

Гуменюк В.А., к.т.н., Жуков И.А., д.т.н., Гуменюк А.В. (НАУ, Украина)

## ПРИМЕНЕНИЕ НЕРАЗДЕЛИМЫХ КОДОВ «М ИЗ N» В ВЫСОКОПРОИЗВОДИТЕЛЬНЫХ ПАРАЛЛЕЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМАХ

*Рассматриваются причины возникновения проблемы автоматического контроля в ЭВМ. Анализируются возможности применения кодов «М из N», относящихся к неразделимым, в высокопроизводительных параллельных вычислительных системах. Предлагается новая схема организации вычислительных операций, с учетом особенностей такого кодирования.*

**Введение.** Вследствие постоянного повышения требований к производительности вычислительных средств и достоверности результатов решения фундаментальных и прикладных задач, объединенных понятием «*Grand challenges*», продолжает оставаться актуальной проблема реализации массовых параллельных вычислительных процессов. Сегодня существует две основные структуры высокопроизводительных вычислительных систем: многомашинные комплексы на базе микропроцессоров с фиксированным по процессорам распределением памяти (распределенная память) и многопроцессорные комплексы с распределяемой в процессе счета памятью (распределяемая память).

**Целью работы** является определение возможностей повышения реальной производительности параллельных вычислительных систем, а также уровня достоверности результатов вычислений в них, на основе неразделимых кодов «М из N».

### Постановка задачи

Анализ снижения реальной производительности высокопроизводительных вычислительных систем с ростом числа  $N$  процессоров [1, 2] позволяет установить две причины, вызывающим такую закономерность:

- пространственная, связанная с недозагрузкой процессоров, вследствие запаздывания данных при передаче их из памяти к процессору или от процессора к процессору;
- временная, связанная с недозагрузкой процессора, вследствие отсутствия данных, поставляемых другими вычислительными процессорами (синхронизация по данным).

С увеличением  $N$  свыше 32 возникают трудности с построением многопроцессорных комплексов вследствие того, что удвоенная задержка коммутатора (передачи процессор-память и память-процессор) входит во время выполнения каждой операции процессора.

Проблема с синхронизацией по данным на больших задачах возникает в высокопроизводительных вычислительных системах вследствие необходимости разработки сложнейшей программы реального масштаба времени, при которой каждый параллельный процесс выполняется на определенном для него процессоре, объеме оперативной памяти, канале передачи данных и т. д. Попытка использования ячеек памяти для синхронизации по данным может привести к ложному результату решения задачи, так как весьма вероятно использование устаревших данных вместо новых.

В [1, 2] сделаны следующие выводы в отношении направлений решения проблемы. Для решения пространственной задачи поставки данных необходимо, в частности, стремиться к:

1. Увеличению производительности одного (каждого) микропроцессора.
2. Исключению из комплекса проблемы когерентности кэш (машины *Cray* не имеют кэш).
3. Исключению из времени выполнения операций исполнительных устройств временных задержек передачи информации между исполнительным устройством и ОЗУ.



4. Увеличению пропускной способности между первичной и вторичной памятью. Для решения проблемы временных соотношений данных необходимо:

1. Задачу синхронизации по данным и распараллеливания вычислительных процессов решать аппаратными средствами в процессе выполнения задачи.
2. Концептуально исключить ОЗУ.
3. Исключить потерю процессорного времени на прерывания и синхронизацию процессов.

Далее не рассматриваются системы с небольшим количеством компонентов, в которых не представляет затруднения полноразрядная передача данных параллельным способом.

Значительные временные задержки в шинах и коммутаторах, связывающих между собой множество компьютеров в системе (например, представленная на рис. 1 архитектура кольца с полной связью по хордам – *Chordal Ring*), приводит к необходимости передачи данных (операндов) через системную шину (коммутаторы) по частям. После накопления всех разрядов на входных регистрах их достоверность проверяется схемой контроля. В работе анализируется возможность альтернативного недвоичного представления цифровой информации в системе.

### Методика решения

Существует взаимосвязь между реальной производительностью вычислительной системы и схемой ее контроля, вносящей непроизводительные временные затраты.

Обычно в микропроцессорах надежность результатов вычислений обеспечивается за счет контроля по четности и применения кодов *ECC*. В таблице 1 приведены способы контроля цифровой информации в ряде современных наиболее высокопроизводительных микропроцессорах.

