

мым становится предвидимым новое событие – окончание обслуживания заявки *B* в этом *OA*. Ссылка на это событие помещается в список будущих событий, и производятся изменения сведений о заявке *B* в массиве заявок. Далее имитируется продвижение в системе заявки *A*. Это продвижение прерывается, когда в соответствии с заданной программой производится запрос на обслуживание заявки *A* в некотором *OA*. При этом происходит обращение к массиву очередей по имени этого *OA* и либо поступление заявки *A* на обслуживание, либо ее постановка в очередь. Изменения, происшедшие в состояниях очередей, обслуживающих аппаратов и заявок, отражаются в изменении соответствующих данных в массиве параметров.

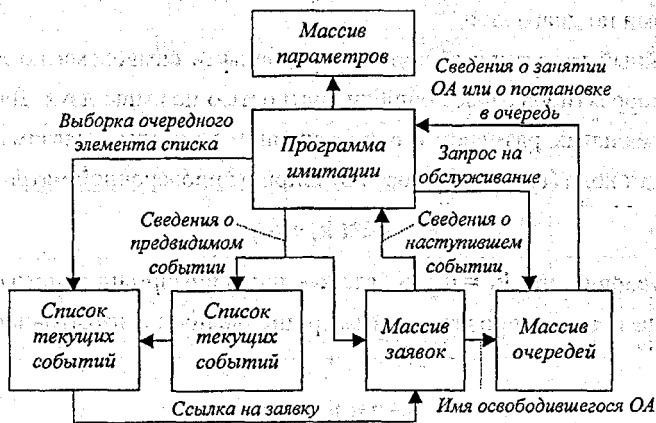


Рис. 1: Взаимодействие программы имитации с массивами БД

Литература.

1. Норенков И.П., Маничев В.Б. Системы автоматического проектирования электронной и вычислительной аппаратуры. – М.: Высшая школа, 1983, 272 с.
2. Северцев Н.А. Надежность сложных систем в эксплуатации и отработке. – М.: Высшая школа, 1989, 432 с.

ОПТИМИЗАЦИЯ ПАРАМЕТРОВ ПРОВЕРОЧНЫХ МАТРИЦ ДЛЯ ЛИНЕЙНОГО И ТРЕХМЕРНОГО ИТЕРАТИВНОГО КОДОВ С ПРОВЕРОЧНЫМИ СИМВОЛАМИ ПО ДИАГОНАЛИ

Д.М. Романенко
(БГТУ, г. Минск)

Наблюдаемое в последние годы резкое увеличение информационных потоков и связанное с этим ужесточение требований к целостности обрабаты-

ваемой информации резко обострили проблему надежного хранения и передачи двоичной информации. Один из способов решения такой проблемы заключается в использовании помехоустойчивого кодирования данных с помощью избыточных кодов.

Одним из этапов как разработки новых, так и совершенствования старых избыточных кодов является определение оптимальных размеров их проверочных матриц. Так как код должен корректировать все ошибки заданной кратности, то основным критерием оптимизации будет выступать величина избыточности, которая должна быть минимальна. В [1] описаны два новых избыточных кода: линейный и трехмерный итеративные коды с проверочными символами по диагонали.

Линейный итеративный код с проверочными символами по диагонали позволяет корректировать все ошибки кратностью не выше двух. Для определения оптимальных размеров его проверочной матрицы выразим величину избыточности кода (r) в зависимости от ширины проверочной матрицы k_1 .

$$r = 2 * (k_1 + k_2). \quad (1)$$

При условии, что $k_1 = k / k_2$, где k – длина информационного слова, k_1 , k_2 – ширина и длина проверочной матрицы, величина избыточности будет равна

$$r = 2 * (k_1 + k / k_1). \quad (2)$$

Оптимальные размеры матрицы будут достигаться при минимуме величины избыточности r . Для определения оптимального k_1 про дифференцируем зависимость $r = f(k_1)$ по k_1 , приравняем полученное выражение к нулю. Решая данное уравнение, получим, что минимум величины избыточности достигается при $k_1 = k / 2$. Следовательно, квадратная матрица является оптимальной с точки зрения избыточности для линейного итеративного кода с проверочными символами по диагонали.

