

Использование модификации алгоритма работы сетей Петри для функционального моделирования логических схем, представленных на вентильном уровне

Д.А. Булах, Г.Г. Казённов, А.В. Лапин

*Национальный исследовательский университет «МИЭТ»,
г. Москва, Россия*

dima@pkims.ru

Основным алгоритмом моделирования цифровых схем, применяемых в цифровых симуляторах, является алгоритм событийного моделирования. Однако неявный выбор очередности срабатывания одновременно переключающихся сигналов может приводить к возникновению отличий в получаемых результатах моделирования.

Предложен новый алгоритм событийного функционального моделирования цифровых интегральных схем, основанный на использовании модифицированного математического аппарата сетей Петри. Аппарат сетей Петри дает возможность имитировать параллельные конструкции с помощью последовательных инструкций. Поэтому нет необходимости разделять события с помощью дельта-задержки. Описанный подход позволяет устранить неоднозначность переключений сигналов, происходящих в один момент времени, за счет отказа от использования дельта-задержки. Представлены результаты работы алгоритма на примере моделирования на вентильном уровне ряда комбинационных и последовательностных схем.

Полученные временные диаграммы, а также время моделирования показывают, что предложенный алгоритм не уступает существующим средствам моделирования в плане достоверности и быстродействия.

Ключевые слова: цифровой симулятор; логическое моделирование; функциональное моделирование; алгоритм событийного моделирования; сети Петри.

Для цитирования: Булах Д.А., Казённов Г.Г., Лапин А.В. Использование модификации алгоритма работы сетей Петри для функционального моделирования логических схем, представленных на вентильном уровне// Изв. вузов. Электроника. – 2017. – Т. 22. – № 4. – С. 379–385. DOI: 10.214151/1561-5405-2017-22-4-379-385

Use of Modification of Petri Nets Operation Algorithm for Functional Simulation of Gate Level Digital Circuits

D.A. Bulakh, G.G. Kazennov, A.V. Lapin

National Research University of Electronic Technology, Moscow, Russia

dima@pkims.ru

The main algorithm for simulation of digital circuits, applied in digital simulators, is the algorithm of the event driven simulation. However, the implicit choice of the sequence functioning of simultaneously switching signals may lead to an appearance of differences in the obtained simulation results.

A new algorithm of the event driven functional simulation of digital integrated circuits, based on application of the modified mathematical apparatus of the Petri nets, has been proposed. In the apparatus of the Petri nets the parallel constructions are imitated using the sequential instructions. Hence, there is no need to separate the events using delta-delays. It has been shown that the described approach enables to eliminate the ambiguity of switching signals, which occur at the same time due to failure because of using the delta-delay. The results of the algorithm work have been presented on an example of simulation on the gate level of combinational and sequential schemes have been presented.

The obtained time diagrams as well as the simulation time show that the algorithm proposed does not yield to existing simulation tools in terms of reliability and simulation time.

Keywords: digital simulator; logical simulation; functional simulation; event driven simulation algorithm; delta-delay; Petri nets.

For citation: Bulakh D.A., Kazennov G.G., Lapin A.V. Use of Modification of Petri Nets Operation Algorithm for Functional Simulation of Gate Level Digital Circuits// Proc. of Universities. Electronics. – 2017. – Vol. 22. – No. 4. – P. 379–385. DOI: 10.214151/1561-5405-2017-22-4-379-385

Введение. Алгоритмы цифровых симуляторов с точки зрения функционирования работают в два этапа. На первом этапе определяется момент модельного времени, в который необходимо пересчитать состояние либо всей схемы, либо некоторого набора элементов для определения новых значений логического уровня сигнала в узлах схемы. На втором этапе проводятся непосредственно моделирование поведения логических вентилей и выяснение новых значений логических уровней на выводах сработавших вентилей.

