

## СПИСОК ЛІТЕРАТУРИ

1. Компас в мире сервис-ориентированной архитектуры (SOA): ценность для бизнеса, планирование и план развития предприятия / [Биберштейн Н., Боуз С., Джонс К. и др.]. – М. : КУДИЦ-ПРЕСС, 2007. – 256 с.
2. Service oriented architecture Modeling Language (SoaML) – Specification for the UML Profile and Metamodel for Services (UPMS). Revised Submission. OMG document: ad/2008-08-04. – [Электронный ресурс] – Режим доступа: <http://www.omg.org/>
3. Cox D. E. Management of the service-oriented architecture life cycle / D.E. Cox, H. Kreger // IBM Systems Journal. – 2005. – Vol. 44, No. 4. – Pp. 709–726.
4. Кудерметов Р. К. Концептуальная и формальная модели систем с сервис-ориентированной архитектурой / Р. К. Кудерметов // Науковий вісник Чернівецького університету : збірник наук. праць. – Вип. 446: Комп'ютерні системи та компоненти. – Чернівці : ЧНУ, 2009. – С. 94–99.
5. Месарович М. Теория иерархических многоуровневых систем / М. Месарович, Д. Мако, И. Такахара. – М. : Мир, 1973. – 344 с.

Надійшла 17.08.2009

Кудерметов Р. К.

ЗАДАЧА ОПТИМІЗАЦІЇ ДЛЯ ІЄРАРХІЧНИХ СИСТЕМ  
НА БАЗІ СЕРВІС-ОРИЄНТОВАНОЇ АРХІТЕКТУРИ

Сформульовано задачу оптимізації для системи, що має ієрархічну багаторівневу структуру та побудована з використанням сервіс-орієнтованої архітектури. Визначено цільові функції кожного рівня такої системи та показано, що ця система має міжрівневу функцію якості. Розглянуто властивості міжрівневої функції за умовою, що система має властивість монотонності.

**Ключові слова:** сервіс-орієнтована архітектура, ієрархічна багаторівнева система, постулат сатисфакції, міжрівнева функція якості.

Kudermetov R. K.

OPTIMIZATION PROBLEM FOR HIERARCHICAL SYSTEMS BASED ON SERVICE-ORIENTED ARCHITECTURE

The problem of optimization is formulated for a hierarchic multilevel system which uses the foundations of Service-oriented architecture. The objective functions of each of such system levels are determined and it is shown that this system has an interlevel performance function. Properties of interlevel performance function are considered provided that the system has a monotonicity property.

**Key words:** Service-oriented architecture, hierarchical multilevel system, satisfaction postulate, interlevel performance function.

УДК 621.7.073-52

Невлюдов И. Ш.<sup>1</sup>, Шостак Б. А.<sup>2</sup>, Бабай О. Н.<sup>3</sup>, Юрьева С. В.<sup>4</sup>

<sup>1</sup>Д-р техн. наук, заведующий кафедрой Харьковского национального университета радиоэлектроники

<sup>2</sup>Канд. техн. наук, ведущий инженер ООО «НПО «Агротехника»»

<sup>3</sup>Доцент Харьковского национального университета радиоэлектроники

<sup>4</sup>Студент Харьковского национального университета радиоэлектроники

## ДИАГНОСТИКА МИКРОСХЕМ ПАМЯТИ В МИКРОПРОЦЕССОРНЫХ УСТРОЙСТВАХ

Рассмотрен новый метод диагностирования работы микросхем памяти, который позволяет проводить диагностику в процессе эксплуатации запоминающих устройств. Приведено описание принципа функционирования аппаратной части, а также описана структура программного обеспечения, позволяющего реализовать разработанный метод диагностики микросхем памяти.

**Ключевые слова:** запоминающее устройство, диагностирование, оперативное запоминающее устройство, постоянное запоминающее устройство, эмуляция, информационный сигнал, стробирующий сигнал, данные.

