

## РАСШИРЕНИЕ ФУНДАМЕНТАЛЬНЫХ ОСНОВ СОВРЕМЕННОЙ ЭЛЕМЕНТНОЙ БАЗЫ КОМПЬЮТЕРНЫХ СИСТЕМ

### Аннотация

В статье кратко рассматриваются теория и методы расширения современной элементной базы цифровых интегральных схем, на основе новых многофункциональных и многоуровневых элементарных схем памяти (МФСП и МУСП). Показано, что МФСП и МУСП обладают способностью изменять алгоритм своей работы за один машинный такт, функционируя в режиме детерминированных, вероятностных или нечетких автоматов памяти.

Предлагаемая новая технология цифровых схем позволяет обрабатывать сигналы и данные с более высокой эффективностью в пересчёте на стоимость, мощность и размеры. Это несомненно будет иметь серьезные последствия для всей цифровой индустрии, так как позволяет построить с нуля устройства и компьютерные системы, обладающие новыми характеристиками и свойствами для широкого спектра приложений в науке, технике и производстве.

**Ключевые слова:** элементная база, переходы в схемах памяти, элементарные реконфигурируемые схемы памяти, принцип программного управления.

### Abstract

This article briefly reviews the theory and methods of the enhancement of modern element base of digital integrated circuits, based on the new multi-functional and multi-level elementary circuits of memory (MFCM and MLCM). It is shown that MFCM and MLCM are able to change the working algorithm in a single machine clock cycle, operating in deterministic, stochastic or fuzzy modes.

The proposed novel technology of digital circuits enables capabilities for processing signals and data with greater efficiencies in cost, power and size. This undoubtedly will have serious implications for the entire digital industry as it allows to build from the ground up devices and computer systems, possessing new characteristics and properties for a wide variety of applications in science, engineering and manufacturing industries.

**Keywords:** electronic components, the transitions in the memory circuits, basic reconfigurable memory circuit, the principle of program management.

## ВВЕДЕНИЕ

Развитие вычислительной техники характеризуется в настоящее время возрастающими требованиями к средствам переработки информации, в первую очередь – к производительности, надежности и расширению функциональных возможностей. Решающая роль принадлежит фундаментальным исследованиям [1].

В настоящее время почти все фирмы-разработчики компьютеров и нейрокомпьютеров стремятся отойти от неймановских вычислительных структур, элементная база которых основана на использовании комбинационных схем и двоичных схем памяти [2–3].

В конце 90-х годов XX века украинским и российским разработчикам вычислительных систем стали доступны программируемые логические интегральные схемы (ПЛИС) нового поколения, в английской аббревиатуре - FPGA (Field Programmable Gates Array)[1; 4]. Следует отметить, что ПЛИС в настоящее время используются многими ведущими производителями компьютерной техники [4-5].

Однако, применение ПЛИС только решило вопрос реконфигурации комбинационных схем, не изменив идеологию современной базы, используемой в сверхбольших интегральных схемах (СБИС).

## ПРОБЛЕМЫ ПОИСКА НОВЫХ РЕКОНФИГУРИРУЕМЫХ ЭЛЕМЕНТАРНЫХ СХЕМ ПАМЯТИ

А в области схем памяти? Как здесь обстоят фундаментальные разработки реконфигурируемых элементарных схем памяти? Здесь, в первую очередь, необходимо отметить работы представителей научных школ, таких как: В.М. Глушкова, М.А. Гаврилова, А.Д. Закревского, В.Г. Лазарева, Э.В. Евреинова, И.В. Прангишвили, Э.А. Якубайтиса, а также работы В.И. Варшавского, А.В. Каляева, А.Н. Мелихова, П.П. Пархоменко, З.Л. Рабиновича, Л.Я. Нагорного, К.Г. Самофалова и других авторов.

Состояние вопроса таково, что в основе элементарных схем памяти сверхбольших интегральных схем (СБИС) используется триггер, который имеет жесткую, закрытую, неизменяемую структуру. Разрывные характеристики электронных ламп, на которых основано действие триггеров, впервые под названием «катодное реле» были описаны русским изобретателем М. А. Бонч-Бруевичем в 1918 г.

Во всем мире продолжается поиск новых схемотехнических решений элементарных схем памяти с качественно новыми свойствами. Исследования в этой области проводились на логических элементах [6-13], на пороговых элементах [14], на нейронных элементах [2-3].

К сожалению, все рассматриваемые работы по перестраиваемой схеме памяти велись за счет перестраиваемых функций возбуждения и выходов, в основе которых находится *RS*-триггер, имеют фундаментальные ограничения:

1. все они работают в автоматном дискретном времени  $t_i (i= 1, 2, \dots, n, \dots)$ ;
2. базовая схема памяти (*RS*-триггер) не позволяет перестраивать работу запоминаемых состояний;
3. описываются все эти устройства автоматами Мили и Мура, которые определяют последовательный характер работы устройств;
4. переход в схемах памяти происходит по одной переменной  $x(t)$ ;
5. используемый принцип программного управления, предложенный Ч. Беббиджем, не позволяет одновременную обработку общей и частной информации.

