymbol{t}_k)$ этой области раскрываются до наборов $(\boldsymbol{b}_i, \boldsymbol{t}_i)$ и выполняется двоичное моделирование (алгоритм 1).

VI. Формальная верификация на основе проверки выполнимости КНФ

Верификация посредством сведения к задаче проверки выполнимости КНФ основана на построении такой КНФ, выполнимость которой свидетельствует о нарушении реализуемости (эквивалентности) исходного описания порожденным. КНФ называется выполнимой, если существует такое присваивание значений ноль и единица ее аргументам, которое обращает КНФ в единицу.

Выделяется восемь алгоритмов решения задач из области верификации на основе проверки выполнимости КНФ.

1. Проверка выполнимости КНФ с помощью одного из известных SAT-решателей [12].

2. Проверка эквивалентности пары полностью определенных описаний: системы ДНФ, комбинационной схемы из вентилей или библиотечных элементов.

Традиционно задача формальной верификации состоит в проверке функциональной эквивалентности пары структурных реализаций одного и того же устройства. При проверке эквивалентности комбинационных схем обе верифицируемые схемы преобразуются в одну комбинационную схему, называемую схемой сравнения [3]. Эта схема получается путем объединения пар одноименных входов сравниваемых схем и подачи пар их одноименных выходов на двухвходовые элементы “исключающее ИЛИ”, выходы которых собираются на элемент ИЛИ. Константа 0 на выходе элемента ИЛИ появляется тогда и только тогда, когда исходные схемы эквивалентны. Для схемы сравнения S строится КНФ разрешения C_S, которая задает все возможные допустимые комбинации сигналов на всех полюсах элементов схемы и которая проверяется на выполнимость – для эквивалентных схем КНФ C_S не выполнима.

3. Построение КНФ разрешения комбинационной схемы из вентилей, библиотечных элементов [10].

4. Построение КНФ разрешения многоблочной структуры (в частном случае системы ДНФ).

КНФ разрешения C_S схемы представляет собой объединение (операцией конъюнкции) КНФ разрешения всех элементов схемы. При ее построении для каждого логического элемента схемы, реализующего функцию $f(z_1, z_2, \ldots, z_k)$ от k своих входов, вводится булева переменная y, строится функция разрешения $\varphi(y, f) = y \sim f(z_1, z_2, \ldots, z_k)$ и соответствующая этой функции КНФ разрешения элемента. Например, КНФ разрешения k-входовых элементов И и ИЛИ имеют вид:

$$\varphi^{\wedge}(y, z_1, z_2, \ldots, z_k) = (z_1 \vee \bar{y}) \wedge (z_2 \vee \bar{y}) \wedge \ldots$$
$$\wedge (z_k \vee \bar{y}) \wedge (\bar{z}_1 \vee \bar{z}_2 \vee \ldots \vee \bar{z}_k \vee y);$$
$$\varphi^{\vee}(y, z_1, z_2, \ldots, z_k) = (\bar{z}_1 \vee y) \wedge (\bar{z}_2 \vee y) \wedge \ldots$$
$$\wedge (\bar{z}_k \vee y) \wedge (z_1 \vee z_2 \vee \ldots \vee z_k \vee \bar{y}).$$

5. Проверка реализуемости исходной системы ЧБФ комбинационной схемой или многоблочной структурой.

В том случае, когда исходное описание содержит неопределенность, оно приводится к виду системы ЧБФ. Искомая КНФ, проверяемая на выполнимость для проверки реализуемости, состоит из:

– КНФ разрешения C_S схемы, которая описывает все допустимые (разрешенные) комбинации сигналов на полюсах элементов схемы (или многоблочной структуры), заданной порожденным описанием;

– КНФ запрета C_P, которая описывает все комбинации сигналов, которые противоречат функциям исходной системы ЧБФ.

В [9] доказано, что система ЧБФ $F(X)$ реализуется комбинационной схемой, если и только если КНФ $C = C_S \wedge C_P$ не выполнима.

6. Построение КНФ запрета системы ЧБФ на основе единичного кодирования многовыходных интервалов [10], позволяющего получать КНФ с дизъюнктами небольшого ранга.

