

10. Вершинин, Ю. Н. Электронно-тепловые и детонационные процессы при электрическом пробое твердых диэлектриков [Текст] / Ю. Н. Вершинин. — Екатеринбург, 2000. — 258 с.
11. Куликов, В. Д. Исследование механизма электрического пробоя ионных кристаллов в наносекундном диапазоне [Текст] / В. Д. Куликов // Журнал технической физики. — 2003. — Том 73, Вып. 12. — С. 26–30.

МЕТОД ФОРМУВАННЯ ПОТУЖНИХ МІКРОВОЛНОВИХ ІМПУЛЬСІВ НАНОСЕКУНДНОЇ ТРИВАЛОСТІ

Проведено аналіз залежності параметрів потужних мікрохвильових імпульсів наносекундної тривалості від особливостей конструкції формувача генератора. Оптимізовані параметри конструкції формувача за критерієм досягнення максимальної потужності генератора. В результаті досліджень показано, що за рахунок оптимізації параметрів конструкції

формувача, коефіцієнт посилення генератора збільшується за потужністю в 11 разів.

Ключові слова: генератор імпульсів ударного збудження, мікрохвильові імпульси наносекундної тривалості.

Шостко Ігорь Светославович, доктор технических наук, доцент, кафедра телекоммуникационных систем, Харьковский национальный университет радиоэлектроники, Украина, e-mail: igor-shostko@yandex.ru.

Шостко Ігор Світославович, доктор технічних наук, доцент, кафедра телекомунікаційних систем, Харківський національний університет радіоелектроніки, Україна.

Shostko Igor, Kharkiv National University of Radio Electronics, Ukraine, e-mail: igor-shostko@yandex.ru

УДК 004.925.83

DOI: 10.15587/2312-8372.2015.37693

**Аноприенко А. Я.,
Федоров Е. Е.,
Иваница С. В.,
Альрабаба Хамза**

ОСОБЕННОСТИ АППАРАТНОЙ РЕАЛИЗАЦИИ ОБОБЩЕННОГО КЛЕТОЧНОГО ТЕТРААВТОМАТА

Предложен вариант обобщенной структуры ячейки клеточного автомата с использованием асинхронного запоминающего устройства, программно настраиваемого через пользовательский интерфейс. Рассмотрены несколько способов аппаратной реализации записи начальных значений в регистр состояний ячейки и считывания результатов со слоя текущих состояний в регистр считывания результатов: последовательный, блочный, с использованием слоя начальных состояний через непосредственную последовательную или блочную адресацию, или каскадную адресацию на основе тетракодов.

Ключевые слова: постбинарный клеточный автомат, тетраавтомат, расширенный кодологический базис, запоминающее устройство, ячейка, тетракод.

1. Введение

Аппаратная реализация алгоритмов и вычислительных процессов является способом достижения максимальной производительности. В общем случае при аппаратной реализации по сравнению с программной достигается ускорение на 1–2 порядка. Однако клеточные автоматы являются довольно специфическим видом вычислительных устройств, в связи с чем известно относительно мало успешных попыток их аппаратной реализации.

В работах [1–3] определены постбинарные клеточные автоматы (ПКА), в который, в отличие от традиционных клеточных автоматов (КА), число состояний ячеек больше двух. В цикле работ, например [4–6] в качестве постбинарной логики и постбинарного кодирования выделены такие направления, как тетралогика и тетракодирование.

В данной работе рассматривается аппаратная реализация обобщенного клеточного тетраавтомата (ОКТА), как частного случая ПКА, который, во-первых, по структуре является машиной клеточных автоматов (МКА), и, во-вторых, расширяет управляемость МКА с возможностью подключения к сети Интернет, поэтому до-

ступ к тетраавтомату может осуществляться с любого удаленного компьютера.

2. Анализ литературных данных и постановка проблемы

Теория КА обязана своим появлением Джону фон Нейману. В конце 40-х годов, накопив огромный практический опыт в создании быстродействующих вычислительных машин, фон Нейман приступил к созданию общей математической теории автоматов [7].

После фон Неймана исследования были продолжены его учеником Станиславом Уламом и Джоном Холландом — сотрудниками университета Лос-Аламос в университете Мичигана [8–10].

Практически параллельно и независимо от исследования Улама и фон Неймана работу над КА начал Конрад Цузе, который в 1969 г. опубликовал книгу «Rechnender Raum» («Вычислительное пространство») — «Calculating Space» [11].

В 1984 г. Кристофер Лэнгтон создал самовоспроизводящийся автомат, клетка которого может находиться в одном из восьми возможных состояний, но при этом для самовоспроизведения требуется гораздо меньшее

количество клеток, чем сотни и тысячи ячеек, необходимых для самовоспроизведения в автомате Неймана [12].

После Лэнгтона теория самовоспроизводящихся автоматов стала двигаться в сторону упрощения. Среди последователей Лэнгтона широко известны работы: Эдгара Кодда (1968), Эдвина Роджера Бэнкса, Н. Н. Чоу, J. Reggia (1993), Вул (1999) и других исследователей, результатами работы которых стало появление множества КА с количеством состояний от 8 до 64 [13–16].