### ВВЕДЕНИЕ

В современных условиях производства и эксплуатации диагностика и контроль запоминающих устройств (ЗУ) является чрезвычайно актуальной задачей. И хотя в настоящее время существует большое количество методов диагностики и соответствующих программно-аппаратных комплексов, методы, позволяющие диагностировать ЗУ в процессе эксплуатации, т. е. не выпаивая микросхему из платы, недостаточно развиты. На сегодняшний момент су-

ществует несколько способов диагностирования микросхем памяти. Наиболее распространенный – тестовое диагностирование. При таком способе вначале формируется диагностическая модель объекта диагностики, генерируется набор тестовых программ и затем производится непосредственно процесс диагностирования.

Редко применяется и еще один из способов диагностирования, основанный на измерении температуры микросхем. При таком способе с помощью

© Невлюдов И. Ш., Шостак Б. А., Бабай О. Н., Юрьева С. В., 2010

специального высокочувствительного термовизиографа считывается информация о распределении тепловых полей объекта диагностирования и на основании расчетов формируется вывод об исправности или неисправности отдельных микросхем памяти.

### ПОСТАНОВКА ЗАДАЧИ

Описанные способы имеют существенные недостатки. В первом случае для диагностирования одного цифрового модуля необходимо выполнить предварительно большой объем работ (формирование диагностической модели, тестовых программ, разработка и изготовление дополнительных аппаратных средств и т. д.). Кроме того, в условиях априорной неопределенности (например, отсутствие принципиальной схемы) реализация такого способа невозможна. Во втором случае недостатками являются: во-первых, наличие специализированного дорогостоящего оборудования, во-вторых, низкая достоверность результатов расчетов. Для решения этой задачи предлагается использовать контроль микросхем памяти с применением принудительного диагностирования [1].

### ИССЛЕДОВАНИЕ ОСОБЕННОСТЕЙ ФУНКЦИОНИРОВАНИЯ МИКРОСХЕМ ПАМЯТИ

Современные системы управления применяют статические и динамические ОЗУ. В первом варианте запоминающими элементами являются триггеры, сохраняющие свое состояние, пока схема находится под питанием и нет новой записи данных. Во втором варианте данные хранятся в виде зарядов конденсаторов, образуемых элементами МОП-структур. Саморазряд конденсаторов ведет к разрушению данных, поэтому они должны периодически (каждые несколько миллисекунд) регенерироваться.

Статические ОЗУ подразделяются на асинхронные, тактируемые и синхронные (конвейерные). В асинхронных ОЗУ сигналы управления могут задаваться либо импульсами, либо уровнями. Эти сигналы могут оставаться неизменным на протяжении многих циклов обращения к памяти. В тактируемых ОЗУ некоторые сигналы обязательно должны быть импульсными. В синхронных ОЗУ синхросигналы памяти тесно связаны с тактовой частотой системы, в них используется конвейеризация тракта продвижения информации, может применяться многобанковая структура памяти и др.

Функционирование ОЗУ во времени регламентируется временными диаграммами, приведенными на рис. 1. Как видно из рисунка, для процесса чтения характерны такие временные интервалы:

$T_1$  – момент времени подачи информационного сигнала (адреса ячейки ОЗУ);

$T_2$  – момент времени подачи стробирующего сигнала (строб R/W);

$T_3$  – момент времени подачи разрешающего сигнала (CS\);

$T_4$  – момент времени подачи информационного сигнала (данные).

Для процесса записи характерны такие временные интервалы:

$T_1$  – момент времени подачи информационного сигнала (адреса ячейки ОЗУ);

$T_2$  – момент времени подачи стробирующего сигнала (строб R/W) и информационного сигнала (данные);

$T_3$  – момент времени подачи разрешающего сигнала (CS\).

Для статических ОЗУ наиболее характерны структуры 2D, 3D и 2DM, но чаще всего используют структуру 2DM. Структура 2D чаще используется при небольшой информационной емкости. В структуре 2D запоминающие элементы (ЗЭ) представляют собой прямоугольную матрицу размерностью  $M = k \times m$ , где  $M$  – информационная емкость памяти в битах,  $k$  – число хранимых слов,  $m$  – их разрядность.