Выделяют две основные группы алгоритмов прогнозирования модельного времени, за которое необходимо пересчитать состояние узлов цифровой схемы, – алгоритмы сквозного и событийного моделирования. Основным алгоритмом моделирования цифровых схем, применяемым на сегодняшний день в цифровых симуляторах, является алгоритм событийного моделирования [1]. При его реализации в отличие от алгоритма сквозного моделирования не выбирается фиксированный шаг моделирования, а при каждом переключении логического уровня выходов каких-либо вентилей схемы ось времени динамически дополняется событиями, а именно моментами времени и перечнем логических вентилей, состояние которых необходимо пересчитать в эти моменты времени. Несмотря на очевидное преимущество алгоритма событийного моделирова-

ния, имеет место такой недостаток, как неоднозначность выбора порядка выполнения параллельных конструкций в случае возникновения ситуации, при которой в один и тот же момент времени необходимо одновременно рассчитать состояние нескольких вентилей. Для разрешения этой неопределенности в событийные алгоритмы моделирования вводится понятие дельта-задержки, с помощью которой имитируется одновременность протекающих событий [2]. Использование такого механизма позволяет выполнить имитацию параллельности исполнения команд. Однако неявный выбор очередности исполнения последовательных команд не позволяет эффективно реализовать параллельное моделирование. Пример схемы, иллюстрирующей проблему выбора очередности, приведен на рис.1.

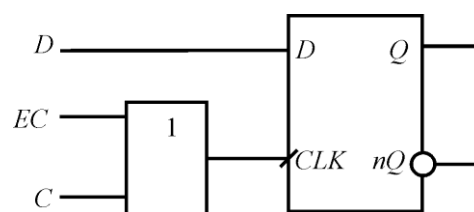
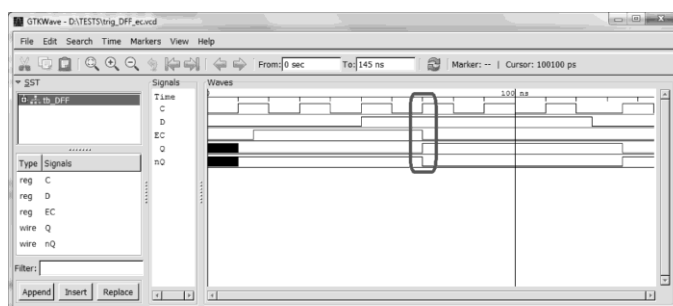
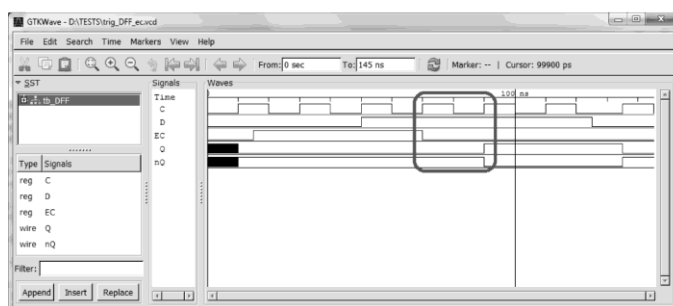


Рис.1. Тестовая схема, иллюстрирующая проблему выбора очередности моделирования
Fig.1. Test scheme illustrating ambiguity of signal switching

Очевидно, что пока на вход *EC* схемы подается высокий уровень, тактовый сигнал *C* не влияет на работу триггера. Допустим, что при моделировании схемы на входе *D* держится высокое значение сигнала, а вход *EC* переключается из низкого значения в высокое и одновременно с этим на вход *C* подается высокий уровень сигнала. В данном случае результат работы устройства будет определяться выбранной симулятором очередностью переключения сигнала. Если алгоритм выберет очередность *C–EC*, первым переключится сигнал *C*. При этом алгоритм будет считать, что на вход *EC* все еще подается высокий уровень сигнала, следовательно переключения выхода не произойдет. Если алгоритм выберет очередность *EC–C*, первым переключится сигнал *EC* и только затем – сигнал *C*. Очевидно, что при этом алгоритм будет считать, что на момент переключения входа *C* на вход *EC* подается уже низкий уровень сигнала, следовательно тактовый сигнал пройдет на триггер и триггер переключится.



a



б

Рис.2. Временные диаграммы неоднозначности выбора очередности исполнения параллельных конструкций: а – *C–EC*; б – *EC–C*

Fig.2. Time diagrams illustrating ambiguity of signal switching while executing parallel instructions. Sequence *C–EC* was chosen (a), sequence *EC–C* was chosen (b)

На рис.2 показано, как выбор очередности переключающихся входов проявляется при моделировании схемы. На рис.2,а изображена временная диаграмма для кода, в котором конструкции расставлены следующим образом: алгоритм считает, что в момент времени 70 нс сначала переключается сигнал C , а затем EC . На рис.2,б приведен результат моделирования в случае выбора симулятором последовательности $EC-C$.

В настоящей работе предлагается алгоритм моделирования цифровых устройств, основанный на модифицированном алгоритме сетей Петри, в котором отсутствует неоднозначность выбора очередности событий.