7. Построение КНФ запрета системы ЧБФ на основе логарифмического и интервального кодирования многовыходных интервалов [11], позволяющего значительно сократить число кодирующих переменных.

Комбинационная схема реализует систему ЧБФ, если она реализует каждый ее многовыходной интервал, или система ЧБФ не реализуется комбинационной схемой, если ею не реализуется хотя бы один многовыходной интервал. В терминах КНФ разрешения многовыходной интервал $(\boldsymbol{u}_i, \boldsymbol{t}_i)$ реализуется схемой, если присваивание значений переменным, выполняющее конъюнкцию $\boldsymbol{u}_i \wedge \bar{\boldsymbol{t}}_i$, не является выполняющим для КНФ разрешения схемы [10]. КНФ, задающая условие нереализуемости интервала, называется КНФ запрета этого интервала.

Если $\boldsymbol{u}_i = x_1^i x_2^i \ldots x_{n_i}^i$, $\boldsymbol{t}_i = f_1^i f_2^i \ldots f_{m_i}^i$, то КНФ P_i запрета интервала $(\boldsymbol{u}_i, \boldsymbol{t}_i)$ состоит из $n_i + 1$ дизъюнктов:

$$P_i = x_1^i x_2^i ... x_{n_i}^i (\overline{f}_1^i \vee \overline{f}_2^i \vee ... \vee \overline{f}_{m_i}^i).$$

Условие нереализуемости системы ЧБФ $F(X)$ задается ее функцией запрета $P = P_1 \vee P_2 \vee \ldots \vee P_l$, где l – мощность множества многовыходных интервалов области определения системы. Представление функции P в общем случае не является КНФ, и чтобы применить SAT-решатель, необходимо функцию запрета P преобразовать к виду КНФ C_P. В [10] предложен метод линейной сложности преобразования функции запрета P к виду КНФ C_P путем кодирования интервалов из области задания исходной системы ЧБФ.

После кодирования функция P принимает вид КНФ $C_P = (P_1^k \wedge P_2^k \wedge \ldots \wedge P_l^k) \wedge Q(W)$, где P_1^k – кодированная форма КНФ запрета i-го интервала, $Q(W)$ – КНФ выбора, обеспечивающая выполнимость КНФ $C = C_S \wedge C_P$ в случае, если выполняется хотя бы одна КНФ $C_S \wedge P_i^k$, и невыполнимость, если все КНФ $C_S \wedge P_i^k$ не выполнимы. Вид этой функции [9] зависит от типа кодирования. Например, при кодировании интервалов кодами w_i длины 1 $Q(W) = \overline{w}_1 \vee \overline{w}_2 \vee \ldots \vee \overline{w}_l$, а КНФ запрета интервалов

$$P_i^k = x_1^i x_2^i ... x_{n_i}^i (\overline{f}_1^i \vee \overline{f}_2^i \vee ... \vee \overline{f}_{m_i}^i \vee w_i).$$

8. Проверка реализуемости исходной системы ЧБФ комбинационной схемой или многоблочной структурой с помощью комбинированного метода.

На основе проведенных исследований, позволивших определить области предпочтительного использования методов верификации каждого класса [7, 9], была показана целесообразность сочетания методов моделирования и сведения к проверке выполнимости КНФ при верификации сложных описаний. Комбинированный метод основан на верификации описаний по частям, используя в зависимости от типов и сложности сравниваемых частей описаний и степени их определенности один из подходящих типов верификации (на основе моделирования или проверки выполнимости КНФ).

VII. Работа программного комплекса верификации

После получения пары сравниваемых логических описаний в упомянутых в разд. 2 форматах, программный комплекс работает следующим образом. Исходное описание приводится к виду системы ЧБФ $F(X)$, если оно содержит неопределенность, или к виду логической схемы во внутренней форме.

Далее, если оба сравниваемых описания не содержат неопределенности, то речь идет о проверке их эквивалентности. Каждое из описаний в этом случае может быть представлено логической схемой. Эти схемы объединяются в одну схему сравнения, для которой формируется КНФ разрешения и производится проверка ее выполнимости. Если КНФ не выполнима, то сравниваемые описания функционально эквивалентны, в противном случае имеет место неэквивалентность описаний, а значит, в итоге, ошибка проектирования.