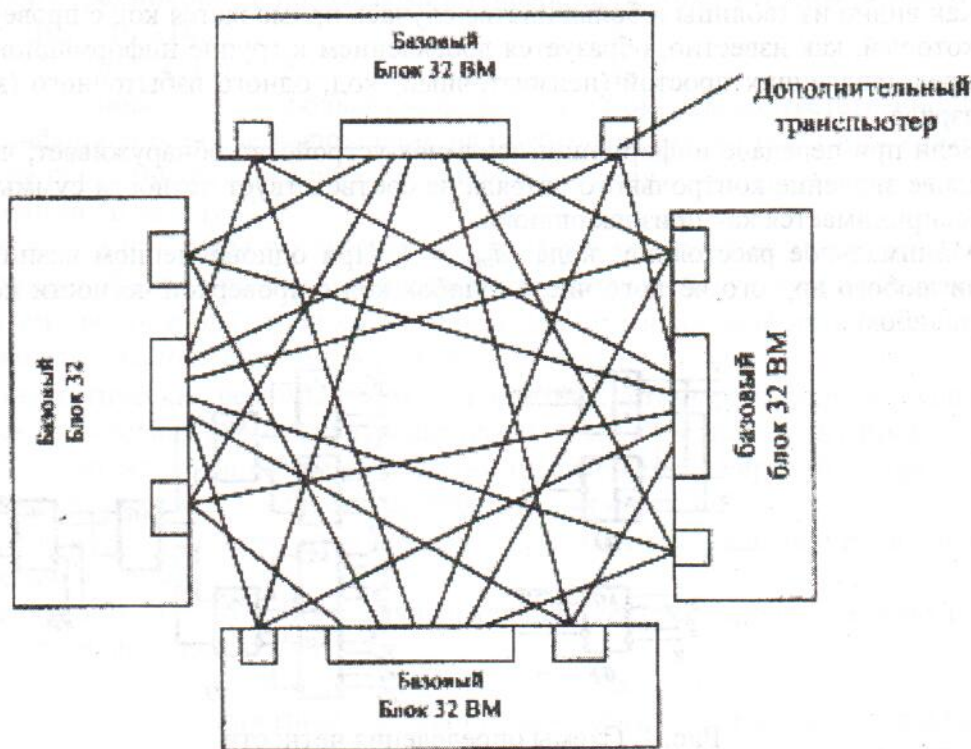


Рис. 1. Архитектура кольца с полной связью по хордам (*Chordal Ring*)



Таблица.1. Способы контроля в высокопроизводительных микропроцессорах.

	SPARC64 V1	UltraSparc III2	Itanium 2	Power4	PA-8700
АЛУ	четность + повтор	нет	нет	нет	нет
Регистры	четность + повтор	нет	нет	нет	нет
TLB	четность + исключение	четность	четность	четность	четность
L-кэш первого уровня	четность + исключение	четность	четность	четность	четность
Теги L-кэша первого уровня	дублирование + исключение	четность	четность	четность	четность
D-кэш первого уровня	ECC + исключение	четность	четность	четность	ECC
Теги D-кэша первого уровня	дублирование + исключение	четность	четность	четность	дублирование
D-кэш второго уровня	дублирование + исключение	ECC	ECC	ECC	нет кэша
Теги D-кэша второго уровня	дублирование + исключение	ECC	четность	дублирование	нет кэша

**Примечание:** повтор - автоматическое повторение операции; исключение - при неисправности отключение микропроцессора без перезагрузки; четность - контроль по четности.

Как видно из таблицы в большинстве случаев применяется код с проверкой четности, который, как известно, образуется добавлением к группе информационных разрядов, представляющих простой (неизбыточный) код, одного избыточного (контрольного) разряда.

Если при передаче информации приемное устройство обнаруживает, что в принятом слове значение контрольного разряда не соответствует четности суммы 1 слова, то это воспринимается как признак ошибки.

Минимальное расстояние кода  $d_{min} = 1$ . При одновременном возникновении двух или любого другого четного числа ошибок код с проверкой четности не обнаруживает ошибок.