При наличии разрешающего сигнала CS\ дешифратор адресного кода DC активизирует одну из выходных линий, т. е. разрешает одновременный доступ ко всем элементам выбранной строки. Элементы каждого столбца соединены вертикальной линией – линией записи/считывания. В столбцах хранятся одноименные биты всех слов. Под воздействием сигнала R/W осуществляется направление обмена усилителями чтения/записи. Недостатком такой структуры является чрезмерное усложнение дешифратора адреса (число выходов дешифратора равно числу хранимых слов).

Структура 3D характеризуется двухкоординатной выборкой запоминающих элементов. Здесь код адреса разрядностью  $n$  делится на две половины, каждая из которых декодируется отдельно, и выбирается запоминающий элемент, который находится на пересечении активных линий выходов обоих дешифраторов. Количество пересечений при этом составляет  $2^{n/2} * 2^{n/2} = 2^n$ . Общее число выходов обоих дешифраторов составляет  $2^{n/2} + 2^{n/2} = 2^{n/2+1}$ . Это меньше, чем для структуры 2D. Недостатком структуры 3D является усложнение элементов памяти, которые имеют двухкоординатную выборку.

ОЗУ структуры 2DM похожи на ОЗУ структуры 2D тем, что возбужденный выход дешифратора выбирает