Самым известным последователем, продолжившим исследования в области упрощения КА, стал английский математик Джон Хортон Конуэй — создатель математической игры «Жизнь» [1, 17–19].

3. Объект, цель и задачи исследования

Объект исследования — обобщенный клеточный тетраавтомат, в котором для кодирования состояния каждой ячейки используется расширенное кодо-логическое пространство [4].

Цель исследования — формирование схемы локального подключения машины КА, схемы работы через браузер клиента с модульной наращиваемой системой ОТКА, схемы работы через специальное клиентское приложение с модульной наращиваемой системой ОКТА, а также разработка структурных схем: начальной записи в ячейку КА через непосредственную адресацию, записи начальных значений путем каскадной адресации, обобщенной структурной схемы многослойной ячейки КА.

Для решения поставленной цели необходимо выполнить такие задачи:

1. Рассмотреть структуры КТА при использовании в качестве сопроцессоров в локальных компьютерах.
2. Для обеспечения доступности широкого круга пользователей и загрузки соответствующих сопроцессоров предложить вариант достаточно универсальной аппаратной реализации КТА как сетевого ресурса.
3. Для организации схемы подключения осуществить две возможности получения доступа к ОКТА: через специальное приложение, загруженное через веб-браузер; через специальное клиентское приложение, подключенное напрямую к универсальному компьютеру с доступом к ОКТА.
4. Разработать и подробно описать работы каждого узла обобщенной структурной схемы ячейки КТА с окрестностями фон Неймана и Мура.

4. Обобщенный клеточный тетраавтомат

В работе [7] рассматривается реализация машины клеточных автоматов САМ-6, предложенная специалистами Массачусетского технического института (США) в 80-е годы. Физически она представляла собой модуль, подключаемый к одному из разъемов персонального компьютера общего назначения. При этом головной компьютер обеспечивал питание, визуализацию результатов на экране, дисковую память, стандартную операционную систему, а все вычисления, связанные непосредственно с реализацией алгоритмов КА, производились на машине САМ. Авторы рассматривали возможность серийного производства САМ с целью сделать такие вычислительные устройства доступными для широкого круга специалистов. Однако эта цель так и осталась недостигнутой. В конечном итоге было реализовано всего несколько опытных образцов.

Рассматриваемые далее структуры, предлагаемые для аппаратной реализации, могут также использоваться как сопроцессоры в локальных компьютерах. Машина клеточных автоматов в этом случае функционирует как сопроцессор персонального компьютера, взаимодействующий с ним через соответствующий интерфейс (например, USB). В этом случае пользователь имеет возможность задавать начальные условия для МКА с помощью устройств ввода ПК и получать результаты, визуализируемые на мониторе (рис. 1).

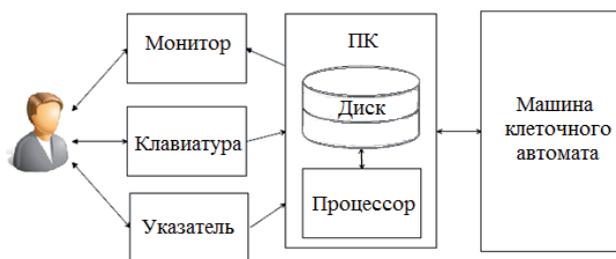


Рис. 1. Схема локального подключения машины клеточных автоматов

Такое использование аппаратно реализованных КА не является достаточно эффективным в современных условиях, так как обеспечить приемлемый уровень доступности для широкого круга пользователей и загрузки соответствующих сопроцессоров в этом случае крайне сложно.

В современных условиях может быть предложен вариант чрезвычайно эффективного решения этой проблемы путем реализации достаточно универсальной аппаратной реализации КТА как сетевого ресурса. В настоящее время уровень развития сетевой инфраструктуры изменился настолько, что предоставление пользователю доступа к специализированным вычислительным ресурсам в качестве сетевых сервисов позволяет обеспечить не только высокий уровень доступности таких ресурсов, но и, как следствие, достаточно высокий уровень их загрузки. Это резко повышает актуальность таких разработок и делает вполне оправданным их применение.

При этом целесообразно обеспечить максимальную производительность аппаратной реализации КА при достаточно высоком уровне ее универсальности. Последнее может быть достигнуто путем реализации некоторой обобщенной структуры, настраиваемой программным путем через соответствующие интерфейсы.

Для реализации машины клеточных автоматов как сетевого ресурса может быть предложена структура, представленная на рис. 2 и рис. 3.