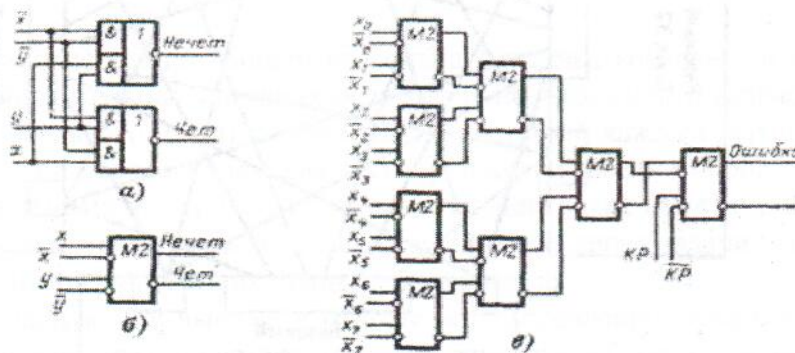


Рис. 1. Схемы определения четности

При построении схем определения четности суммы 1 слова используют логические элементы с парафазным выходом, подобные изображенному на рис. 1, а) и б). Показанные схемы выполняют операцию сложения по модулю 2 (условное обозначение M2) для двоичных переменных  $x$  и  $y$ . На рис. 1, в) показана схема определения признака четности байта.



Структура схеми проверки четности является многоступенчатой, т. е. слово делится на несколько групп разрядов, в каждой из которых проверка четности производится прямым способом (первая ступень), далее производится проверка четности для групп второй ступени, образованных из групп первой ступени, четности которых в этом случае рассматриваются как обычные двоичные разряды, и т. д. до окончательной проверки четности суммы 1 всего слова. В последней ступени четность байта сравнивается со значением контрольного разряда  $KP$ .

Основными недостатками такого способа контроля являются недостаточный уровень вероятности обнаружения большинства категорий ошибок и большая временная задержка, которую вносит схема определения четности (Рис. 1).

В [3] установлено, что традиционный способ представления двоичной информации в ЭВМ, при котором обе цифры двоичного разряда представлены одним триггером (взаимозависимое представление) привел к потере естественной контролеспособности позиционных счислений, что в свою очередь обусловило использование в ЭВМ различных избыточных кодовых построений, позволяющих обнаруживать или исправлять ошибки. Так в каждом  $i$ -ом узле ЭВМ избыточная информация строится по определенному закону  $C_i$  по отношению к основной информации. При переработке основной информации в зависимости от вида ее изменения  $A_i$  по определенным законам и правилам  $K_i$ , не совпадающим с законом  $A_i$  изменения основной информации (за исключением случая дублирования основного информационного и контрольного автоматов) видоизменяется и контрольная информация. При этом преобразование происходит таким образом, чтобы полученная в результате выполнения обеих автономных операций ( $A_i$  над основной и  $K_i$  над контрольной информацией) коды взаимно соответствовали первоначальному закону  $C_i$ .

Таким образом, основной логический закон организации аппаратного контроля узла ЭВМ при традиционном представлении информации определяется [4] как

$$\left. \begin{aligned} I_{oi} &\equiv (C_i) I_{koi}; \\ I_{oi} / A_i &\equiv (C_i) I_{koi} / K_i, \end{aligned} \right\}$$

где  $I_{oi}$  и  $I_{koi}$  – основная и контрольная информация в  $i$ -ом узле соответственно;  $I_{oi} / A_i$  – основная информация после выполнения над ней операции  $A_i$ ;  $I_{koi} / K_i$  – контрольная информация после преобразования ее по закону  $K_i$ ; символ  $\equiv (C_i)$  означает, что информация сравнима по закону  $C_i$ .

При таком принципе построения аппаратного контроля не может быть обеспечено безотказное функционирование органов контроля по следующим причинам [3]:

- система контроля строится на элементах, по паспортной интенсивности отказов одинаковых с элементами контролируемых узлов;
- энергетические режимы работы элементов системы контроля по существу не отличаются от режимов работы контролируемых элементов, поскольку процедура контроля по модулю основана на выполнении вычислительных операций по правилам вычетов одновременно с выполнением контролируемых операций;
- резервирование элементов контроля приводит к ухудшению ряда параметров вычислительной системы.