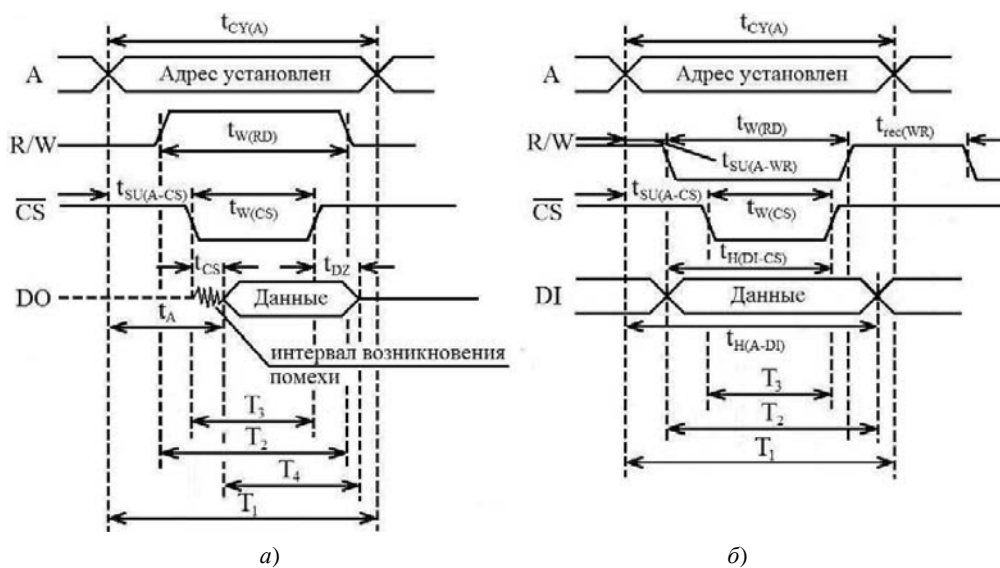


Рис. 1. Временные диаграммы процессов чтения (а) и записи (б) в статическом ЗУ

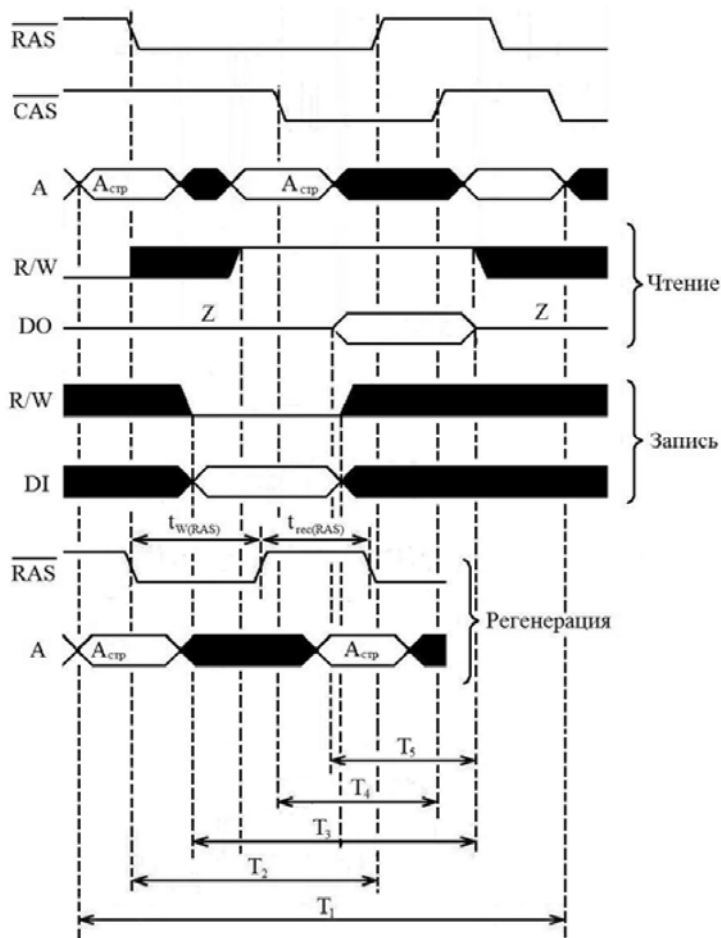


Рис. 2. Временные диаграммы динамического ЗУ

целую строку, но длина ее намного больше разрядности хранимых слов. При этом уменьшается и число строк матрицы, и число выходов дешифратора. Для выбора строки служит часть адресного кода  $A_{n-1} \dots A_k$ , остальные разряды адреса используются при выборе слова в данной строке. Диаграмма работы ЗУ типов 2D, 3D и 2DM отличается только обратной последовательностью подачи сигналов (CS\) и (R/W).

В динамических ЗУ данные хранятся в виде зарядов емкостей МОП-структур, и основой ЗЭ является конденсатор небольшой емкости. Особенностью динамических ЗУ является мультиплексирование шины адреса. Адрес делится на два полуадреса, один из которых представляет собой адрес строки, а другой – адрес столбца матрицы ЗЭ. Полуадреса подаются на одни и те же выводы корпуса ИС поочередно. Подача адреса строки сопровождается соответствующим стробом RAS\, а адреса столбца – стробом CAS\). В режимах регенерации адрес столбца не нужен. На рис. 2 приведены временные диаграммы работы динамического ОЗУ.

Для выполнения операций чтения информации выделяются такие этапы времени:

$T_1$  – момент времени подачи информационного сигнала (адреса ячейки);

$T_2$  – момент времени подачи стробирующего сигнала (RAS\);

$T_3$  – момент времени подачи стробирующего сигнала (строб R/W);

$T_4$  – момент времени подачи стробирующего сигнала (CAS\);

$T_5$  – момент времени подачи информационного сигнала (данные).

Для процесса записи выделяются такие временные интервалы:

$T_1$  – момент времени подачи информационного сигнала (адреса ячейки);

$T_2$  – момент времени подачи разрешающего сигнала (RAS\);

$T_3$  – момент времени подачи стробирующего сигнала (строб R/W) и информационного сигнала (данные);

$T_4$  – момент времени подачи стробирующего сигнала (CAS\).