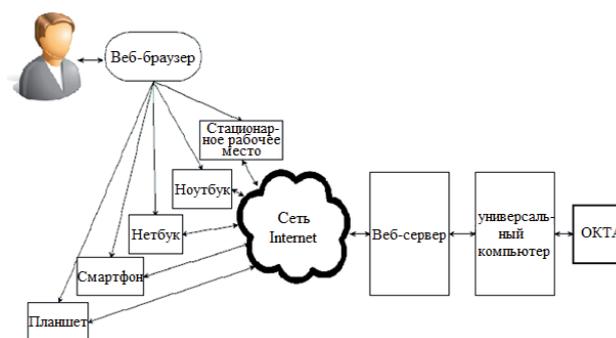


Рис. 2. Работа через браузер клиента с модульной наращиваемой системой ОКТА

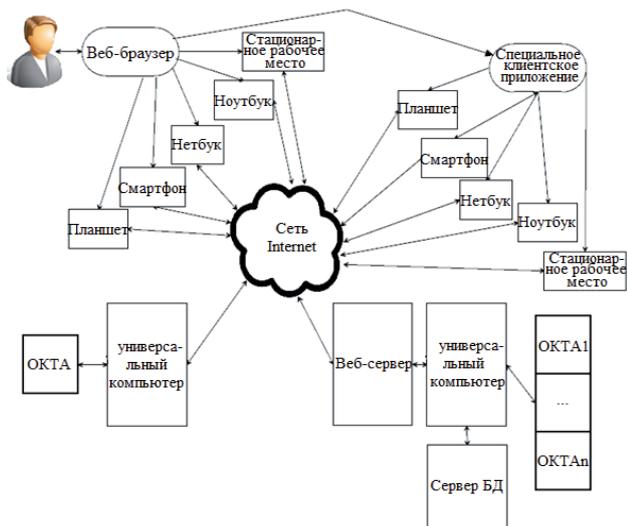


Рис. 3. Работа через специальное клиентское приложение с модульной наращиваемой системой ОКТА

Рассматриваемая схема устройства машины клеточных автоматов как сетевого рисунка, в данной работе определена как обобщенный постбинарный клеточный тетраавтомат.

При такой организации схемы подключения просматривается две возможности получения доступа к ОКТА:

1) через специальное приложение, загруженное через веб-браузер, тогда доступ будет осуществлен через веб-сервер к универсальному компьютеру, к которому подключен ОКТА;

2) через специальное клиентское приложение напрямую к универсальному компьютеру с доступом к ОКТА.

5. Обобщенная структурная схема ячейки клеточного автомата

Для аппаратной реализации ОКТА, например, на базе FPGA, в простейшем случае достаточным является традиционный подход к структурной организации ячейки. Традиционная структурная схема предполагает, что i -я ячейка КА представляет собой элементарный автомат, состоящий из комбинационной схемы, на вход которой подаются состояния соседей, а на выходе формируется новое состояние i -й клетки, и запоминающего устройства, предназначенного для хранения состояния ячейки до следующей итерации. На рис. 4, а приведена соответствующая структурная схема автомата с соседством фон Неймана, а на рис. 4, б — структурная схема КА с соседством Мура.

Структурная схема ячейки (i, j) для двумерного КА с типом соседства фон Неймана изображена на рис. 4, а, где ЗУ — элементы памяти и КС — комбинационная схема, реализующей функцию (1):

$$z_{i,j}(t+1) = f_{i,j}(z_{i,j-1}(t), z_{i,j+1}(t), z_{i-1,j}(t), z_{i+1,j}(t)), \quad (1)$$

где $z_{i,j-1}(t), z_{i,j+1}(t), z_{i-1,j}(t), z_{i+1,j}(t)$ — текущие состояния соседних КА $z_{i,j}$.

Комбинационная схема КА с типом соседства Мура (рис. 4, б) реализует функцию (2):

$$z_{i,j}(t+1) = f_{i,j}(z_{i,j-1}(t), z_{i,j+1}(t), z_{i-1,j-1}(t), z_{i-1,j}(t), z_{i-1,j+1}(t), z_{i+1,j-1}(t), z_{i+1,j}(t), z_{i+1,j+1}(t)), \quad (2)$$

где $z_{i,j-1}(t), z_{i,j+1}(t), z_{i-1,j-1}(t), z_{i-1,j}(t), z_{i-1,j+1}(t), z_{i+1,j-1}(t), z_{i+1,j}(t), z_{i+1,j+1}(t)$ — текущие состояния соседних КА $z_{i,j}$.

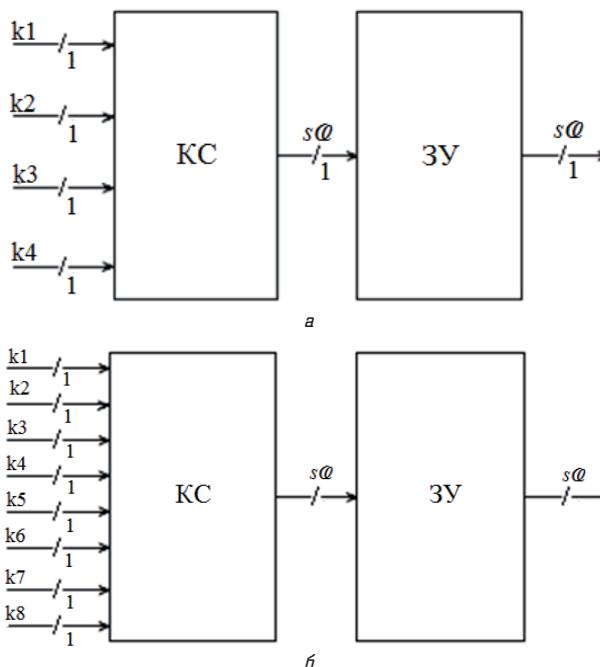


Рис. 4. Традиционная структурная схема ячейки КА по типу соседства фон Неймана (а) и типу соседства Мура (б)

Недостатком такой схемы является неуниверсальность, так как при жесткой реализации логики работы ячейки через комбинационную схему невозможно при необходимости модифицировать алгоритм работы КА.

