

Выводы

Практическая значимость исследования состоит в том, что модель можно использовать при проектировании технологических процессов и операций, а также прогнозировать эксплуатационные параметры изделия с учётом входных, выходных и режимных параметров. При этом на каждой операции при помощи режимов можно прогнозировать и устранять нежелательные и опасные факторы, что не учитывалось в работах [3-6].

Научная новизна работы состоит в управлении технологическими процессами как на уровне операций, так и на уровне общей технологической систем с прогнозированием эксплуатационных показателей, что позволяет существенно экономить время проектирования с использованием ЭВМ.

Список литературы: 1. Белжман Р., Жиджи Э. Динамическое программирование и современная теория управления. М.: МИР, 1984. 207с. 2. Меткин Н.П., Щёголев В.А. Математические основы технологической подготовки гибких производственных систем. М.: Издательство стандартов, 1985. 254с. 3. Невлюдов И.М., Анпилогов Е.М. О моделировании технологического процесса с учётом явлений технологической наследственности // АСУ и приборы автоматики. 1984. Вып. 72. С. 89-93. 4. Дзюндзюк Б.В., Анпилогов Е.М., Анпилогова И.Е., Кравченко Ю.В. Модель управления различными наследуемыми факторами в системе «Человек-Машина» // АСУ и приборы автоматики. 2005. Вып. 132. с. 110-114. 5. Дальский А.М. Технологическое обеспечение надёжности высокоточных деталей машин. М.: Машиностроение, 1975. 225с. 6. Яцерицин П.И., Рыжов Э.В., Аверченков В.И. Технологическая наследственность в машиностроении. Минск: Наука и техника, 1977. 255с.

Поступила в редколлегию 02.08.2007

Анпилогов Евгений Михайлович, канд. техн. наук, доцент кафедры охраны труда ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-13-60.

Анпилогова Ирина Евгеньевна, инженер каф. охраны труда ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-13-60.

Дзюндзюк Борис Васильевич, зав. каф. охраны труда ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-13-60.

Марченко Людмила Ивановна, ст. пр. каф. охраны труда ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-13-60.

Ларченко Лина Викторовна, доцент каф. охраны труда ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-13-60.

УДК 681.586.37:004.5

Н.Я. КАКУРИН, Ю.В. ЛОПУХИН, Н.Н. БЫКОВА

ПРОГРАММНОЕ СРЕДСТВО ДЛЯ АНАЛИЗА ПРЕОБРАЗОВАНИЙ ЧИСЕЛ

Рассматривается структура и возможности программного средства для анализа числа тактов преобразования и статистики преобразований преобразователей кодов, функционирующих по методу накопления эквивалентов

1. Постановка задачи

Основными параметрами преобразователей кодов по методу накопления являются быстродействие (число тактов преобразования) и аппаратные затраты (число корпусов ИС или число вентиляей). К достоинству преобразователей кодов этого типа относятся возможность изменения соотношения между быстродействием и аппаратными затратами за счет выбора числа шагов преобразования, значений шагов и стратегии использования различных шагов преобразования.

Стратегия использования шагов преобразования может быть последовательной или параллельной.

При последовательной стратегии показания разрядных счетчиков, значения которых равны или превышают значение шага, уменьшают на значение этого шага.

Если же во всех преобразуемых разрядах значения цифр оказываются меньше этого шага, происходит переход на более меньший шаг, и далее ведется уменьшение значений

разрядов на значение меньшего шага и т. д., пока не будут обнулены все разрядные счетчики.

Последовательная стратегия по сравнению с параллельной структурно реализуется достаточно просто. Аппаратурные затраты на построение основного блока ПК – формирователь элементов – будут наибольшими.

Для оценки сверху числа тактов преобразования целых чисел с последовательной стратегией в многошаговых ПК выполняют по формулам:

$$\begin{aligned} N_1^{\text{цел}} &= K-1; \\ N_2^{\text{цел}} &= \lceil (K-1)/a \rceil + a - 1; \\ N_3^{\text{цел}} &= \lceil (K-1)/b \rceil + \lceil (b-1)/a \rceil + a - 1; \\ N_4^{\text{цел}} &= \lceil (K-1)/c \rceil + \lceil (c-1)/b \rceil + \lceil (b-1)/a \rceil + a - 1, \end{aligned} \quad (1)$$

где K – основание системы счисления на входе; a, b, c – соответственно второй, третий и четвертый шаги преобразования (первый шаг всегда равен 1); $N_1^{\text{цел}}, N_2^{\text{цел}}, N_3^{\text{цел}}, N_4^{\text{цел}}$ – соответственно максимальное число тактов преобразования одно-, двух-, трех-, четырехшагового ПК целых чисел.

Для значений шагов преобразования должны выполняться следующие ограничения:

$$1 < a < b; a < b < c; b < c \leq K-1.$$

Недостатком использования максимальных значений $N_i^{\text{цел}}$ ($i = \overline{1,4}$) для оценки быстродействия являются завышенные требования, которые для определенных подмножеств преобразуемых чисел будут значительны и потребуют применения в структуре ПК более быстродействующих ИС.