Трехмерный итеративный код с проверочными символами по диагонали, описанный в [1], может использоваться в двух видах: со стандартной и урезанной проверочными матрицами. Для определения оптимальных размеров данных матриц была разработана специальная программа на языке Delphi. Полученные результаты представлены в таблицах 1 и 2 (для стандартной матрицы – табл. 1, для урезанной матрицы – табл. 2).

Таблица 1

Длина информационного слова, k	Размерность матрицы $k_1 \times k_2 \times k_3$	Величина избыточности, r	Относительная избыточность $G_{отн}$, %
8	$2 \times 2 \times 2$	28	350
16	$2 \times 2 \times 4$ $4 \times 2 \times 2$	44	275.0
32	$4 \times 4 \times 2$	64	200.0
64	$4 \times 4 \times 4$	96	150.0
128	$8 \times 4 \times 4$	152	118.7
256	$8 \times 8 \times 4$	224	87.5
512	$8 \times 8 \times 8$	352	68.7
1024	$16 \times 8 \times 8$	560	54.7
4096	$16 \times 16 \times 16$	1344	32.8

Таблица 2

Длина информационного слова, k	Размерность матрицы $k_1 \times k_2 \times k_3$	Величина избыточности, r	Относительная избыточность $G_{отн}$, %
8	$2 \times 2 \times 2$ $4 \times 2 \times 1$	21	262.5
16	$4 \times 4 \times 1$ $4 \times 2 \times 2$	33	206.2
32	$4 \times 4 \times 2$	49	153.1
64	$4 \times 4 \times 4$ $8 \times 4 \times 2$	81	126.6
128	$8 \times 8 \times 2$ $8 \times 4 \times 2$	129	100.8
256	$8 \times 8 \times 4$	193	75.4
512	$8 \times 8 \times 8$ $16 \times 8 \times 4$	321	62.7
1024	$16 \times 8 \times 8$ $16 \times 16 \times 4$	513	50.1
4096	$16 \times 16 \times 16$ $32 \times 16 \times 8$	1281	31.3

Как видно из табл.1 и табл.2, в обоих случаях оптимальной является проверочная матрица, близкая к кубической, либо кубическая (если при заданной длине информационного слова такую матрицу можно построить). При использовании урезанной проверочной матрицы, как правило, существует две оптимальные матрицы. С точки зрения аппаратной реализации и построения

алгоритма коррекции целесообразно использовать проверочную матрицу с $k_1=k_2$, т.е. с квадратной проверочной матрицей для линейного итеративного кода в плоскости.

Литература.

1. Урбанович П.П., Романенко Д.М. Свойства и алгоритмы аппаратной реализации нового вида итеративных кодов для систем памяти // Новые информационные технологии: третья международная конференция NITE'2000, т. 2 – Мн.: БГЭУ, 2000. – с. 159–164.

АЛГОРИТМЫ КОРРЕКЦИИ МНОГОКРАТНЫХ ОШИБОК ТРЕХМЕРНЫМ ИТЕРАТИВНЫМ КОДОМ С ПРОВЕРОЧНЫМИ СИМВОЛАМИ ПО ДИАГОНАЛИ

Д.М. Романенко

(БГТУ, г. Минск)

Поступательное и все ускоряющееся улучшение основных эксплуатационных параметров (быстродействие, емкость ОЗУ – оперативных запоминающих устройств – и др.) персональных компьютеров и других средств вычислительной техники, продиктованное необходимостью внедрения новых информационных технологий, базируется на улучшении адекватных параметров, прежде всего, полупроводниковых устройств хранения и выдачи информации (ОЗУ, ПЗУ, ППЗУ). Принятое за рубежом направление усовершенствования таких ЗУ основывается на создании систем памяти, интегрированных на целой полупроводниковой пластине (WSI – wafer scale integration).

При использовании WSI в качестве ЗУ естественным представляется обобщение итеративных кодов на трехмерный случай. Возможность такого обобщения появляется благодаря тому, что в WSI суммарная емкость системы набирается из однотипных элементов – блоков. Физически такой способ кодирования можно представить в виде куба или параллелепипеда, состоящего из n одинаковых накопителей (отдельных блоков кристалла), "наложенных" друг на друга. Боковые стенки этого куба образуются совокупностью элементов четности n матриц. Верхняя грань является совокупностью элементов четности всех элементов, содержащихся в каждой матрице. В дальнейшем на основе общих характеристик кодов прямого произведения получим основные характеристики кода и рассмотрим его техническую реализацию применительно к WSI.