В работе [7] с целью повышения гибкости и универсальности было предложено заменить комбинационную схему асинхронным запоминающим устройством с загружаемой таблицей правил перехода. Но каких-либо конкретных решений предложено не было.

В данной работе предлагается обобщенная структурная схема ячейки для КА (рис. 5), которая может быть использована в качестве программируемого сетевого ресурса. В результате одновременно решается вопрос аппаратной реализации КА и возможность доступа к такому устройству широкого круга пользователей.

На вход асинхронного комбинационного запоминающего устройства ЗУ подаются предыдущие состояния соседних ячеек (s_1, \dots, s_k) . По сигналу записи (зап) в ЗУ с шины данных (MD) может быть загружена таблица правил КА с явно указанными новыми состояниями ячейки в зависимости от предыдущих состояний соседних ячеек.

При подаче соответствующих значений состояний соседей на адресные входы ЗУ из таблицы правил по адресу строки s_1, \dots, s_k считывается новое состояние КА и записывается в синхронизированный регистр состояний (RG состояний).

На базе данной структурной схемы могут быть реализованы как традиционные КА, так и модифицированные КА.

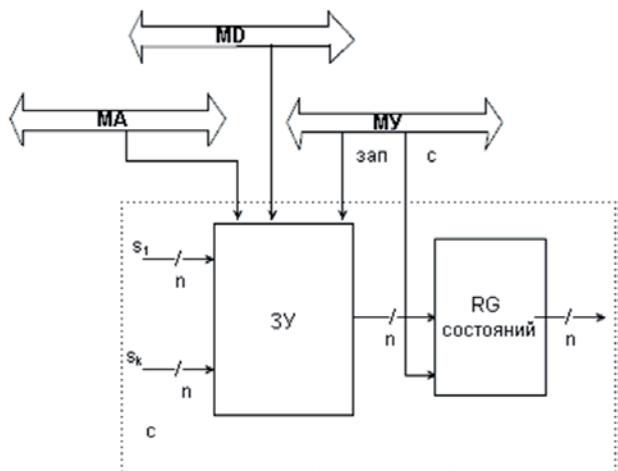


Рис. 5. Структурная схема ячейки клеточного автомата

6. Программно-аппаратные способы организации слоя начальных состояний ячеек клеточного автомата

Важной задачей при построении обобщенной схемы ячейки КА является такая организация процесса записи начальных значений, которая позволит в общем случае существенно ускорить этот процесс.

Этот процесс может быть организован последовательным и блочным способом. Наиболее экономичным вариантом с точки зрения аппаратных затрат является последовательный вариант.

На рис. 6 приведена структурная схема взаимодействия элемента (i, j) двумерного ПКА с окрестностью фон Неймана: возможность осуществления сдвига начального значения с последовательным заполнением всех регистров ПКА начальными значениями.

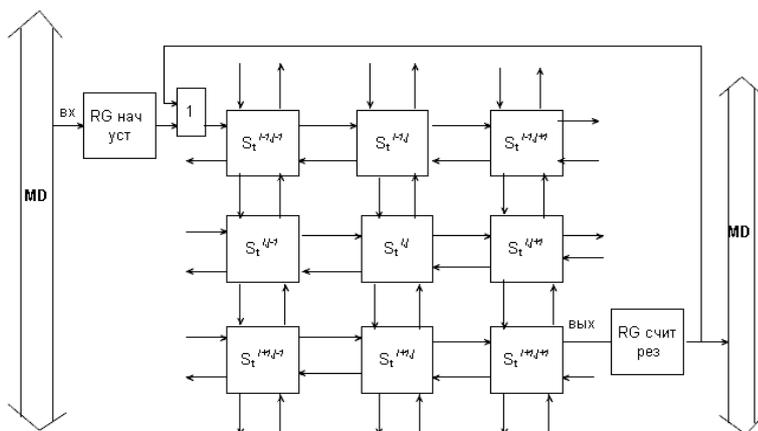


Рис. 6. Структурная схема КТА с окрестностью фон Неймана

На рис. 7 представлена аналогичная схема для типа соседства Мура.

В этом случае не требуется введения дополнительных связей между ячейками. Но необходимо обе-

спечить возможность осуществления операции сдвига в каждой ячейке.

Время последовательной поразрядной записи в устройство напрямую зависит от количества ячеек автомата. При этом на вход подается i -й разряд с шины данных и записывается в одну ячейку КА через регистр начальной записи, на следующем такте это значение сдвигается в следующую ячейку КА и на вход регистра начальной записи подается новый разряд с шины данных.