## МНОГОФУНКЦИОНАЛЬНЫЕ СХЕМЫ ПАМЯТИ

Использование многофункциональных (МФСП) и многоуровневых (МУСП) схем памяти, которые функционируют в автоматном непрерывном времени  $T_i = t_i + \Delta_i$ , позволяет снизить основные фундаментальные ограничения современной базы СБИС, на основе которой строятся компьютеры и нейрокompьютеры и другие устройства компьютерной техники [15].

На рис. 1.а представлено автоматное дискретное время, в котором точка  $t_i$  обозначает время перехода в новое состояние в устройствах с памятью на триггерах под воздействием входной переменной  $x(t)$ . На рис. 1.б представлено автоматное непрерывное время. В устройствах с памятью на МФСП с двухтактной системой синхронизации во время  $t_{\tau 1}$  возможен переход в новое состояние в подмножестве  $\pi_i$  состояний  $a_i(t)$  под воздействием входной переменной  $x_1(t)$ . А во время  $\Delta_0$  возможен переход в новое состояние  $a_k(\Delta)$  в подмножестве  $\mu_j$  состояний под воздействием входной переменной  $e_1(\Delta)$  состояний. Таким образом, в МФСП возможен переход по двум переменным  $x_1(t)$  и  $e_1(\Delta)$  за время  $T_0$  в состояние  $a_k(\Delta)$ , которое сохраняется во времени  $T_{\tau 1}-T_0$  в схеме памяти (табл. 1).

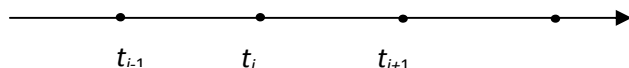


Рис. 1. а. Автоматное дискретное время

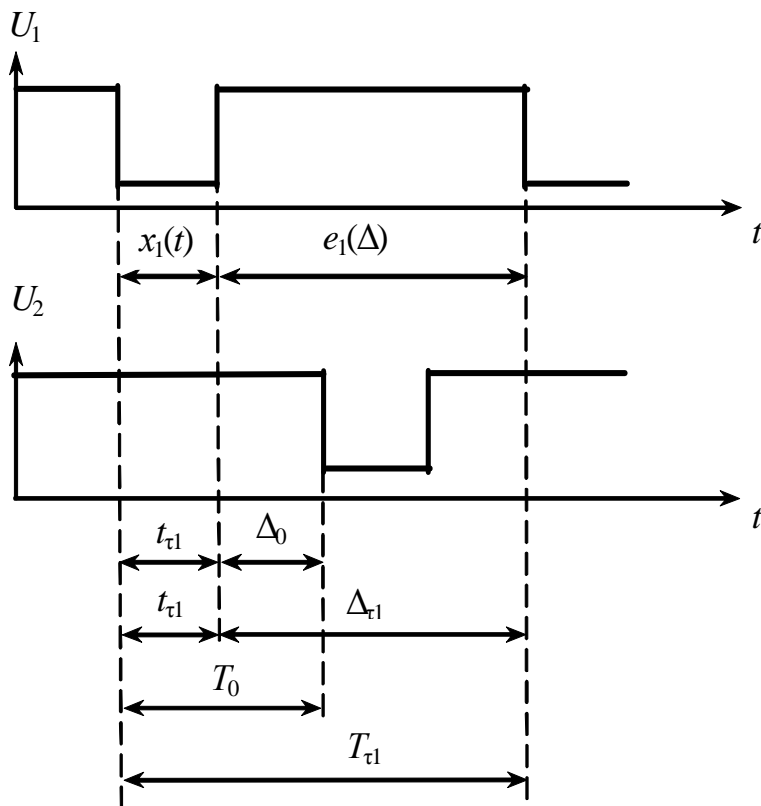


Рис. 1. б. Автоматное непрерывное время при двухсерийной синхронизации

Таблица 1

Матрица состояний многофункциональной схемы памяти

	$\mu_1$	$\mu_2$	....	$\mu_n$
$\pi_0$	$a_{10}$	$a_{20}$	...	$a_{n0}$
$\pi_1$	$a_{11}$	$a_{21}$	...	$a_{n1}$
$\pi_2$	$a_{12}$	$a_{22}$	...	$a_{n2}$
...	...	...	...	...
$\pi_m$	$a_{1m}$	$a_{2m}$	...	$a_{nm}$

Автор дает формулировку расширенной теоремы о структурной полноте [15].

*Каждая система элементарных автоматов, содержащая элементарный многофункциональный автомат (МФСП), который имеет полные системы переходов, выходов и систему функций сохранения состояний (где количество функций не меньше двух), и любую функционально полную систему логических элементов, является структурно полной системой.*

Эта теорема и обуславливает новую элементную базу СБИС.

Входная переменная  $e(\Delta)$  в теории [16] алгоритмов называется «пустым словом нулевой длины», которой в автоматное дискретное время не выделена длина и под ее действием триггер не способен перейти в новое состояние. В автоматном непрерывном времени входной переменной

$e(\Delta)$  выделено время, в связи с тем, что она активно участвует в переходах многофункциональных схемах памяти.

На основе работы многофункциональной схемы памяти, используемой в устройствах компьютерной техники, автором разработаны многофункциональные автоматы 1-го, 2-го и 3-го рода, частным случаем которых являются последовательные автоматы Мили и Мура [15].