Вероятность  $D_l$  получения в вычислительной системе безошибочного (достоверного) результата определяется в [ ] как

$$D_l = 1 - (1 - P_{kl} * P_{nl} * P_{ml}) * R_l, \quad (1)$$

где  $P_{kl}$  – вероятность безотказной работы органов контроля;  $P_{nl}$  – вероятность охвата контролем оборудования вычислительной системы;  $P_{ml}$  – методическая вероятность обнаружения ошибок категории  $l$ ;  $R_l$  – вероятность возникновения ошибок категории  $l$  ( $l = 1, 2, \dots, \Psi$ ).

Из выражения (1) следует, что проблема не может быть решена на основе традиционного представления цифровой информации: с взаимозависимым представлением 0 и 1 в одном разряде одним триггером, так как не обеспечиваются условия  $P_{kl} = 1$ ,  $P_{nl} =$



1,  $P_{ml} = 1$ , при выполнении которых  $D_l = 1$  независимо от значений  $R_l$ , или  $R_l = 0$  (что, естественно, практически недостижимо). Так как, с увеличением значений модуля (при контроле по модулю) вероятность  $P_{ml}$  возрастает незначительно и это – практически полностью компенсируется неизбежным при этом уменьшением вероятности  $P_{kl}$ .

В качестве альтернативы традиционному двоичному представлению информации в вычислительных системах может быть использовано представление цифровой информации кодами « $M$  из  $N$ », в котором  $N$  – количество позиций (каждая из которых представлена отдельным триггером) в разряде числа;  $M$  – количество «единиц» в этих позициях (остальные  $N - M$  позиций содержат «нули»). Таким образом, соотношение «нулей» и «единиц» в рамках данного кода фиксировано. Такие коды относятся к неразделимым, в которых разряды кодового слова невозможно разделить на информационные и избыточные.

Изменения соответствия одноразрядных кодовых комбинаций и цифр выбранного основания  $p$  счисления приводят к образованию других алфавитов.

Как следует из работ [3, 5] при использовании кода « $M$  из  $N$ » возможно практическое решение проблемы автоматического контроля в вычислительных системах. Остановимся кратко на некоторых важных особенностях таких кодов. Максимальное количество комбинаций в одном разряде (определяется диапазон выбора величин основания счисления) вычисляется по формуле

$$C_N^M = N! / [M! * (N - M)!], \text{ где } N! = N * (N - 1) * (N - 2) * \dots * 1.$$

Величину основания  $p$  счисления выбирают из условия

$$C_N^M \geq p.$$

Наибольший практический интерес, с учетом необходимости согласования с имеющимися в эксплуатации ЭВМ, представляют значения  $p$ , выбранные из множеств  $p = 2^k$  (где  $k = 2, 3, 4, \dots$ ) и  $p = 10^m$  (где  $m = 1, 2, 3, \dots$ ).

Важнейшей характеристикой, определяющей свойство кода обнаруживать ошибки, является минимальное кодовое расстояние:

$$d = \min d(X, Y), \text{ при } X \neq Y.$$

Для кода « $M$  из  $N$ », в зависимости от четности параметра  $N$ , получаем следующие диапазоны изменения величины  $d(X, Y)$ :

для четного  $N$

$$2 \leq d(X, Y) \leq 2M, \text{ при } 1 \leq M \leq N/2;$$

$$2 \leq d(X, Y) \leq 2(N - M), \text{ при } N/2 + 1 \leq M \leq N - 1;$$

для нечетного  $N$

$$2 \leq d(X, Y) \leq 2M, \text{ при } 1 \leq M \leq (N - 1)/2;$$

$$2 \leq d(X, Y) \leq 2(N - M), \text{ при } (N - 1)/2 \leq M \leq N - 1.$$

Таким образом, для кода « $M$  из  $N$ » во всех случаях  $d \geq 2$ .

Для обнаружения любой ошибки кратности  $l$  или меньше необходимо и достаточно, чтобы минимальное расстояние кода было хотя бы на единицу больше чем  $l$  [5]:  $d \geq l + 1$ .

Следовательно, код « $M$  из  $N$ » позволяет обнаружить любую одиночную ошибку.