Недостатком такой организации КА является слишком медленный процесс записи и считывания поля КА.

Частично преодолеть этот недостаток позволяет блочная запись, которая может быть организована различными способами, при этом размер одновременно записываемого блока данных может составлять от одного до 8-ми байт (в соответствии с разрядностью магистрали данных).

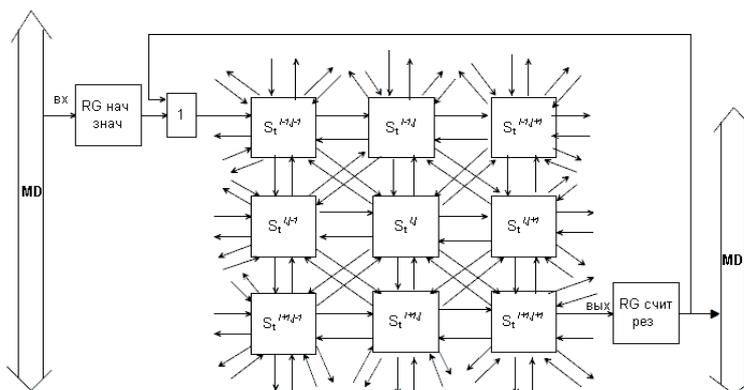


Рис. 7. Структурная схема КТА с окрестностью Мура, обеспечивающего последовательную загрузку начального состояния КА и считывания его результатов

Аналогичным образом в КА может быть организовано считывание результатов на каждом шаге для последующего хранения или визуализации этих значений. При этом на рис. 6, 7 от регистра считывания к первой ячейке КА организована обратная связь. Так, при считывании результатов значение с выхода регистра считывания подается на вход первого элемента. За один шаг осуществляется запись результатов на шину данных, запись из регистра считывания результатов в первую ячейку КА.

При таком способе организации записи начальных значений и считывания результатов вся работа КА может быть организована в одном слое.

Предлагается два программно-аппаратных способа организации слоя начальных состояний:

- 1) запись в ячейку через непосредственную адресацию (рис. 8);
- 2) запись в ячейку через каскадную адресацию (используя тетрады).

На рис. 8 приведена структурная схема слоя начальных состояний с использованием непосредственной адресации. Слой начальных состояний КА представляет собой блок регистров, размерностью $n \times n$, при этом размерность слоя начальных состояний должна быть равна слою текущих состояний. Через

адресную шину (МА) по сигналу записи (зап) в дешифратор (DC) передается адрес элемента, в который будут записаны данные с шины данных (MD).

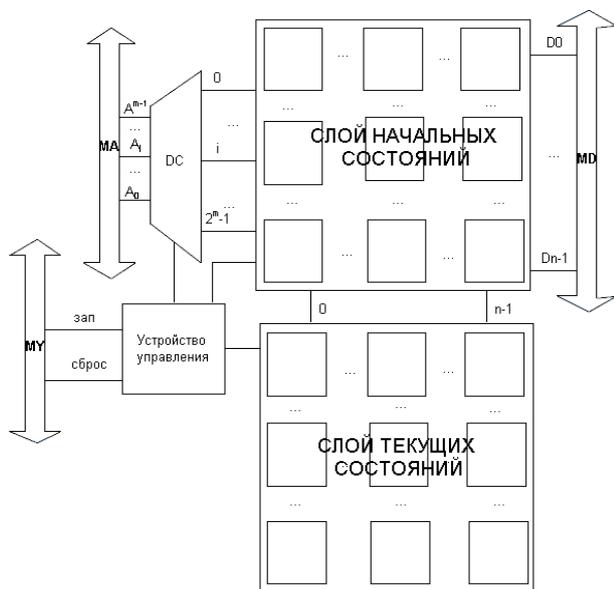


Рис. 8. Структурная схема начальной записи в ячейку КА через непосредственную адресацию

Все регистры слоя начальных состояний могут быть сброшены в 0 по сигналу «сброс». Сигналы управления передаются через соответствующую шину управления. Дешифратор DC состоит из 2-х частей, дешифратора, предназначенного для адресации столбцов (или координаты X), DCX, и дешифратора для адресации строк (или координаты Y), DCY.

Каскадная адресация и запись начальных значений является наиболее эффективным по быстродействию способом задания начальных значений КА. На рис. 9 представлена структурная схема записи начальных значений путем каскадной адресации (приведен пример для размерности поля КА 16 × 16).

Принцип работы схемы, приведенной на рис. 9, следующий. На вход регистра данных (RG) с шины данных (MD) по сигналу записи (зап) подается значение, которое в общем случае может представлять собой одно из значений тетракода {1, M}. По сигналу «сброс» значение регистра сбрасывается в 0. Возможность задания значения A тетракода на данной структурной схеме не предусмотрено. С адресной шины на входы дешифраторов: DCX0, DCY0, DCX1, DCY1, DCX2, DCY2, подаются адресные разряды, которые представляют собой одно из значений тетракода {0, 1, M}. Комбинационные схемы управления (КС упр1, КС упр2, КС упр3) формируют сигналы записи, которые подаются на соответствующие входы «зап» всех регистров каждого уровня. При этом каждая КС отвечает за формирование сигнала записи на своем уровне. Формирование сигнала зависит от выходов и от соответствующих выходов регистров предыдущего уровня, также участвующих в формировании адреса соответствующих регистров.