Закон функционирования многофункционального абстрактного автомата в 1-го рода задается уравнениями:

$$\begin{cases} a(t) = \delta_0(a(\Delta-1), x(t)); \\ a(\Delta) = \delta_e(a(t), e(\Delta)); \\ y_L^1(t) = \lambda_1(a(\Delta-1), x(t)), \\ a(t), a(\Delta) \in \pi_j; i = 0, 1, 2, \dots; \Delta = 0, 1, 2, \dots \end{cases} \quad (1)$$

Закон функционирования многофункционального абстрактного автомата 2-го рода задается уравнениями:

$$\begin{cases} a(t) = \delta_0(a(\Delta-1), x(t)); \\ a(\Delta) = \delta_e(a(t), e(\Delta)); \\ y_L^2(T) = \lambda_2(a(t), a(\Delta)), \\ a(t), a(\Delta) \in \pi_j; i = 0, 1, 2, \dots; \Delta = 0, 1, 2, \dots \end{cases} \quad (2)$$

Закон функционирования многофункционального абстрактного автомата 3-го рода задается уравнениями:

$$\begin{cases} a(t) = \delta_0(a(\Delta-1), x(t)); \\ a(\Delta) = \delta_y(a(t), e(\Delta)); \\ y_L^3(\Delta) = \lambda_3(a(\Delta), e(\Delta)), \\ a(t) \notin \pi_j, a(\Delta) \in \pi_j; i = 0, 1, 2, \dots; \Delta = 0, 1, 2, \dots \end{cases} \quad (3)$$

Установлением законов функционирования многофункциональных абстрактных автоматов 1-го, 2-го и 3-го рода обобщенного многофункционального абстрактного  $M$ -автомата заканчивается определение абстрактного автомата.

Детерминированные переходы в многофункциональных автоматах 1-го и 2-го рода являются однозначными в подмножестве  $\pi_i$  состояний, а в автоматах 3-го рода – укрупненными в подмножестве  $\pi_i$  состояний всей матрицы (табл. 1).

Асинхронный  $RS$ -триггер состоит из двух групп, в каждой из которых содержится один логический элемент И-НЕ (ИЛИ-НЕ). В многофункциональных схемах памяти (МФСП) в каждой группе может содержаться не менее двух логических элементов И-НЕ (ИЛИ-НЕ). В отличие от инженерного подхода построения двоичных триггеров на основе асинхронного  $RS$ -триггера, разработана теория построения двух классов МФСП: класса  $L$  и класса  $L^M$  по необходимым требованиям количества  $M$  запоминаемых состояний и/или количеству изменяемых подмножеств  $\pi_i$  состояний.

Число запоминающих состояний  $K_i$  в  $i$ -й группе МФСП определяется по формуле:

$$K_i = 2^{R_i} - 1, \quad (1)$$

где  $R_i$  – число логических элементов в  $i$ -ой группе МФСП.

Количество  $M$  устойчивых состояний  $a(\Delta)$  МФСП, хранящихся под воздействием сохраняющих  $e(\Delta)$  входных сигналов, определяется формулой:

$$M = \sum_{i=1}^m K_i, \quad (2)$$

где  $K_i$  - число запоминающих состояний в  $i$ -й группе МФСП.

Число  $r_e$  различных наборов сберегающих  $e(\Delta)$  входных сигналов МФСП можно определять формулой:

$$r_e = \prod_{i=1}^m K_i \quad (3)$$

где  $K_i$  - число запоминающих состояний в  $i$ -й группе МФСП.

Таким образом, снижены существующие фундаментальные ограничения:

1. расширено автоматное дискретное время до уровня непрерывного времени;
2. базовая многофункциональная схема памяти позволяет перестраивать работу запоминаемых подмножеств  $\pi_i$  состояний;
3. расширены последовательные автоматы Мили и Мура до уровня многофункциональных автоматов Мараховского;
4. переход в многофункциональных схемах памяти может происходить по двум переменным  $x(t)$  и  $e(\Delta)$ , что позволяет перестраивать подмножества  $\pi_i$  состояний.

### МНОГОУРОВНЕВЫЕ СХЕМЫ ПАМЯТИ

Многофункциональные схемы памяти (рис. 2) имеют два множества входных сигналов  $x(t)$  и  $e(\Delta)$ . Особенностью этих двух наборов входных сигналов является то, что устанавливающий  $x_i(t)$  входной сигнал поглощает сохраняющий  $e_j(\Delta)$  входной сигнал, если они поступают в одно и тоже время.

$$x_i(t) = x_i(t) \cup e_j(t) \quad (4)$$

Это очень важное фундаментальное понятие для построения многоуровневых схем памяти (МУСП). На основе МФСП строится многоуровневая структура, в которой генерация входных сигналов  $e(\Delta)$  для МФСП осуществляется выходными сигналами автомата стратегии (рис. 3). Автомат стратегии может быть многозначным триггером или многоуровневой схемой памяти [15].