Также следует отметить, что при работе микропроцессорных систем используются и постоянные запоминающие устройства (ПЗУ). В процессе их эксплуатации также необходим контроль операции записи в них информации и ее последующего считывания. Для этого можно применить рассматриваемый

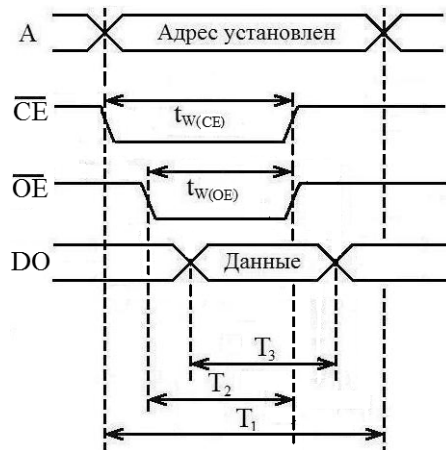


Рис. 3. Временная диаграмма работы для микросхемы ПЗУ 27C512

метод диагностики. Постоянные запоминающие устройства хранят информацию, которая либо вообще не изменяется, либо изменяется редко и не в оперативном режиме. ПЗУ обычно выполняются по структуре 2DM [2]. На рис. 3 приведен пример временной диаграммы работы для микросхемы ПЗУ 27C512.

Для выполнения операции чтения информации выделяются такие этапы времени:

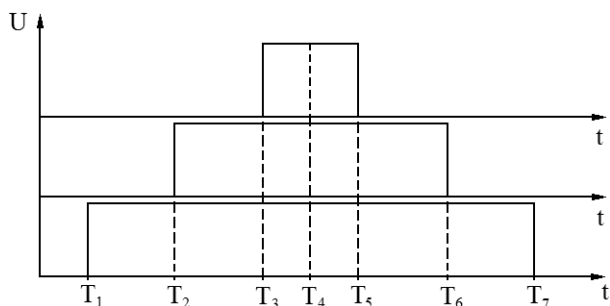
$T_1$  – момент времени подачи информационных сигналов (адреса ячейки) и разрешающего сигнала (CE\);

$T_2$  – момент времени подачи разрешающего сигнала (OE\);

$T_3$  – момент времени подачи информационных сигналов (данные).

### ПРИМЕНЕНИЕ ЭМУЛЯЦИИ РАБОЧИХ ВОЗДЕЙСТВИЙ ДЛЯ ДИАГНОСТИРОВАНИЯ МИКРОСХЕМ ПАМЯТИ

Для проведения процесса контроля с использованием эмуляции рабочих воздействий к исследуемому цифровому модулю подключаются соответствующие переходные разъемы. Эмуляционное устройство в соответствии с заданной микропрограммой осуществляет передачу всех информационных и управляющих сигналов между управляющей ЭВМ и исследуемым модулем. Микропрограмма эмуляционных воздействий формируется заранее на этапе разработки программ в той или иной системе управления технологическим оборудованием. Таким образом, эмуляция рабочих воздействий является основой формирования тестовых воздействий на все шины микросхем памяти.



**Рис. 4.** Последовательность подачи активных уровней на входы микросхемы статического ОЗУ для операции записи/чтения информации

Как известно, для тестирования ОЗУ необходима активизация входов микросхемы при помощи аппаратной части системы принудительной диагностики [3]. По функциональной принадлежности входные выходы делятся на разрешающие, информационные и стобирующие. Соответственно, активизация входов происходит в определенной последовательности.

Таким образом, как видно из приведенных выше временных диаграмм работы ЗУ, для тестирования микросхем необходимо произвести имитацию цикла работы ЗУ (рис. 4).

Процесс записи определенных данных производится в 7 этапов:

1. Установить на адресные входы дешифратора адресного кода DC адрес требуемой ячейки (момент времени  $T_1$ ).
2. Затем установить строб чтения/записи R/W и определенный набор данных для тестирования (момент времени  $T_2$ ).
3. Для разрешения работы микросхемы установить сигнал CS\ (момент времени  $T_3$ ).
4. Записать данные (момент времени  $T_4$ ).
5. Сбросить сигнал CS\ (момент времени  $T_5$ ).

6. Сбросить строб чтения/записи R/W и набор данных для тестирования (момент времени  $T_6$ ).