Верхний уровень каскада представляет собой 4 регистра, на входы данных которых подаются значения из регистра данных (RG). На вход R — сигнал «сброс». На вход записи (зап) — подается сигнал с КС. Выходы каждого регистра

подаются на КС следующего уровня для формирования сигнала записи регистров для этого уровня. Размерность блоков регистров каждого последующего уровня больше в 4 раза блока регистров предыдущего уровня.

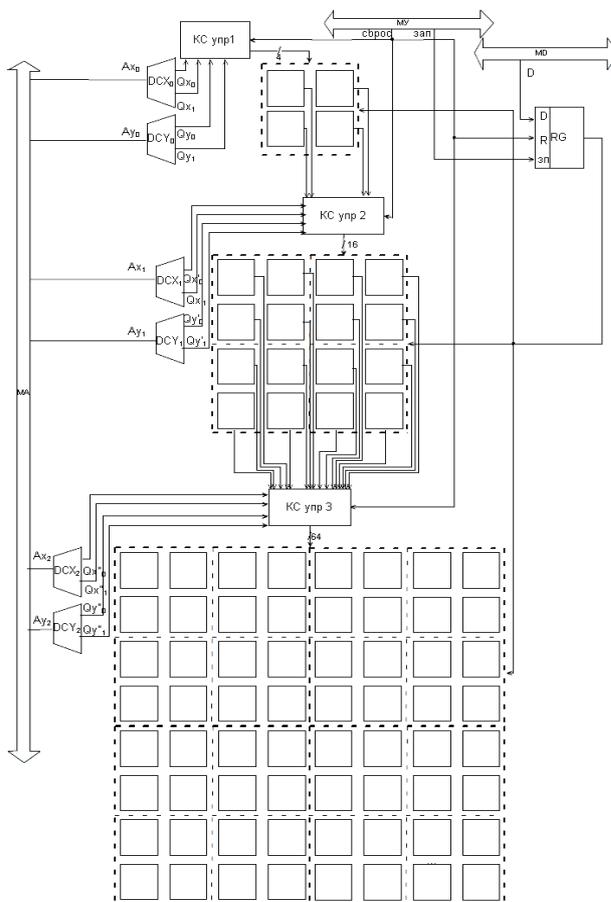


Рис. 9. Запись начальных значений путем каскадной адресации для размерности поля 16 × 16

7. Обобщенная структурная схема ячейки КА

На рис. 10 приведена обобщенная структурная схема многослойной ячейки КА. Основными компонентами данной схемы КА являются: ЗУ — асинхронное комбинационное запоминающее устройство; RG сост — регистр состояний; RG маски — регистр маски; RG нач уст — регистр начальной установки входных значений.

Шина данных (MD) подключена к компьютеру, через специальный программный интерфейс пользователь может управлять работой КА. ЗУ представляет собой таблицу правил, в которой в зависимости от комбинации состояний соседних клеток явным образом определено, в какое состояние переключается текущая клетка.

На вход ЗУ с шины управления (МУ) подается управляющий сигнал «зап» (запись), по которому в КЗУ с шины данных может осуществляться запись правил КА в виде строк таблицы переходов КА. Пользователь сам программно задает таблицу переходов КА, которая загружается в ЗУ.

В RG маски по сигналу записи записываются через шину данных значения маски КА. Содержание и принцип маскирования также программируются через программный интерфейс пользователем и загружаются в регистр маски.

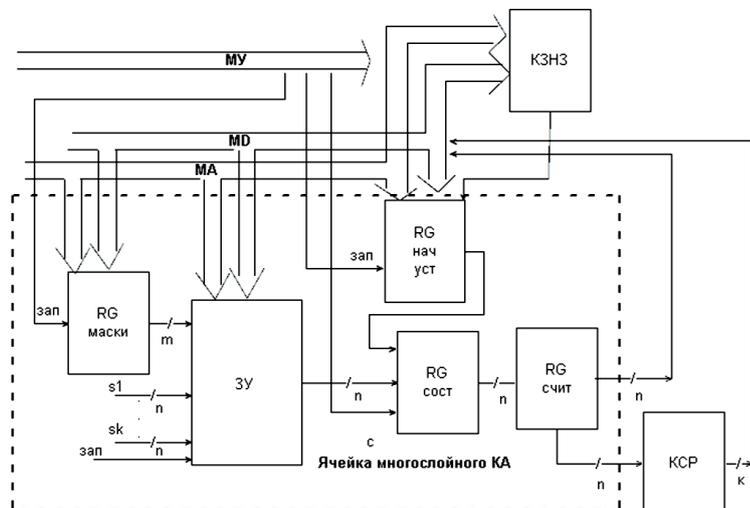


Рис. 10. Обобщенная структурная схема многослойной ячейки КА

В зависимости от состояний соседей (s_1, \dots, s_k) и содержимого регистра маски на каждом шаге работы устройства из таблицы ЗУ выбирается соответствующее значение нового состояния i -й ячейки КА и записывается в регистр состояний.