Если исходное описание содержит неопределенность, то речь идет о проверке реализуемости исходного описания с неопределенностью порожденным описанием. Порожденное описание представляет собой логическую схему или многоблочную структуру (в частном случае состоящую из одного блока, задающего систему ДНФ). В этом случае порожденное описание предварительно преобразуется в логическую схему, предназначенную для моделирования [6].

Далее производится параллельное троичное моделирование [6] логической схемы S одновременно на всех интервалах, на которых определена система ЧБФ F исходного описания. Преимущество такого моделирования над двоичным моделированием состоит в том, что: 1) интервалы не требуется расщеплять на наборы, что позволяет резко сократить длину требуемых для моделирования векторов; 2) элементы моделируемой логической схемы просматриваются только один раз. Однако по результатам троичного моделирования не всегда удается однозначно ответить на вопрос, реализуется ли система F схемой S на всех интервалах, т.е. может остаться некоторое множество I интервалов системы F, для которых необходим дополнительный анализ. Для такого анализа используется метод [9] сведения к задаче проверки выполнимости КНФ C, которая состоит из двух КНФ: КНФ C_P запрета, формируемая для множества I многовыходных интервалов, и КНФ C_S разрешения логической схемы S. КНФ $C = C_P \wedge C_S$ выполнима тогда и только тогда, когда схема S не реализует хотя бы один интервал из I, а значит, не реализует и систему F.

Если в процессе верификации будет установлена неэквивалентность или нереализуемость исходного описания, то производится диагностика ошибок в порожденном описании, заключающаяся в нахождении фрагмента исходного описания, не реализуемого порожденным описанием.

VIII. Структура программного комплекса верификации

Программный комплекс состоит из семи основных модулей для решения следующих задач:

– синтаксической проверки и преобразования исходного описания верифицируемого устройства, заданного в форматах SDF, SBF, LOG или 2-connect языка SF, к виду системы ЧБФ или логической схемы во внутренней форме;

– синтаксической проверки и преобразования порожденного описания многоблочной структуры, заданного в форматах SDF, LOG или 2-connect языка SF, к виду логической схемы во внутренней форме;

– построения КНФ, готовой для проверки выполнимости SAT-решателем: 1) КНФ запрета исходной системы ЧБФ, 2) КНФ разрешения многоблочной структуры для различных случаев задания ее блоков,

3) КНФ схемы сравнения (для случая, когда сравниваемые описания задают комбинационные схемы);

– проверки выполнимости сформированной КНФ с помощью SAT-решателя, в качестве которого используется Minisat [12];

– двоичного или троичного параллельного моделирования логической схемы на наборах или интервалах значений аргументов системы ЧБФ;

– управления процессом верификации, которое заключается в: 1) распознавании типов исходных данных; 2) выборе подходящего метода верификации, наиболее эффективного для каждого конкретного случая; 3) диагностике ошибок, если таковые имеются; 4) организации связи между описанными модулями.

IX. Результаты исследования программных средств

Программные средства разрабатывались в среде программирования Visual C++ 6.0. Программный комплекс обеспечивает: 1) анализ верифицируемых пар проектных решений и определение типа задачи верификации; 2) выбор наиболее эффективного пути решения задачи верификации в зависимости от способа задания, типов проектных решений и их сложности; 3) преобразование форматов задания проектных решений; 4) настройку на базис библиотечных элементов.

Было проведено сравнение по быстродействию предложенных программных средств верификации со средствами верификации, реализованными в рамках программных комплексов “Modelsim” [13] фирмы Mentor Graphics и “СиВер” [14] синтеза и верификации, разработанного в ОИПИ НАН Беларуси. Каждый рассматриваемый пример включал систему полностью или частично определенных булевых функций, заданных на наборах или интервалах значений входных переменных, и комбинационную схему из элементов КМОП библиотеки, полученную в результате проектных операций программного комплекса [5]. Часть примеров взята из практики проектирования промышленных контроллеров, часть сгенерирована псевдослучайным образом, а остальные примеры (benchmarks) взяты из источников в сети Интернет [15].