Полное множество категорий ошибок, возможных в таком коде:

- кодовая комбинация содержит  $M - i$  «единиц», где  $i = 1, 2, \dots, M$ ;
- кодовая комбинация содержит  $M + i$  «единиц», где  $i = 1, 2, \dots, N - M$ ;
- кодовая комбинация содержит точно  $M$  «единиц», из которых  $i$  – ложные, где  $i = 2, 4, \dots, M$ , если  $M$  – четное, или  $i = 2, 4, \dots, M - 1$ , если  $M$  – нечетное.

Обнаружение ошибок первых двух категорий (с нарушением структуры кода) обеспечивается с вероятностью  $P_{ml} = 1$ , а третьей – с вероятностью  $P_{ml} = 0$ . Следовательно, достоверность  $D_l$  цифровой информации полностью определяется вероятностью  $R_3$  образования ошибки третьей категории [3]:  $D_l = 1 - R_3$ .



Определим зависимость изменения вероятности  $R_3$  от изменения  $M$  при фиксированных значениях  $p$ .

Образование необнаруживаемой ошибки  $R_3$  возможно при совместном проявлении ошибок первых двух категорий в течении времени  $t = a * \Delta t_k$ , где  $\Delta t_k$  – разрешающая способность автоматического контроля по времени;  $0 < a < 1$  [5].

Увеличение  $M$  в диапазоне  $1 \leq M \leq N/2$ , при фиксированном  $p$ , позволяет уменьшить  $N$  (и соответственно сократить аппаратные затраты), что приводит к изменению составляющих вероятности  $R_3$  – изменению вероятностей первых двух категорий.

Вероятность  $R_3$  при  $1 < M < L$  определяется выражением

$$R_3 = \sum_{i=1}^L P_{\text{пр } i} * P_{\text{по } i},$$

где  $L = M$ , при  $N/M \leq 2$ ;  $L = N - M$ , при  $N/M > 2$ ;  $P_{\text{пр } i}$  и  $P_{\text{по } i}$  – вероятности пропадания и возникновения  $i$  символов (например, «единиц») в каждой комбинации соответственно.

Вероятности  $P_{\text{пр } i}$  и  $P_{\text{по } i}$  определяются соответственно выражениями

$$P_{\text{пр } i} = \sum_{i=1}^L C_M^i * R_0^i * (1 - R_0)^{M-i},$$

где  $R_0$  – вероятность пропадания одного символа, и

$$P_{\text{по } i} = \sum_{i=1}^L C_{N-M}^i * P_0^i * (1 - P_0)^{N-M-i},$$

где  $P_0$  – вероятность появления одного лишнего символа.

Поэтому,

$$R_3 = \sum_{i=1}^L C_M^i * R_0^i * (1 - R_0)^{M-i} * C_{N-M}^i * P_0^i * (1 - P_0)^{N-M-i}.$$

Для упрощения анализа далее будем считать, что  $R_0 = P_0$ , тогда

$$R_3 = \sum_{i=1}^L C_M^i * C_{N-M}^i * P_0^{2i} * (1 - P_0)^{N-2i}.$$

Вероятность  $P_0$  определяется выражением

$$P_0 = e^{-\lambda t K_3} = e^{-\lambda K \Delta t K_3},$$

где  $K_3$  – количество логических элементов, необходимых для представления одной цифры в ЭВМ.

Для современной элементной базы  $\lambda \approx 10^7$  1/с, а время задержки импульса при прохождении через микросхему не более  $0,3 * 10^{-6}$  с [3].

Поэтому, при  $K = \sum_{i=1}^L C_M^i * C_{N-M}^i$  получаем:

$$R_3 \approx \sum_{i=1}^L C_M^i * C_{N-M}^i * (0,3 a * K_3)^{2i} 10^{-26i}.$$

Из анализа приближенных значений вероятности  $R_3$  для множества значений  $M$  и  $N$  следует, что такая вероятность, определяемая для кодов, имеющих одинаковое количество кодовых комбинаций, уменьшается при возрастании величины  $M$  от 1 до  $N - 1$ , если это возрастание сопровождается уменьшением  $N$ , и остается постоянным, при тех же условиях, если  $N$  не изменяется. Вследствие весьма малой степени вероятности двух и более ложных переходов  $0 \rightarrow 1$  и такого же количества ложных переходов  $1 \rightarrow 0$  в одной кодовой комбинации за время  $t = a * \Delta t_k$  на практике вероятность  $R_3$  может быть определена приближенным выражением:

$$R_3 \approx 0,9M(N - M) a^2 K_3^2 * 10^{-27}.$$

Таким образом, представление  $p$ -ичных цифр кодом « $M$  из  $N$ » позволяет практически решить проблему автоматического контроля вычислительной системы. В кодовых комбинациях не выделяются контрольные и информационные символы, избыточ-