7. Сбросить адрес требуемой ячейки (момент времени  $T_7$ ).

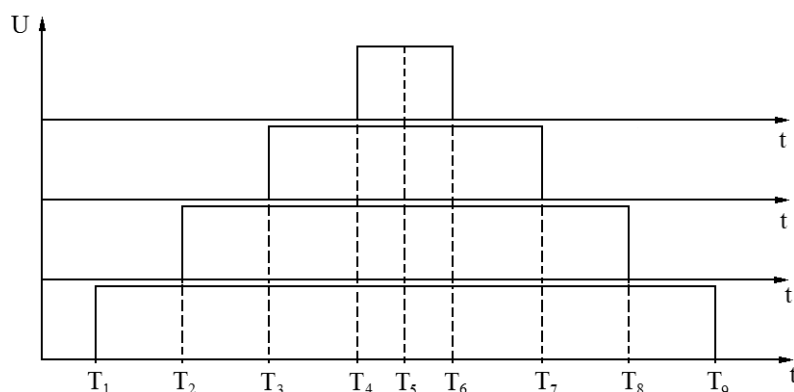
Для процесса чтения записанных данных необходимо 7 этапов:

1. Установить на адресные входы дешифратора адресного кода DC адрес требуемой ячейки (момент времени  $T_1$ ).
2. Затем установить строб чтения/записи R/W и сигнал разрешения по выходу OE\ (момент времени  $T_2$ ).
3. Для разрешения работы микросхемы установить сигнал CS\ (момент времени  $T_3$ ).
4. Считать данные (момент времени  $T_4$ ).
5. Сбросить сигнал CS\ (момент времени  $T_5$ ).
6. Сбросить строб чтения/записи R/W и сигнал разрешения по выходу OE\ (момент времени  $T_6$ ).
7. Сбросить адрес требуемой ячейки (момент времени  $T_7$ ).

Затем необходимо сравнить выходные данные с данными диагностической модели и определить неисправности.

Соответственно, для тестирования динамических ОЗУ при имитации цикла работы ЗУ необходимо 9 этапов (рис. 5):

1. Установить полуадрес необходимой ячейки (момент времени  $T_1$ ).
2. Установить сигнал RAS\ (момент времени  $T_2$ ).
3. Затем установить строб чтения/записи R/W и определенный набор данных для тестирования (момент времени  $T_3$ ).
4. Установить сигнал CAS\ (момент времени  $T_4$ ).
5. Записать данные (момент времени  $T_5$ ).
6. Сбросить сигнал CAS\ (момент времени  $T_6$ ).



**Рис. 5.** Последовательность подачи импульсов на входы микросхемы динамического ОЗУ для операции записи/чтения информации

7. Сбросить строб чтения/записи R/W и подаваемые данные (момент времени  $T_7$ ).

8. Сбросить сигнал RAS\ (момент времени  $T_8$ ).

9. Сбросить полуадрес необходимой ячейки (момент времени  $T_9$ ).

Для процесса чтения записанных данных необходимо 9 этапов:

1. Установить полуадрес необходимой ячейки (момент времени  $T_1$ ).

2. Установить сигнал RAS\ (момент времени  $T_2$ ).

3. Затем установить строб чтения/записи R/W (момент времени  $T_3$ ).

4. Установить сигнал CAS\ (момент времени  $T_4$ ).

5. Считать данные (момент времени  $T_5$ ).

6. Сбросить сигнал CAS\ (момент времени  $T_6$ ).

7. Сбросить строб чтения/записи R/W (момент времени  $T_7$ ).

8. Сбросить сигнал RAS\ (момент времени  $T_8$ ).

9. Сбросить полуадрес необходимой ячейки (момент времени  $T_9$ ).

Затем так же, как в первом случае, выходные данные сравниваются с данными диагностической модели и определяются неисправности.