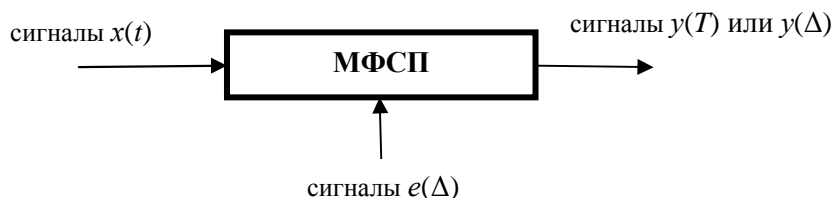
Устанавливающие входные сигналы  $x_M(t)$  поступают на автомат стратегии и  $x_y(t)$  на МФСП параллельно, а внутри многоуровневой схемы памяти из автомата стратегии генерируются сохраняющие входные сигналы  $e(\Delta)$  на МФСП. Это позволяет одновременно за один машинный такт  $T$  запоминать в автомате стратегии общую информацию, а в МФСП – частную. Такое новое качество позволяет перестраивать структуру запоминания подмножества  $\pi_i$  состояний в МФСП без потери быстродействия [15].

Из структуры МФСП можно вычислить количество состояний всей МУСП, число которое определяется по формуле:

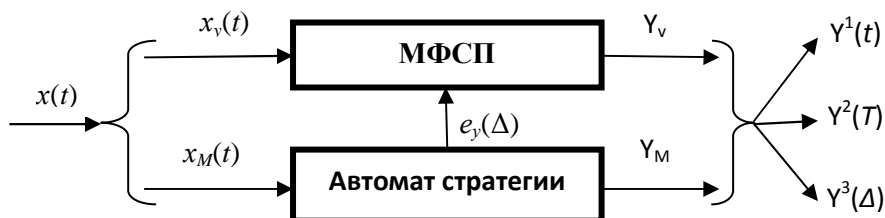
$$M = m_k \times r_e, \quad (5)$$

где  $m_k$  - число групп в МФСП;

$r_e$  - число наборов  $e_j(\Delta)$  входных сигналов МФСП, определяемых по формуле (3).



**Рис. 2.** Структура многофункциональной схемы памяти



**Рис. 3.** Структура многоуровневой схемы памяти

Разработаны два типа многоуровневых схем памяти (МУСП): класса  $L_N$  с общим автоматом стратегии (рис. 3) и  $L_N^B$  с автоматами стратегии для каждой группы МФСП, в которых число логических элементов больше 1 [15].

### АППАРАТНЫЕ И ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ ПРЕДЛАГАЕМЫХ СХЕМ ПАМЯТИ

Многофункциональные схемы памяти способны запоминать большее число запоминаемых состояний, чем многозначный триггер, имеющий одни и те же ограничения логических элементов И-НЕ (ИЛИ-НЕ).

Поясним это на примере. Предположим, логические элементы И-НЕ (ИЛИ-НЕ) имеют 10 входных узлов и нагрузочную способность на 10 логических элементов.

Тогда, максимальный многозначный триггер (многостабильные элементы (МСЭ)) сможет запоминать не больше 10 состояний, используя в каждой из десяти групп по одному логическому элементу И-НЕ (ИЛИ-НЕ). В этом случае, количество внутренних связей – 90, входных и выходных узлов 20, нагрузочная внешняя способность каждого логического элемента остается равной 1.

Многофункциональная схема памяти класса  $L^M$  по сравнению с максимальным многозначным триггером, имеет в одной группе 2, а в другой группе 3 логических элемента И-НЕ (ИЛИ-НЕ) и 2 логических элемента И (ИЛИ) способна тоже запоминать 10 состояний (на 30% меньше логических элементов), имеет 7 внутренних связей (почти в 13 раз меньше внутренних связей), входных и выходных узлов – 14 (что на 33% меньше), нагрузочную внешнюю способность каждого логического элемента – 9 (что в 9 раз больше).

К тому же, эта МФСП имеет возможность еще работать как 21 триггер, перестраивая свои подмножества  $\pi_i$  сохраняемых состояний, что триггер принципиально осуществить не в состоянии.

Максимальная МФСП класса  $L^M$  на ограничениях таких же логических элементах И-НЕ (ИЛИ-НЕ), имеющая 10 групп по 10 логических элементов И-НЕ (ИЛИ-НЕ) и по одному логическому элементу И (ИЛИ) в каждой группе, может иметь до 10230 состояний, что превышает возможности 10-значного триггера в 1023 раза, а самое главное – это функциональная возможность перестраивать подмножества  $\pi_i$  сохраняемых состояний.

МУСП обладают свойством одновременно запоминать и сохранять общую и частную информации, что позволяет их одновременную обработку в комбинационных схемах. Такую

параллельную обработку общей и частной информации осуществить принципиально невозможно в устройствах с памятью на триггерах. Такая процедура в устройствах с памятью на триггерах происходит последовательно в два этапа: вначале настраивается автомат стратегии, обрабатывающий общую информацию, а потом выбирается в другом многофункциональном устройстве для обработки частная информация.

МУСП может быть не только двухуровневая, но и многоуровневая, за счет построения многоуровневого автомата стратегии.

МУСП обладает меньшими аппаратными затратами на одно запоминаемое состояние, обладает меньшими внутренними связями и т.д. по сравнению с триггером [15].

### ДОПОЛНИТЕЛЬНЫЕ ПЕРЕХОДЫ В ПРЕДЛАГАЕМЫХ СХЕМАХ ПАМЯТИ

Однозначные переходы в двоичных схемах памяти используются только в детерминированном режиме (рис.4). Отсюда и вытекает построение последовательных машин с памятью на триггерах, в которых используется последовательное программное обеспечение на всех языках программирования.