Результаты исследования показали (сравнение с “Modelsim” приведено в [16]), что предложенный комплекс средств верификации обладает высоким быстродействием, выполняя верификацию рассматриваемых примеров на порядки быстрее по сравнению с программой VHDL-моделирования из “Modelsim” и средствами верификации из “СиВер”. Например, верификация примера RCKL (SBF: 32 аргумента, 7 функций, 96 интервалов; схема: 745 библиотечных элемента) с помощью предложенных средств выполнена за 0,05 с, с помощью СиВер – за 5,85 с, “Modelsim” потребовал более чем 10000 с.

Описанные средства верификации ориентированы на тестирование логических описаний верифицируемых устройств большой размерности (порядка 100 аргументов и 10000 интервалов системы булевых функций) и включены в состав программного комплекса автоматизации проектирования логических схем в библиотечном базисе, оптимизированных по энергопотреблению [4].

Литература

[1] Wiemann A. Standardized functional Verification. Springer, San Carlos, CA USA. 2008. 289 p.

[2] Ganai M., Gupta A. SAT-Based Scalable Formal Verification Solutions. New York: Springer-Verlag. 2007. 338 p.

[3] Kuehlmann A., Cornelis A.J. van Eijk. Combinational and Sequential Equivalence Checking // in: Logic synthesis and Verification (editors S.Hassoun, T.Sasao, R.K.Brayton). Kluwer Academic Publishers. 2002. P. 343–372.

[4] Бибило П.Н. Кремниевая компиляция заказных СБИС. Минск: Ин-т техн. кибернетики АН Беларуси, 1996. 268 с.

[5] Бибило П.Н., Черемисинова Л.Д., Кардаш С.Н. и др. Синтез логических КМОП схем с пониженным энергопотреблением // Проблемы разработки перспективных микро- и наноэлектронных систем - 2012. Сб. трудов / под ред. акад. РАН А.Л. Стемпковского. М.: ИППМ РАН, 2012. С. 73-78.

[6] Cheremisinova L., Novikov D. Simulation-based approach to verification of logical descriptions with functional indeterminacy // Information Theories & Applications (IJ ITA). 2008. V. 15. № 3. P. 218-224.

[7] Новиков Д.Я., Черемисинова Л.Д. Исследование методов верификации описаний с функциональной неопределенностью на основе моделирования // Автоматика и вычислительная техника. 2012. № 5. С. 13-25.

[8] Закревский А.Д., Поттосин Ю.В., Черемисинова Л.Д. Логические основы проектирования дискретных устройств. М.: Физматлит, 2007.

[9] Черемисинова Л.Д., Новиков Д.Я. Формальная верификация описаний с функциональной неопределенностью на основе проверки выполнимости конъюнктивной нормальной формы // Автоматика и вычислительная техника. 2010. № 1. С. 5–16.

[10] Cheremisinova L., Novikov D. SAT-Based Approach to Verification of Logical Descriptions with Functional Indeterminacy // 8th Int. Workchop on Boolean problems, Freiberg (Sachsen). Sept. 18–19, 2008. P. 59-66.

[11] Cheremisinova L., Novikov D. SAT-based method of verification using logarithmic encoding // Information Science and Computing. 2009. № 15. P. 107-114.

[12] Een N., Sorensson N. An Extensible SAT-solver // Theory and Applications of Satisfiability Testing (SAT 2003). Proc. of the Intern. Conf. Santa Margherita Ligure, Italy. May 5–8, 2003. Microsoft Research. P. 502-518.

[13] ModelSim: HDL Simulation / Mentor Graphics Corporation [Electronic resource]. – Mode of access: http://www.mentor.com/products/fv/modelsim/. Date of access: 2014.

[14] Бибило П.Н., Кардаш С.Н., Романов В.И. СиВер – система синтеза и верификации комбинационных логических схем // Информатика. 2006. № 4 (12). С. 79-87.

[15] Espresso examples / University of California Berkeley [Electronic resource]. – Mode of access: http://www1.cs.columbia.edu/~cs4861/sis/espresso-examples/ex. Date of access: 2014.

[16] Бибило П.Н., Новиков Д.Я. Верификация логических схем, реализующих системы частичных булевых функций // Информатика. 2011. № 3. С. 68-76.