Подача данных последовательностей импульсов на микросхему памяти и считывание полученных данных производится при помощи системы принудительной диагностики. Зонд для подачи тестовой информации подключается к ОЗУ прямо на плате. При этом формирование тестовой информации может производиться любым из существующих методов. Например, тест «шахматный код» используется для проверки взаимовлияния ячеек, содержащих информацию в обратном коде. В ОЗУ через зонд записывается информация, имеющая шахматное распре-

ление (01010101, 10101010, 01010101 и т. д.), затем производится последовательное считывание и проверка этой информации. «Сканирующий» тест предназначен для проверки ОЗУ в условиях максимальной статической помехи, вызванной суммарным током утечки всех ячеек ОЗУ, находящихся в одном состоянии. Производится запись нулей (единиц) во все ячейки ОЗУ, затем производится последовательное считывание и проверка информации. После этого во все ячейки ОЗУ записываются единицы (нули) и процесс повторяется [4]. Для оценки эффективности необходимо проанализировать количество тестовых наборов отдельных типов микросхем памяти [5]. В процессе диагностики при полном переборе двоичных комбинаций общее количество тестовых наборов определяется как

$$N = 2^{(I+U)},$$

где  $N$  – общее количество тестовых наборов;  $I$  – разрядной информационной шины;  $U$  – количество управляющих сигналов.

Аппаратная часть системы принудительной диагностики подключается к ЭВМ посредством 32-разрядного унифицированного интерфейсного модуля сопряжения PCI 1733 фирмы Advantech (рис. 6). Программное обеспечение системы принудительной диагностики состоит из трех частей:

1. Программа, формирующая последовательность управляющих и стробирующих кодов для каждого тестового набора.

2. Программа, активизирующая работу системы принудительной диагностики.

3. Программа, анализирующая полученные на предыдущем этапе данные.

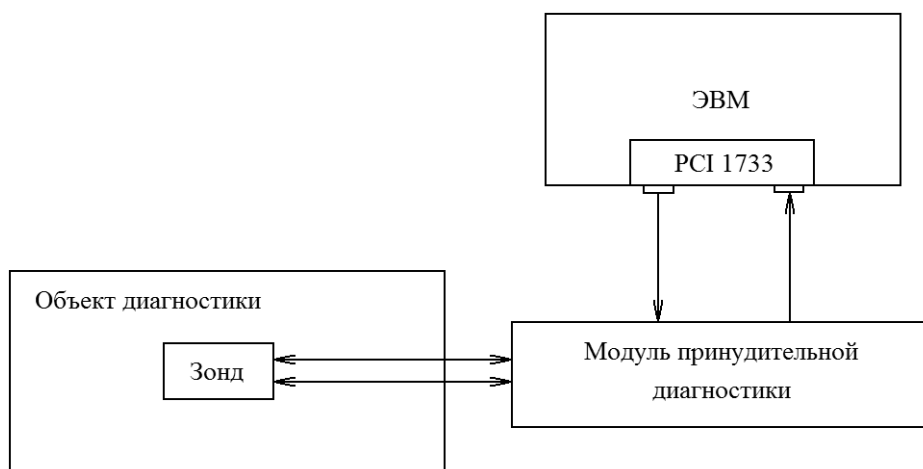


Рис. 6. Схема подключения аппаратных узлов

## ВЫВОДЫ

Предложенная система принудительного тестирования позволяет проводить оперативную диагностику микросхем ОЗУ. Время проверки микросхем составляет несколько секунд (например, для микросхемы М62256 время проверки составляет 3 секунды). Представленный метод позволяет сохранить целостность платы и подходит для большинства современных ЗУ. Также в перспективе возможно использования этого метода и для памяти с конвейерной организацией. Применение данного метода позволяет существенно ускорить процесс диагностики ЗУ, что крайне необходимо в условиях современного производства.