В целях реальной оценки быстродействия разработан программный пакет "CONVERTER", позволяющий промоделировать процесс преобразования чисел для последовательной стратегии и определить конкретное число тактов преобразования, а не оценку сверху.

2. Назначение и структура программного средства

К назначениям программного средства относятся: возможность имитации работы преобразователя кодов по методу накопления эквивалентов;

возможность работы в двух режимах: режиме преобразования целых и дробных чисел, режиме статистики преобразования целых и дробных чисел.

При разработке данного программного продукта было выбрана наиболее оптимальная по скорости проектирования и эффективности визуальная среда программирования Borland Delphi 7.0 для Windows 2000/XP.

Для нормальной работы данной программы необходимы следующие технические средства: IBM-совместимый компьютер с процессором не менее Pentium 233 МГц, монитор типа VGA, SVGA, ОЗУ 16 Мб, мышь, дисковод FDD (для установки программы), HDD (необходимая емкость свободного пространства определяется исходя из сложности решаемой задачи, но составляет не менее 1,5 Мб). Используемыми программными средствами являются: ОС Windows 95 и выше. Для более корректной работы программы рекомендуется ОС Windows 2000.

Разработанное программное обеспечение сформировано как независимый модуль и для своего функционирования не требует дополнительного программного обеспечения.

3. Функционирование программы в режиме преобразования чисел

На рис. 1 представлена структурная схема алгоритма режима преобразования чисел.

Этот режим используется для преобразования числа из K -ичной системы счисления в двоичную. Программа выдает не только окончательный результат преобразования, но и промежуточные результаты в виде таблицы: состояние счетчика, величину эквивалента и состояние накапливающего сумматора на каждом такте преобразования. Имеется возможность задавать различные число шагов преобразования от 1 до 8, основание системы счисления на входе преобразователя – от 3 до 15, разрядность числа – от 2 до 12, веса шагов преобразователя.

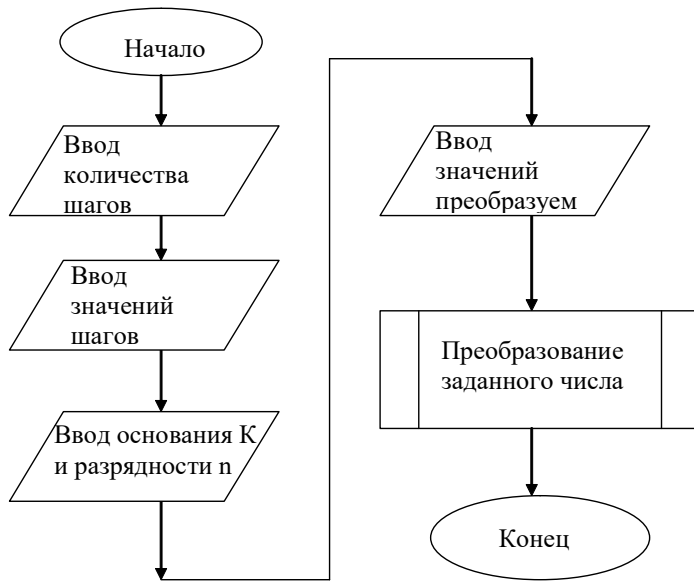


Рис. 1

Для того чтобы произвести расчет в данном режиме, необходимо запустить программу путем активизации исполняемого модуля Converter.exe.

После этого следует с помощью мышки активизировать главное меню программы или нажать клавишу Alt на клавиатуре и перейти к пункту Режим. В раскрываемом меню следует выбрать пункт Режим преобразования. Для запуска данного режима работы программы имеется специальная горячая клавиша – F1. Нажав ее сразу после запуска программы, можно без перемещения по пунктам меню прийти к рассмотренному выше результату. После данных манипуляций на экране отобразится окно для

ввода исходных данных (рис. 2).

В этом окне имеются пять текстовых полей для ввода данных, три из которых снабжены кнопками инкремента. С помощью этих элементов управления необходимо задать исходные данные для режима преобразования чисел: количество шагов преобразователя, основание системы счисления на входе, разрядность преобразуемого числа, собственно само число, которое нужно преобразовать, и значения весов преобразователя. Программа содержит ряд проверок, которые предотвращают ввод некорректных данных. Если же пользователь по каким-либо причинам желает отказаться от расчета, то следует нажать кнопку Отмена. После этого программа, не сохраняя данные, вернется в исходное состояние.