Основная функция регистра состояний заключается в хранении состояний КА до записи новых значений. Запись новых значений может осуществляться либо с регистра начальных значений, либо на каждом такте работы устройства с АЗУ. По сигналу записи в регистр начальных значений с шины данных загружаются начальные состояния КА.

8. Выводы

На сегодняшний день КА широко применяются во многих отраслях знания для исследования реальных различных сложных процессов. Поэтому рассмотрение аппаратно-программного решения КА (равно как и для КТА) набирает все большую актуальность. В данной работе предложен вариант обобщенной структуры ячейки КА с использованием асинхронного запоминающего устройства, настраиваемого программно через пользовательский интерфейс. Также рассмотрены несколько способов аппаратной реализации записи начальных значений в регистр состояний ячейки и считывания результатов со слоя текущих состояний в регистр считывания результатов: последовательный, блочный, с использованием слоя начальных состояний через непосредственную последовательную или блочную адресацию, или каскадную адресацию на основе тетракодов.

В результате проведенных исследований

1. Рассмотрена структура КТА при использовании в качестве сопроцессоров в локальных компьютерах. При этом даны подробные описания, выявлена проблематика, приведены соответствующие схематические структуры.

2. Для ускорения быстродействия аппаратной реализации КА предложено использовать многослойную структурную организацию КА, с использованием слоя начальных состояний, слоя текущих состояний, слоя считывания результатов.

3. Для организации схемы подключения рассмотрены возможности получения доступа к ОКТА как

через специальное приложение, загруженное через веб-браузер, так и через клиентское приложение, обращающееся напрямую к универсальному компьютеру с доступом к ОКТА.

4. Подробно рассмотрена обобщенная структурная схема многослойного клеточного автомата и предложена обобщенная структурная схема многослойной ячейки КА, отличающаяся повышенной производительностью и расширенной функциональностью по сравнению с известными КА.

Литература

1. Коноплева, А. П. Игра «Жизнь» Дж. Конвея на базе гиперкодов [Текст] / А. П. Коноплева, А. Я. Аноприенко // Материалы III международной научно-технической конференции «Информатика и компьютерные технологии-2007», 11–13 декабря 2007 года. — Донецк, ДонНТУ, 2007. — С. 254–257.
2. Аноприенко, А. Я. Развитие идеи применения гиперкодов в моделировании клеточных автоматов [Текст] / А. Я. Аноприенко, А. П. Коноплева // Наукові праці Донецького національного технічного університету. Серія «Інформатика, кібернетика та обчислювальна техніка». — Донецк, ДонНТУ, 2008. — Вып. 9(132). — С. 115–118.
3. Аноприенко, А. Я. Управляемый постбинарный клеточный автомат / А. Я. Аноприенко, А. П. Коноплева [Текст] // Материалы II всеукраїнської науково-технічної конференції «Інформаційні управляючі системи і комп'ютерний моніторинг (ІУС і КМ 2011)», 12–13 апреля 2011 г. — Донецк: ДонНТУ, 2011. — Т. 2. — С. 215–219.
4. Аноприенко, А. Я. Постбинарный компьютеринг и интервальные вычисления в контексте кодо-логической эволюции [Текст] / А. Я. Аноприенко, С. В. Иваница. — Донецк: ДонНТУ, УНИТЕХ, 2011. — 248 с.
5. Аноприенко, А. Я. Особенности реализации постбинарных логических операций [Текст] / А. Я. Аноприенко, С. В. Иваница // Научно-теоретический журнал «Искусственный интеллект». — 2011. — № 2. — С. 110–121.
6. Аноприенко, А. Я. Тетралогики, тетравычисления и ноокомпьютинг [Текст] / А. Я. Аноприенко, С. В. Иваница. — Донецк: ДонНТУ, УНИТЕХ, 2012. — 308 с.
7. Piwonska, A. Searching Cellular Automata Rules for Solving Two-Dimensional Binary Classification Problem [Text] / A. Piwonska, F. Seredynski, M. Szaban // Lecture Notes in Computer Science. — 2012. — Vol. 7495. — P. 121–130. doi:10.1007/978-3-642-33350-7_13
8. Martinez, G. J. Localization Dynamics in a Binary Two-Dimensional Cellular Automaton: The Diffusion Rule [Text] / G. J. Martinez, A. Adamatzky, H. V. McIntosh // Game of Life Cellular Automata. — 2010. — P. 291–315. doi:10.1007/978-1-84996-217-9_16
9. Schiff, J. L. Cellular automata: a discrete view of the world [Text] / J. L. Schiff. — A John — Wiley & Sons inc, Publication. University of Auckland, 2008. — 279 p.
10. Cenek, M. Evolving Cellular Automata [Text] / M. Cenek, M. Mitchell // Encyclopedia of Complexity and Systems Science. — 2009. — P. 3233–3242. doi:10.1007/978-0-387-30440-3_191
11. Sutner, K. Linear Cellular Automata and de Bruijn Automata [Text] / K. Sutner // Cellular Automata. Springer Netherlands. — 1999. — Vol. 460. — P. 303–319. doi:10.1007/978-94-015-9153-9_12
12. Pivato, M. Ergodic Theory of Cellular Automata [Text] / M. Pivato // Encyclopedia of Complexity and Systems Science. — 2009. — P. 2980–3015. doi:10.1007/978-0-387-30440-3_178
13. Breukelaar, R. Using a genetic algorithm to evolve behavior in multi dimensional cellular automata [Text] / R. Breukelaar, T. Bäck // Proceedings of the 2005 conference on Genetic and evolutionary computation — GECCO'05. — ACM Press, 2005. — P. 107–114. doi:10.1145/1068009.1068024
14. Schiff, J. L. Cellular Automata [Text] / J. L. Schiff. — John Wiley & Sons, Inc., 2007. — 254 p. doi:10.1002/9781118032381