## СПИСОК ЛИТЕРАТУРЫ

1. Невлюдов И. Ш. Программно-технические средства диагностики цифровых модулей систем управления технологическим оборудованием / Невлюдов И. Ш., Омаров М. А., Шостак Б. А. – Х. : НТМТ, 2008. – 216 с.
2. Угрюмов Е. П. Цифровая схемотехника / Угрюмов Е. П. – СПб. : БХВ – Санкт-Петербург, 2000. – 528 с.
3. Хаханов В. И. Техническая диагностика цифровых и микропроцессорных структур: учеб. пособие / Хаханов В. И. – Киев : ИСИО, 1995. – 242 с.
4. Хаханов В. И. Контроль и диагностика вычислительных устройств и систем : учеб. пособие / Хаханов В. И. – Харьков : ХВУ, 1997. – 304 с.

5. Хаханов В. И. Техническая диагностика элементов и узлов персональных компьютеров : учеб. пособие / Хаханов В. И. – Киев : ИЗМН, 1997. – 308 с.

Надійшла 01.09.2009

Невлюдов І. Ш., Шостак Б. О., Юр'єва С. В., Бабай О. М.  
**ДІАГНОСТИКА МІКРОСХЕМ ПАМ'ЯТІ В МІКРОПРОЦЕСОРНИХ ПРИСТРОЯХ**

Розглянуто новий метод діагностування роботи мікросхем пам'яті, який дозволяє проводити діагностику у процесі експлуатації запам'ятовуючих пристроїв. Наведено структурну схему апаратної частини, а також описано структуру програмного забезпечення, яке дозволяє реалізувати розроблений метод діагностики мікросхем пам'яті.

**Ключові слова:** запам'ятовуючий пристрій, діагностування, оперативний запам'ятовуючий пристрій, постійний запам'ятовуючий пристрій, емуляція, інформаційний сигнал, стробуючий сигнал, дані.

Nevlyudov D. E. I., Shostak C. T. B., Yurieva S., Babay O.  
**DIAGNOSTICS OF MEMORY CHIPS IN MICROPROCESSOR-BASED UNITS**

A new method of memory chips diagnostics is described which makes it possible to perform diagnostics of operating memory devices. The block diagram of hardware component is shown and software structure is described, which permits to implement the developed method of memory chips diagnostics.

**Key words:** memory, diagnostics, random-access memory, read-only memory, emulation, information signal, strobe, data.

УДК 621.372.061

Рибін О. І.<sup>1</sup>, Наталенко С. С.<sup>2</sup>, Ніжебецька Ю. Х.<sup>3</sup>

<sup>1</sup>Д-р техн. наук, завідувач кафедри Національного технічного університету України «Київський політехнічний інститут»

<sup>2</sup>Аспірант Національного технічного університету України «Київський політехнічний інститут»

<sup>3</sup>Студент Національного технічного університету України «Київський політехнічний інститут»

## ВЛАСТИВОСТІ ПЕРЕТВОРЕННЯ RTF

Проводиться порівняння точності дискретних перетворень RTF (Root domain Transfer Function) та Фур'є при використанні операцій диференціювання та інтегрування. Відмічена доцільність використання немінімальних різницевоїх формул диференціювання при застосуванні перетворення RTF.

**Ключові слова:** обробка сигналів, перетворення Фур'є, власне число, диференціювання, інтегрування.

## ВСТУП. ПОСТАНОВКА ЗАДАЧІ

Методи аналізу сигналів та систем на базі дискретизації сигналів в області натуральних координат [1–5] та в області ортогональних перетворень [6–11] знайшли широке застосування в сучасній техніці, медицині тощо, що пов'язано з комп'ютеризацією наукових досліджень і технічних розробок. При аналізі лінійних систем (розв'язанні лінійних диференціальних

рівнянь їх динамічної рівноваги) найбільш поширеними є методи, побудовані на дискретних Фур'є та Z-перетвореннях, що пов'язано з відомими властивостями цих перетворень [6–10]. Так, як відомо, операції диференціювання в області натуральних координат в області перетворення Фур'є відповідає множення на  $j \cdot \omega$ , що дозволяє звести диференціальні лінійні рівняння до відповідних їм алгебраїчних рівнянь.

© Рибін О. І., Наталенко С. С., Ніжебецька Ю. Х., 2010