

4. Родитель добавляется в список свободных узлов, а два его потомка удаляются из этого списка;

5. Одной дуге, выходящей из родителя с большей вероятностью, ставится в соответствие бит 1, другой — бит 0;

6. Шаги, начиная со второго, повторяются до тех пор, пока в списке свободных узлов не останется только один свободный узел. Он и будет считаться корнем дерева;

7. Двигаясь от корня дерева до буквы алфавита можно записать соответствующую ей кодовую комбинацию.

В программе исходная вероятность букв задаётся в виде двумерного массива. При построении эффективного кода используются четыре многомерных массива – массив вспомогательных букв, массив битов, массив имён рёбер, массив путей дерева. По мере заполнения массива вспомогательных букв (шаги 2-6) заполняется массив имен ребер и массив бит. Затем для каждой букве по таблице битов и вспомогательных букв определяется код Хаффмана, а также параллельно строится дерево.

Разработанное приложение успешно используется при проведении лабораторно-практических занятий в Гродненском филиале кафедры «Информационные системы и технологии» Международного института дистанционного образования. Для удобства работы студентов приложения имеет справочную систему, которая содержит теорию построения кода Хаффмана и описание интерфейса.

УДК 621.391

### **Способ сжатия информации в автоматизированных системах.**

Уласюк Н.Н.

Белорусский национальный технический университет

В системах передачи данных от различных информационных систем, территориальная сосредоточенность пунктов сбора информации налагает жесткие технические требования к средствам передачи данных. Особо выделяются системы эксплуатационного контроля с обратными связями. Например, в интегрированной автоматизированной системе технологического контроля (ИАСТК) контроль за работой турбогенераторов на АЭС и их систем возбуждения и охлаждения осуществляется системой виброконтроля и защиты «Лукомль», контроллерами сбора аналоговой и цифровой информации, регистраторами РАС-ИТМ, а также системой централизованного контроля А701-03. Скорость передачи и достоверность информации внутри ЛВС предельно высока и достаточна для принятия оперативных действий. Циклический обмен данными между РСТК и первичными системами сбора аналоговой и дискретной информации осуществляется с

интервалами времени  $T$ , не превышающими 1(2) с. Общая скорость передачи данных в используемых сетях первичных интерфейсов (RS232, RS485...) на порядок меньше скорости в ЛВС и согласуется с помощью специального программного обеспечения.

Для выравнивания скоростных потоков сети первичных интерфейсов и ЛВС предлагается метод повышения пропускной способности каналов низкоскоростной сети до уровня ЛВС. Импульсы передачи данных в каждом канале далее рассматриваются как перекрывающиеся во времени импульсы.

Особенностью использования метода сжатия данных является многократное сжатие исходных данных до минимально возможного значения, которое определяется минимальным количеством узловых точек для интерполятора. Например, для числа узловых точек 3, децимации между ними 64 и разрядности одного отсчета 32 минимальное значение составляет 96 Бит.

С целью снижения временных затрат на многократные процедуры компрессии/декомпрессии указанные схемы можно реализовать на быстродействующих процессорах, созданных на основе перепрограммируемых логических матриц Spartan 6 LX150 FPGA, можно заменить побитный ввод информации на вход компрессора алгоритмом параллельно-последовательного ввода. В результате производительность компрессора увеличивается как минимум в 64 раза.

УДК 004.94:378

### **Построение программного обеспечения на основе непрерывной интеграции**

Дадыкин А.К., Ермолаев А.А.

Белорусский национальный технический университет

Существует множество различных техник, облегчающих разработку и сопровождение программного обеспечения (ПО) в промышленных масштабах: тестирование, система управления версиями, система отслеживания ошибок, автоматизированная система сборки и развертывания и т.д. Непрерывная интеграция (Continuous Integration, CI) объединяет все эти компоненты в единое целое.

Непрерывная интеграция – это практика разработки программного обеспечения, которая заключается в выполнении частых автоматизированных сборок проекта для скорейшего выявления и решения интеграционных проблем.

Термин CI был введен Мартином Фаулером и Кентом Бекон. Данный термин был придуман ими для обозначения практики частой интеграции