ная информация распределена равномерно между всеми позициями этих комбинаций, не зависит от вида операции, представляя возможность для охвата контролем всего множества полноразрядных чисел. Поэтому можно считать условие  $P_{n1} = P_{n2} = 1$  (где  $P_{n1}$  и  $P_{n2}$  – вероятности охвата контролем оборудования вычислительной системы, в расчете на обнаружение ошибок первых двух категорий, соответственно) принципиально достижимым.

Рассмотрим возможность выполнения условия  $P_{k1} = P_{k2} = 1$ , где  $P_{k1}$  и  $P_{k2}$  – вероятности безотказной работы органов контроля, рассчитанных на обнаружение первых двух категорий соответственно.

В минимальной дизъюнктивной нормальной форме (ДНФ) выражения, соответствующие признакам  $A_1$  и  $A_2$  ошибочной записи числа (например, операнда  $X$ ) при  $M > 1$  определяют алгоритмы работы органов контроля:

для первой категории ошибок

$$A_{1X} = \bigcap_{(i)} x_{0,i} \cup \bigcap_{(i)} x_{1,i} \cup \dots \cup \bigcap_{(i)} x_{p^*-1,i} = \bigcup_{n=0}^{p^*-1} \bigcap_{(i)} x_{n,i}, \quad (2)$$

где  $x_{n,i}$  – сигнал, соответствующий «единице» в  $i$ -й позиции кодовой комбинации, представляющей цифру  $n$  в коде « $M + 1$  из  $N$ »,

$$\sum_{(i)} x_{n,i} = M + 1; \quad 0 \leq n \leq p^* - 1; \quad p^* \neq p;$$

для второй категории ошибок

$$A_{2X} = \bigcap_{(i)} \bar{x}_{0,i} \cup \bigcap_{(i)} \bar{x}_{1,i} \cup \dots \cup \bigcap_{(i)} \bar{x}_{p^{**}-1,i} = \bigcup_{n=0}^{p^{**}-1} \bigcap_{(i)} \bar{x}_{n,i}, \quad (3)$$

где  $\bar{x}_{n,i}$  – сигнал, соответствующий «нулю» в  $i$ -й позиции кодовой комбинации, представляющей цифру  $n$  в коде « $M - 1$  из  $N$ »,

$$\sum_{(i)} \bar{x}_{n,i} = M + 1; \quad 0 \leq n \leq p^{**} - 1;$$

$$p^{**} \neq p; \quad \text{при } M = N/2, \quad p^* = p^{**}.$$

Отличительной особенностью функционирования устройств контроля, соответствующих выражениям (2) и (3), является включение их элементов только при возникновении ошибок. Вероятность  $P_{\Sigma,l}$  безотказной работы элемента контроля определяется как

$$P_{\Sigma,l} = P_{\Sigma,l}^* (1 - R_{l,1}) + P_{\Sigma,l}^{**} R_{l,1},$$

где  $P_{\Sigma,l}^*$  и  $P_{\Sigma,l}^{**}$  – вероятности безотказной работы элементов контроля, при отсутствии и наличии ошибки кратности  $l$  в контролируемом разряде;  $R_{l,1}$  – вероятность возникновения ошибки кратности  $l$  в данном разряде.

Высокая надежность таких устройств обеспечивается уровнем надежности современной элементной базы и естественным резервированием конъюнктивных элементов в функции от количества ложных цифровых сигналов (при отказах части общей схемы контроля ошибки в кодовых комбинациях, представляющих полноразрядное число, могут быть обнаружены в нормально функционирующих разрядах [3]).