В отличие от последовательных компьютеров, использующих в качестве памяти триггеры, построение компьютеров на схемах МФСП и МУСП дополняет количество переходов на элементарном уровне схем памяти, что позволяет расширить возможности детерминированного и вероятностного программного обеспечения [18–19].

Рассмотрим в виде направленных графов однозначные переходы, которые осуществляются в триггерах и в схемах МФСП и МУСП при неизменном сохраняющем входном сигнале  $e(\Delta)$ .

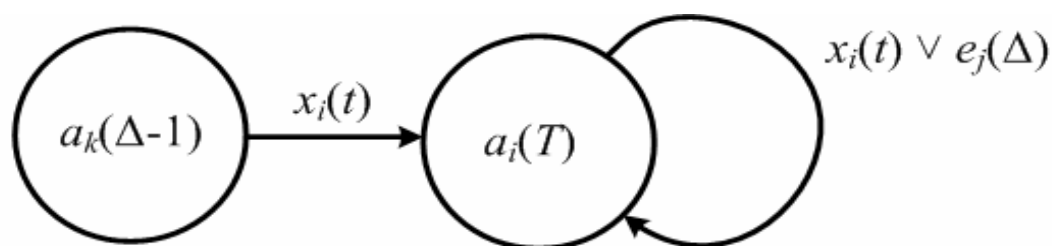


Рис. 4. Однозначный переход

При использовании МФСП и МУСП возможен детерминированный укрупненный переход по всей матрице (табл. 1) по двум переменным входных сигналов  $x(t)$  и  $e(\Delta)$ , который позволяет осуществлять иерархический переход из одного уровня (строки матрицы) на другой (рис. 5).

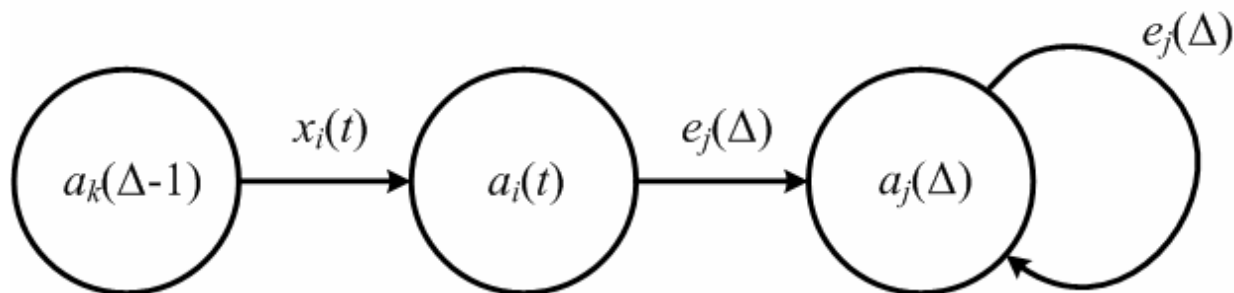


Рис. 5. Укрупненный переход

Недетерминированный вероятностный переход из определенного состояния  $a_k(\Delta-1)$  осуществляется при поступлении элементарного входного слова  $p_v(T) = x_p(t), e_j(\Delta)$ . Поступающий устанавливающий входной сигнал  $x_p(t)$ , при котором устанавливается состояние на всех выходных узлах схемы памяти не активными сигналами, не сохраняется ни при одном входном сигнале  $e_j(\Delta)$ . Вероятностный переход под воздействием входного сигнала  $e_j(\Delta)$  из слова  $p_v(T)$  осуществляется в состояние одного определенного подмножества (блока  $\pi_j$ ) состояний МФСП (рис. 6).

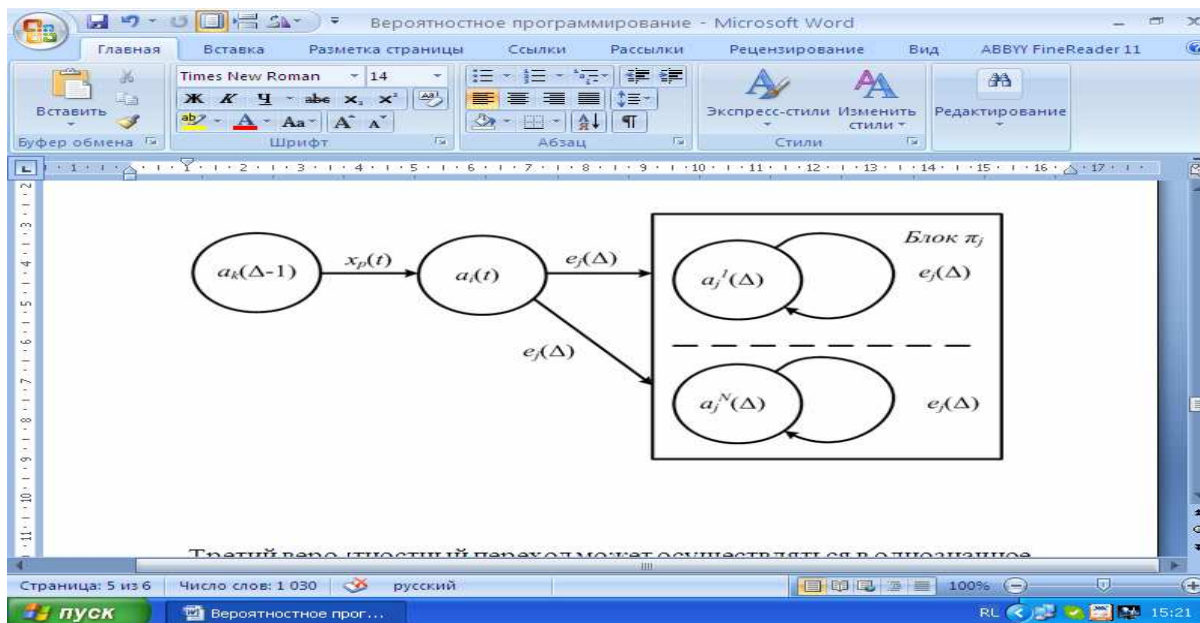


Рис. 6. Вероятностный переход в состояние подмножества (блока  $\pi_j$ )