Сети Петри. Классический вариант сетей Петри [3] предполагает наличие понятий состояний (позиций) $S = \{s_1, s_2, \dots, s_N\}$ и переходов $T = \{t_1, t_2, \dots, t_M\}$, отношений между вершинами $F = \{f_1, f_2, \dots, f_N\}$, соответствующих дугам ориентированного графа сети, а также меток $M = \{m_1, m_2, \dots, m_N\}$. Пример простой сети Петри представлен на рис.3.

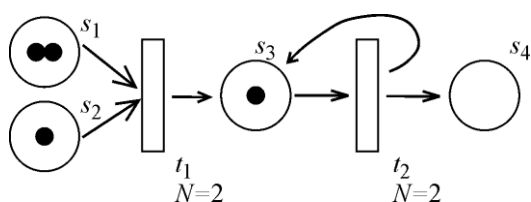


Рис.3. Пример сети Петри первого рода
Fig.3. An example of simple Petri net

В каждом из состояний сети s_i находится некоторое число меток m_i , являющихся характеристикой состояния. Например, на рис.3 для состояния s_1 число меток $m_1 = 2$, для состояний s_2 и s_3 соответственно $m_2 = 1$ и $m_3 = 1$, для состояния s_4 число меток $m_4 = 0$. Функционирование сети Петри заключается в переносе меток между состояниями посредством

срабатывания переходов. Перенос меток осуществляется в три этапа (рис.4).

Этап 1. Момент времени t_- – перенос меток от состояний ко входам каждого из переходов (рис.4,а).

Этап 2. Момент времени t_0 – срабатывание перехода, переходы забирают метки со входов, выдают метки на выходе (рис.4,б).

Этап 3. Момент времени t_+ – перенос меток от выходов каждого из переходов к соответствующим состояниям (рис.4,в).

Каждое из этих действий выполняется одновременно для всех переходов в графе сети Петри: сначала для всех переходов сети выполняется перенос меток от состояний на входы переходов, затем для всех переходов выполняется перенос меток через переходы и, наконец, для всех переходов выполняется перенос меток от переходов к выходным состояниям. Применение подобного многоступенчатого аппарата позволяет естественным образом описать параллельность протекания процессов, а в терминах сетей Петри – распространение меток от состояний к состояниям через переходы, не выбирая очередность срабатывания переходов.

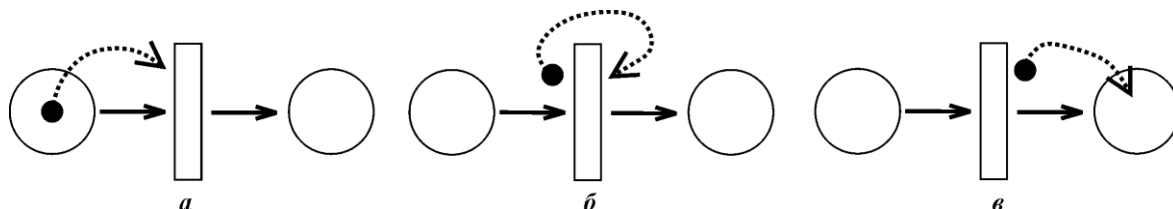


Рис.4. Этапы процесса переноса меток в сетях Петри: а – момент времени t_- ;
б – момент времени t_0 ; в – момент времени t_+

Fig.4. Stages of transmitting a dot in Petri net: а – stage t_- ; б – stage t_0 ; в – stage t_+

Модификация сетей Петри для логического моделирования. Сети Петри при моделировании цифровых схем использовались и раньше. Однако этот аппарат применялся либо для описания поведения схем как один из способов описания автомата [4, 5], либо для описания элементов в терминах обычных сетей Петри [6, 7]. В первом случае моделирование происходит обычными симуляторами с традиционными алгоритмами реализации, во втором – появляется существенная избыточность по числу элементов сетей Петри для описания даже простейших логических вентилей.

Предлагаемый алгоритм моделирования основан на модификации алгоритма функционирования сетей Петри. Суть модификации для моделирования схем, задаваемых на вентильном уровне, заключается в следующем.