15. Boccara, N. Phase Transitions in Cellular Automata [Text] / N. Boccara // Encyclopedia of Complexity and Systems Science. — 2009. — P. 6771–6782. doi:10.1007/978-0-387-30440-3_405
16. Tempesti, G. Self-Replication and Cellular Automata [Text] / G. Tempesti, D. Mange, A. Stauffer // Encyclopedia of Complexity and Systems Science. — 2009. — P. 1–24. doi:10.1007/978-3-642-27737-5_477-7
17. Weisstein, E. W. Elementary Cellular Automaton [Electronic resource] / E. W. Weisstein // MathWorld — A Wolfram Web Resource. — Available at: \www/URL: http://mathworld.wolfram.com/ElementaryCellularAutomaton.html
18. Cook, M. Universality in Elementary Cellular Automata [Text] / Matthew Cook // Complex Systems. — 2004. — № 15. — P. 1–40. — Available at: \www/URL: http://www.complex-systems.com/pdf/15-1-1.pdf
19. Степанцов, М. Е. Клеточные автоматы как модели нелинейных явлений [Текст] / М. Е. Степанцов // Труды девярых математических чтений «Математические методы и приложения». — Москва: МГСУ, 2002. — С. 141–142.

ОСОБЛИВОСТІ АПАРАТНОЇ РЕАЛІЗАЦІЇ УЗАГАЛЬНЕНОГО КЛІТИННОГО ТЕТРААВТОМАТА

Запропоновано варіант узагальненої структури комірки клітинного автомата з використанням асинхронного пристрою запам'ятовування, програмно настроюваного через інтерфейс користувача. Розглянуто кілька способів апаратної реалізації запису початкових значень в регістр станів осередку та зчитування результатів із шару поточних станів в регістр зчитування результатів: послідовний, блоковий, з використанням шару початкових станів через безпосередню послідовну або блокову адресацію, або каскадну адресацію на основі тетракодов.

Ключові слова: постбінарний клітинний автомат, тетраавтомат, розширений кодо-логічний базис, запам'ятовувачий пристрій, осередок, тетракод.

Аноприєнко Олександр Яковлевич, кандидат технічних наук, професор кафедри комп'ютерної інженерії, Донецький національний технічний університет, Донецьк, Україна, e-mail: anoprien@gmail.com.

Федоров Евгений Евгеньевич, кандидат технічних наук, професор кафедри комп'ютерних наук, Донецький національний технічний університет, Красноармійськ, Україна, e-mail: fedorovee75@mail.ru.

Іваниця Сергей Васильевич, асистент, кафедра комп'ютерної інженерії, Донецький національний технічний університет, Донецьк, Україна, e-mail: ivanitsa-serg@rambler.ru.

Альрабаба Хамза, аспірант, кафедра комп'ютерної інженерії, Донецький національний технічний університет, Донецьк, Україна, e-mail: aspirant-11@mail.ru.

Аноприєнко Олександр Якович, кандидат технічних наук, професор кафедри комп'ютерної інженерії, Донецький національний технічний університет, Донецьк, Україна.

Федоров Євген Євгенович, кандидат технічних наук, професор кафедри комп'ютерних наук, Донецький національний технічний університет, Красноармійськ, Україна.

Іваниця Сергій Васильович, асистент, кафедра комп'ютерної інженерії, Донецький національний технічний університет, Донецьк, Україна.

Альрабаба Хамза, аспірант, кафедра комп'ютерної інженерії, Донецький національний технічний університет, Донецьк, Україна.

Anoprienko Alexander, Donetsk National Technical University, Donetsk, Ukraine, e-mail: anoprien@gmail.com.

Fedorov Yevgeny, Donetsk National Technical University, Krasnoarmeynsk, Ukraine, e-mail: fedorovee75@mail.ru.

Ivanitsa Sergey, Donetsk National Technical University, Donetsk, Ukraine, e-mail: ivanitsa-serg@rambler.ru.

Alrababa Hamza, Donetsk National Technical University, Donetsk, Ukraine, e-mail: aspirant-11@mail.ru