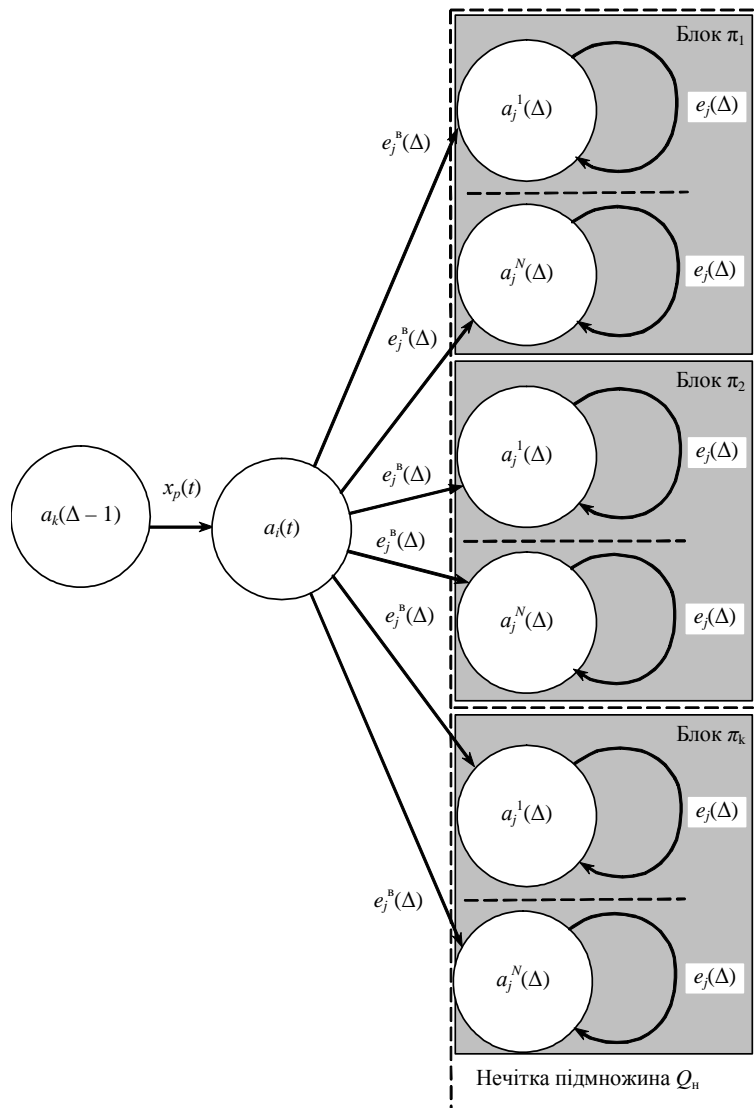
Вероятность  $P_v$  этого перехода определяется экспериментально при исследовании МУСП. Так, при многократных испытаниях МФСП с двумя группами по два логических элемента И-НЕ в группе вероятность  $P_v$  перехода в одно из состояний блока  $\pi_j$  равнялась 0,3, а для второго – 0,7.

Недетерминированный нечеткий переход из определенного состояния  $a_k(\Delta-1)$  в состояние определенного нечеткого подмножества (блоков)  $Q_n$ , состоящего из четких блоков  $\pi_j$ , осуществляется при поступлении элементарного входного слова  $p_n(T) = x_p(t), e_j^e(\Delta)$ . Поступающий входной сигнал  $x_p(t)$ , при котором выходной сигнал МФСП не сохраняется ни при одном входном сигнале  $e_j(\Delta)$  дает возможность осуществить вероятностный нечеткий переход под воздействием вероятностного входного сигнала  $e^e(\Delta)$  в состояние нечеткого подмножества  $Q_n$  МФСП (рис. 7).

Вероятность  $P_n$  нечеткого перехода определяется экспериментально при исследовании работы МУСП.

Такая возможность позволила в программном обеспечении ввести новую структуру команды, в которой добавлен еще общий код команды, способный изменять структуру команд в процессоре в одном машинном такте  $T$ , что в известных языках программирования осуществить принципиально невозможно.





**Рис. 7.** Нечеткий переход в нечеткое подмножество  $Q_n$

Для устройств, которые используют в качестве памяти МФСП и МУСП, такие команды используются как иерархические. Это связано с тем, что они рассчитаны на параллельную обработку информации, где общая и частная информация обрабатывается параллельно за один машинный такт  $T$ , и для этого требуют дополнительной ячейки в адресе команды, которая бы определяла код операнда общей информации.