1. Состояние представляет собой узел цифровой схемы, следовательно в каждом состоянии не может находиться более одной метки. Наличие метки в состоянии эквивалентно значению уровня логической единицы в узле, отсутствие метки – уровню логического нуля. В случае реализации многозначной логики предлагается воспользоваться формализмом цветных сетей Петри – вариантом сетей Петри, в котором каждая метка имеет некоторую внутреннюю характеристику, обычно обозначаемую цветом. В терминах многозначного логического моделирования цвет показывает состояние логического уровня. Например, в современных симуляторах при моделировании цифровых устройств применяются логический «0», логическая единица «1», неопределенное состояние «U», неизвестное состояние «X», высокоимпедансное состояние «Z» и другие значения логических уровней.

2. Переход представляет собой логический вентиль. В простейшем случае для симулятора комбинационных схем переходы осуществляют функционирование в соответствии с заранее предопределенным набором встроенных логических примитивов: «НЕ», «ИЛИ», «ИЛИ-НЕ», «И», «И-НЕ», «ИСКЛЮЧАЮЩЕЕ ИЛИ».

3. Для срабатывания перехода необязательно наличие метки во входных состояниях, все зависит только от типа перехода. Так, для появления метки в выходном состоянии перехода, соответствующего логическому вентилю «НЕ», необходимо, чтобы на входном состоянии метки не было (см. рис.4,а,б), что соответствует таблице истинности инвертора.

4. Факт наличия метки в состоянии на выходе перехода зависит не только от срабатывания перехода, но и от его типа. Наличие меток в состоянии на выходе перехода обуславливается типом логического вентиля и таблицей истинности, описывающей его работу (аналогично п.3).

Реализация симулятора. Для проверки работоспособности предложенного алгоритма моделирования цифровых комбинационных схем разработан логический симулятор на языке программирования C++, в котором реализован описанный модифицированный аппарат сетей Петри. Аппарат позволяет провести событийное моделирование цифровой схемы, описанной на вентильном уровне с использованием встроенных логических примитивов. В качестве входных данных в данной версии используется упрощенное подмножество языка Verilog. На рис.5 показаны временные диаграммы корректной работы схемы триггера (см. рис.1), полученные в результате моделирования разработанным симулятором.

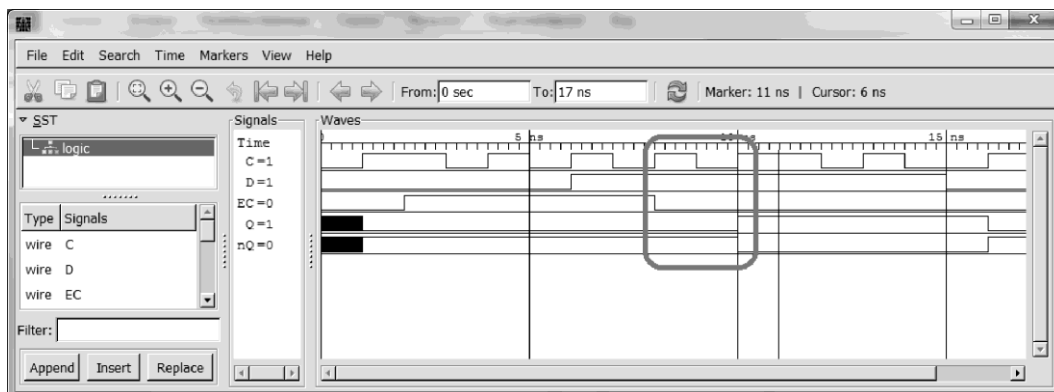


Рис.5. Результат моделирования тестовой схемы с помощью симулятора, основанного на использовании модифицированного алгоритма сетей Петри
Fig.5. Results of test scheme simulation using the modified algorithm of Petri nets simulation

Работоспособность алгоритма проверена на следующих комбинационных и последовательностных схемах: мультиплексоры, демультиплексоры различной разрядности, сумматоры; сдвиговые регистры различной разрядности; схемы из набора iscas85. Результаты моделирования схем с применением разработанного алгоритма соответствуют действительности.

Заключение. Использование предложенной модификации алгоритма работы сетей Петри позволило разработать и программно реализовать работу симулятора моделирования цифровых схем, описанных на вентильном уровне. Алгоритм, несмотря на более требовательную к ресурсам реализацию (три действия на каждый шаг моделирования), выполняет расчет всех вентилей одновременно. В результате устраняется недостаток событийного алгоритма моделирования. Использование сетей Петри поддерживает возможность создания параллельной реализации программного кода, что подтверждается выбранным математическим аппаратом.

В дальнейшем для разработанного симулятора планируется реализовать поддержку возможности моделирования схем, описываемых на поведенческом уровне с возможностью распараллеливания, а также добавить поддержку входного формата VHDL.