Таким образом, отпадает необходимость в обязательном накоплении  $n$ -разрядных операндов во входных регистрах операционного устройства и последующем их контроле (например, при контроле четности, последовательно выполняются операции свертки и сравнения), который приводит к непроизводительным временным затратам. Появляется возможность организации последовательной (конвейерной) передачи отдельных разрядов обоих операндов (при арифметических операциях) или блоков из  $k$  разрядов, непосредственно в исполнительное устройство соответствующей разрядности



(одноразрядное или  $k$ -разрядное) с последующей фиксацией частичного результата в соответствующих разрядах полноразрядного регистра результата. Заметим, что при использовании, например кода «3 из 6» алфавит может содержать до 20 различных символа, из которых целесообразно выбрать 16 ( $2^4$ ). Поэтому, при фиксированной точности представления чисел в системе, разрядность всех регистров сокращается в 4 раза. Схема, реализующая заявленный принцип, представлена на рис. 3. и включает некоторую память (например, ассоциативную память), множество матричных исполнительных устройств без входных регистров ( $k$ -разрядная логика, где  $k = 1, 2, \dots; k < n$ ).

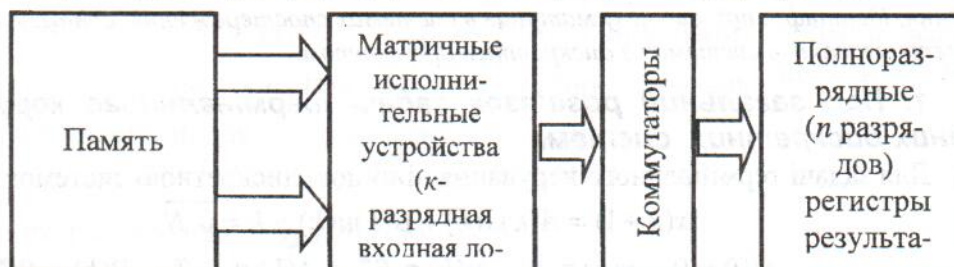


Рис. 3. Упрощенная схема передачи и обработки  $n$ -разрядных операндов блоками по  $k$  разрядов.

Вследствие большого значения  $P_{3,1}$  отказ элемента можно считать практически невозможным событием, поэтому естественная избыточность предоставляет дополнительные возможности для еще большего повышения обнаруживающей способности органов контроля, так как элементы в них не образуют последовательной цепи в вероятностном смысле, Эффект действия таких органов контроля во всех разрядах числа оказывается еще большим, поскольку, достаточно обнаружить ошибку в одном из разрядов числа, для того, чтобы признать контролируемое число ошибочным [3]. Сопоставление значений достоверности, определенной для кодов с различными  $M$ , позволяет сделать вывод о том, что все представители этого семейства кодов обеспечивают возможность получения практически достоверных результатов вычислений.

### Выводы

Анализ свойств кода « $M$  из  $N$ » и структур соответствующих вычислительных устройств показывает, что выполнение в них функции контроля не связано с дополнительными временными задержками и не снижает общую производительность вычислительной системы. Поэтому целесообразно применение кодов « $M$  из  $N$ » в высокопроизводительных вычислительных системах, для повышения производительности процессоров, входящих в ее состав, а также для повышения достоверности результатов при решении критичных задач. Кроме того, при фиксированной производительности вычислительной системы, повышение производительности отдельных процессоров позволит сократить их количество.

### Литература

1. Бурцев В.С. Новые подходы к оценке качества вычислительных средств. // Параллелизм вычислительных процессов и развитие архитектуры суперЭВМ. – М.: Нефть и газ. 1997. – С. 28-40.
2. Бурцев В.С. Система массового параллелизма с автоматическим распределением аппаратных средств суперЭВМ в процессе решения задачи // Юбилейный сборник трудов институтов Отделения информатики, вычислительной техники и автоматизации РАН, - М., 1993. Том II, с. 5-27.
3. Брюхович Е.И. О проблеме автоматического контроля в ЭВМ и контролеспособности позиционных счислений // УСиМ. – 1977. - №4. –С. 71-75.
4. Дадаев Ю.Г. Теория арифметических кодов / –М.: Радио и связь. – 1981. – 274 с.
5. Брюхович Е.И., Гуменюк В.А. Исследование достоверности результатов вычислений при представлении чисел в ЭВМ « $M$  из  $N$ » кодом / Сб. науч. Трудов «Техническая диагностика электронных систем». – К.: Наукова думка. – 1982. –172 с.