В связи с этим предлагается ввести в коды команд дополнительный код для общей операции.

Вид параллельной трехадресной команды

Код общей операции	Код частной операции	Код адреса ячейки 1-го операнда	Код адреса ячейки 2-го операнда	Код адреса ячейки 3-го операнда
--------------------	----------------------	---------------------------------	---------------------------------	---------------------------------

Вид параллельной одноадресной операции

Код общей операции	Код частной операции	Код адреса ячейки операнда
--------------------	----------------------	----------------------------

Надо отметить, что код частной операции обрабатывает информацию операнда, а код общей операции изменяет код частной операции.

### **ПРИНЦИП ИЕРАРХИЧЕСКОГО ПРОГРАММНОГО УПРАВЛЕНИЯ**

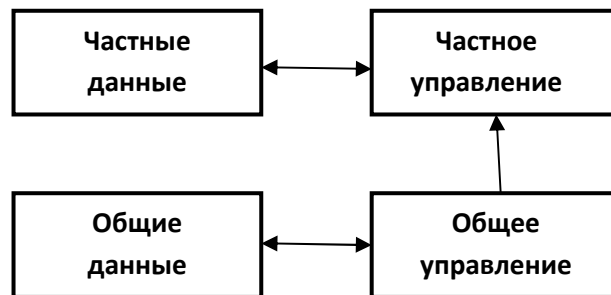
Чарльз Бэббидж предложил фундаментальный принцип программного управления, заключающийся в том, что информация разбивается на два типа: обрабатываемую и управляющую. Этот принцип послужил теоретической предпосылкой для работ по автоматизации численных методов уже в первых вычислительных машинах [17].

Автор предложил принцип иерархического программного управления [18]. Принцип заключается в том, что информация, обрабатываемая и управляющая, разбивается на частную и общую. Общая и частная информация взаимосвязаны между собой по вертикали управляющей информации от общей информации к частной. Обрабатываются частная и общая информация параллельно по отношению друг к другу (рис. 8). Одной из основных временных характеристик обработки информации в этом случае является более быстрая обработка частной информации по отношению к общей, а одной из функциональных характеристик – изменение алгоритма обработки частной информации при определенной обработке общей информации.

Общую информацию можно также представить как частную и общую. Такое иерархическое разделение информации конечно и возможно до определенного минимального объема управляющей информации.

Частная управляющая информация при не менее, чем трехуровневом программном управлении может осуществлять детерминированные однозначные и укрупненные переходы, а также вероятностные или нечеткие [18]. Общая («корневая») управляющая информация должна обрабатываться однозначно.

Использование вероятностных и нечетких результатов вычислений наряду с появившейся возможностью многофункциональности результатов вычислений расширяют функциональные возможности вычислительных устройств и создают предпосылки для повышения уровня «машинного интеллекта» [20].

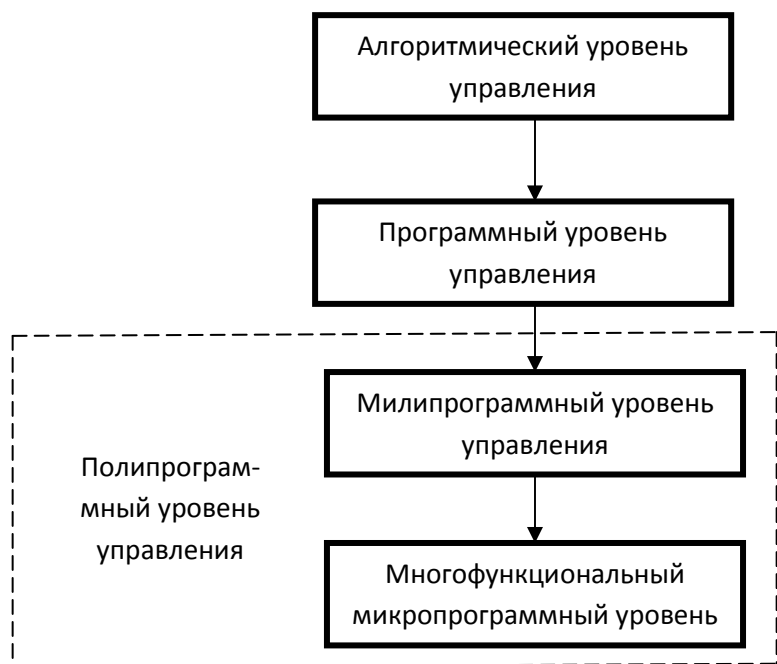


**Рис. 8.** Принцип иерархического программного управления

### **ЧЕТВЕРТЫЙ УРОВЕНЬ УПРАВЛЕНИЯ**

В настоящее время используются три уровня управления: алгоритмический, программный и микропрограммный [4–6; 17].

Четвертый уровень управления основывается на принципе иерархического программного управления и позволяет обрабатывать одновременно общую и частную управляющую информацию, которая может быть использована для построения иерархических реконфигурируемых вычислительных устройств на основе существующих аппаратных компонентов элементарных МФСП и МУСП.



**Рис. 9.** Полипрограммный уровень управления

Для того, чтобы построить реконфигурированные устройства управления на основе МФСП и МУСП, автор вводит четвертый - милипрограммный уровень управления для обработки общей информации, которая управляет многофункциональным микропрограммным уровнем при обработке частной информации (рис. 9).