Литература

1. Gunes M., Thornton M.A., Kocan F., Szygenda S.A. A survey and comparison of digital logic simulators // 48th Midwest Symposium on Circuits and Systems. – 2005. – Vol. 1. – P. 744–749.
2. Суворова Е.А., Шейнин Ю.Е. Проектирование цифровых систем на VHDL. – СПб.: БХВ-Петербург, 2003. – 576 с.
3. Diaz M. Petri nets: fundamental models, verification and applications. – Wiley, 2009. – 656 p.
4. Yakovlev A., Gomes L., Lavagno L. Hardware design and petri nets. – Kluwer Academic Publishers, 2000. – 335 p.
5. Beerel P.A., Ozdag R.O., Ferretti M. A designer's guide to asynchronous VLSI. – Cambridge University Press, 2010. – 352 p.
6. Idzikowska E. Petri net models of VHDL control statements // The International Workshop on Discrete-Event System Design, 2001.
7. Веселов А.А. Распределенная модель устройств цифровой и вычислительной техники на основе сетей Петри // Интернет-журнал «Науковедение». – 2015. – Т.7. – №3.
URL: <http://naukovedenie.ru/PDF/124TVN315.pdf> (дата обращения: 7.04.2017)

Поступила после доработки 12.04.2017 г.; принята к публикации 25.04.2017 г.

Булах Дмитрий Александрович – кандидат технических наук, доцент кафедры проектирования и конструирования интегральных микросхем Национального исследовательского университета «МИЭТ» (Россия, 124498, г. Москва, г. Зеленоград, пл. Шокина, д. 1), *dima@pkims.ru*

Казённов Геннадий Георгиевич – доктор технических наук, профессор-консультант кафедры проектирования и конструирования интегральных микросхем Национального исследовательского университета «МИЭТ» (Россия, 124498, г. Москва, г. Зеленоград, пл. Шокина, д. 1), *gkazn@mail.ru*

Лалин Александр Владимирович – аспирант кафедры проектирования и конструирования интегральных микросхем Национального исследовательского университета «МИЭТ» (Россия, 124498, г. Москва, г. Зеленоград, пл. Шокина, д. 1), *xanderius@mail.ru*

References

1. Gunes, M., Thornton, M.A., Kocan, F., Szygenda, S.A. A survey and comparison of digital logic simulators. *48th Midwest Symposium on Circuits and Systems*, 2005, vol. 1, pp. 744–749.
2. Suvorova E.A., Shejnin Yu.E. *Proektirovanie tsifrovyykh sistem na VHDL* [Digital systems design using VHDL]. St. Petersburg, BKHV – Peterburg, 2003. 576 p. (In Russian).
3. Diaz M. *Petri nets: fundamental models, verification and applications*. Wiley, 2009. 656 p.
4. Yakovlev A., Gomes L., Lavagno L. *Hardware Design and Petri Nets*. Kluwer Academic Publishers, 2000. 335 p.
5. Beerel P.A., Ozdag R.O., Ferretti M. *A Designer's Guide to Asynchronous VLSI*. Cambridge University Press, 2010. 352 p.
6. Idzikowska E. *Petri net models of VHDL control statements*. The International Workshop on Discrete-Event System Design, 2001.
7. Veselov A.A. Raspredeleonnaya model' ustroystv tsifrovoy i vychislitel'noy tekhniki na osnove setej Petri [Distributed model of the digital and computing devices based on Petri nets]. *Online Magazine «Science» – Internet-zhurnal «Naukovedenie»*, 2015, vol.7, no.3. Available at: <http://naukovedenie.ru/PDF/124TVN315.pdf> (accessed: 07.04.2017). (In Russian).

Submitted 12.04.2017; accepted 25.04.2017.

Bulakh Dmitry A. – PhD, associate professor of the Integrated Circuits Design Department, National Research University of Electronic Technology (Russia, 124498, Moscow, Zelenograd, Shokin sq., 1), *dima@pkims.ru*

Kazennov Gennadiy G. – doctor of technical sciences, full professor of the Integrated Circuits Design Department, National Research University of Electronic Technology (Russia, 124482, Moscow, Zelenograd, Shokin sq., 1), *gkazn@mail.ru*

Lapin Alexander V. – PhD student of the Integrated Circuits Design Department, National Research University of Electronic Technology (Russia, 124482, Moscow, Zelenograd, Shokin sq., 1), *xanderius@mail.ru*