### **ТИПОВЫЕ РЕКОНФИГУРИРУЕМЫЕ УСТРОЙСТВА**

К типовым реконфигурируемым устройствам относятся: устройства управления, регистры, счетчики, процессоры и компьютеры, структуры и функциональные схемы которых достаточно подробно описаны в работе [15].

### **ЗАКЛЮЧЕНИЕ**

Таким образом, снижение фундаментальных ограничений в существующей элементной базе СБИС, применяемой в устройствах компьютерных систем, за счет предлагаемых многофункциональных и многоуровневых схем памяти, позволяет создать качественно новую информационную технологию.

Эта новая информационная технология при ее применении позволит создавать конкурентоспособные устройства компьютерной техники с повышенными функциональными возможностями, то есть с повышенным «машинным интеллектом». В 2010 году появилось сообщение о вероятностном чипе, который перевернул мир вычислений отказом от двоичной логики. Это лишний раз говорит о существующих ограничениях современной базы компьютерных систем, использующих в качестве памяти триггеры.

## ЛИТЕРАТУРА

1. Каляев И.А., Левин И.И., Семерников Е.А., Шмойлов В.И. Реконфигурируемые мультиконвейерные вычислительные структуры.– Ростов-на-Дону: Издательство ЮНЦ РАН, 2008 – 397 с.
2. IBM работает над созданием "компьютерного мозга" – <http://www.cybersecurity.ru/it/82336.html>
3. Nabil Imam, Filipp Akopyan, John Arthur, Paul Merolla, Rajit Manohar, Dharmendra S Modha A Digital Neurosynaptic Core Using Event-Driven QDI Circuits. – Almaden, San Jose, CA, 2014. – 17 p.
4. Палагин А.В. Реконфигурируемые вычислительные системы: Основы и приложения / А.В. Палагин, В.Н. Опанасенко. – К.: Просвіта, 2006. – 280 с.
5. Bolotski M. Abacus: A Reconfigurable Bit Parallel Architectures. Ph.Dd.Thesis - Massachusetts Institute of Technologies, 1996. –126 p.
6. Глушков В.М., Капитонова Ю.В., Мищенко А.Т. Логическое проектирование дискретных устройств. – К.: Наук. думка, 1987. – 264 с.
7. Граф Р.Ф. Энциклопедия электронных схем / Р.Ф. Граф, В. Шиитс. – М.: ДМК, 2007. – 249 с.
8. Евреинов Э.В., Прангишвили И.В. Цифровые автоматы с настраиваемой структурой (однородные среды). – М.: Энергия, 1974. –240 с.
9. Забара С.С., Комухаев Э.И., Куссуль М.Э., Сахарин В.Г. Проектирование логических схем нейрокompьютера в элементном базисе ПЛИС XC2000 // УСиМ. – 1993. –№ 1. – С. 9–15.
10. Компьютерна схемотехніка (краткий курс) / Р.О. Процюк, В.Н. Корнейчук, П.В. Кузьменко, В.П. Тарасенко. – К.: ПП «Корнійчук», 2006. – 433 с.
11. Кузьо М.М., Русин Б.П., Шмойлов В.И. Однородні середовища – елементна база високопродуктивних обчислювальних машин // УСиМ. – 2000. – №4. С. 52–62.
12. Мищенко В.А., Козюминский В.Д., Семашко А.Н. Многофункциональные автоматы и элементная база цифровых ЭВМ / Под. ред. В.А. Мищенко. – М.: Радио и связь, 1981. – 249 с.
13. Якубайтис Э.А. Многофункциональные логичные модели//АВТ, 1976. -№2. С.1-15.
14. Айзенберг Н.Н., Иваськив Ю.Л. Многозначная пороговая логика. – К.: Наук. Думка, 1977. – 148 с.
15. Мараховский Л. Ф. Основы новой информационной технологии. Фундаментальные основы проектирования реконфигурируемых устройств компьютерных систем и искусственного нейрона: монография. Л. Ф. Мараховский, Н. Л. Михно – Germany: Saarbrcken, LAP LAMBERT, 2013. – 347 с.
16. Глушков В.М. Теория алгоритмов. –К.: КВИРТУ, 1961. – 167 с.
17. Справочник по цифровой вычислительной технике: (процессоры и память) / Б.Н.Малиновский, Е.И.Брюхович, Е.Л.Денисенко и др. / Под ред. Б.Н.Малиновского. – К.: «Техніка», 1979. – 366 с.
18. Мараховский Л.Ф. Основы теории проектирования дискретных устройств. Логическое проектирование дискретных устройств на схемах автоматной памяти: монография. – Киев: КГСУ, 1996.–128 с.
19. Marachovsky L.F. Basic Concepts to Build the Next Generation of Reconfigurable Computing Systems.– International Journal Of Applied And Fundamental Research. – 2013. – № 2 – URL: [www.science-sd.com/455-24170](http://www.science-sd.com/455-24170) (20.11.2013) – 6 с/.
20. Заде Л.А. Понятие лингвистической переменной и его применение к принятию приближенных решений: Пер. с англ.. – М.: Мир, 1976. – 165 с.
21. Попов. Л. Вероятностный чип перевернул мир вычислений отказом от двоичной логики – <http://www.membrana.ru/particle/3399>, 2010. –9 с.