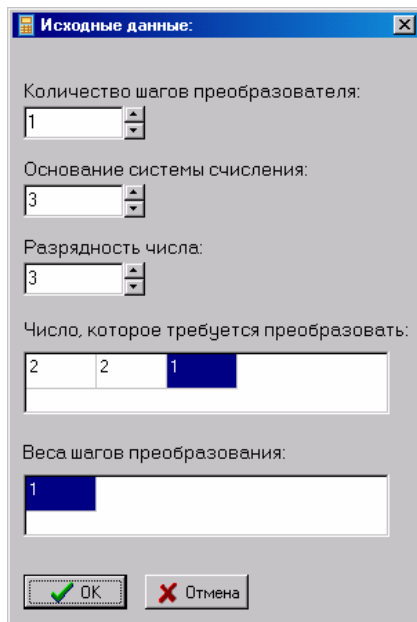


Рис. 2

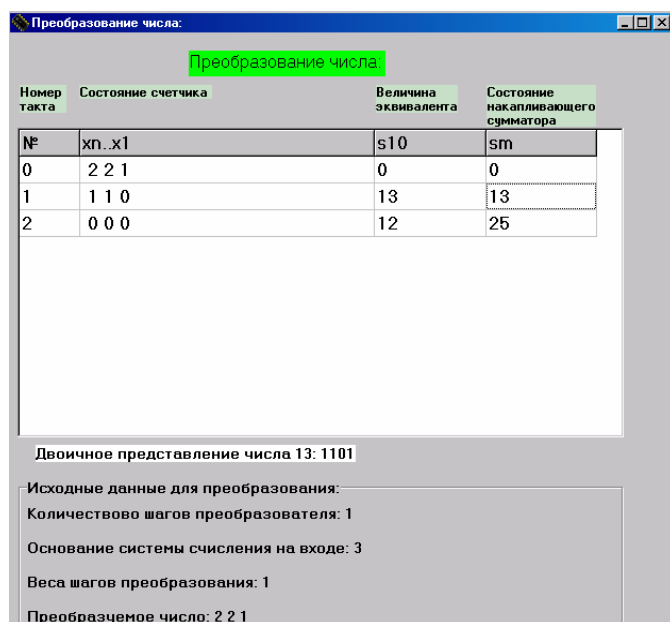


Рис. 3

Когда все значения будут введены в окно, следует нажать кнопку ОК для запуска преобразователя. После того, как программа произведет расчет, на экране высветится

окошко с результатами преобразования введенного числа (рис. 3). Данное окно выполнено с использованием современного многодокументного подхода к проектированию приложений на платформе Windows и является дочерним по отношению к главному окну программы. Это дает немалые преимущества для пользователя – имеется возможность сразу одновременно видеть на экране несколько результатов расчета в разных режимах работы приложения, сравнивать один с другим, упорядочивать окна, каскадировать их и т.д.

Программа отображает значения эквивалентов и состояния накапливающего сумматора в десятичном виде. Для того чтобы увидеть двоичные эквиваленты, необходимо щелкнуть левой кнопкой мыши на интересующем нас значении накапливающего сумматора, после чего в окошке с результатами отобразится двоичное представление данного числа. Тот же самый результат можно получить, если перемещаться по таблице клавишами клавиатуры с обозначениями стрелок.

Для того чтобы закрыть данное окошко, необходимо нажать кнопку с обозначением крестика.

4. Функционирование программы в режиме статистики

Режим статистики служит для отображения результатов преобразования заданного множества чисел в виде графической зависимости, которая позволяет определить количество преобразованных чисел за определенное число тактов преобразования. Программа отображает результаты в виде гистограммы. Кроме того, предусмотрена возможность получения статистики в виде таблицы.

Математическое ожидание рассчитывается по формуле: $M = (\sum_{i=0}^{i=N_{\max}} i \cdot N_i) / 2^n$, где M —

математическое ожидание числа тактов преобразования i ; N_i — количество чисел, преобразуемых за i тактов; N_{\max} — максимальное значение числа тактов преобразования (зависит от основания системы счисления K и значений шагов преобразования a, b, c); n — число входных преобразуемых разрядов.

Затем рассчитывается дисперсия преобразования по формуле:

$$D = (\sum_{j=1}^{j=2^n} (M - N_j)^2) / 2^n \cdot (2^{n-1}),$$

N_j – количество тактов, за которое преобразуется число a_j .

Результат записывается в переменную D . Для вывода на экран значений переменных a и D используется функция `FormatFloat`, которая форматирует их до двух знаков после запятой.

Имеется возможность задавать различные число шагов преобразования - от 1 до 8, основание системы счисления на входе преобразователя – от 3 до 15, разрядность чисел – от 2 до 12, веса шагов преобразователя. Кроме этого, данный режим позволяет вручную задавать набор преобразуемых чисел или автоматически на выбор. При автоматическом формировании чисел программа делает полный перебор всех возможных значений в диапазоне от 0 до $n-1$, где n - разрядность преобразуемых чисел.

Данный режим запускается как и предыдущий путем выбора пункта Режим статистики из меню Режим или нажатием горячей клавиши F2 на клавиатуре.

Сначала выбирается максимальный шаг преобразования и соответственно ему присваивается значение ноль. В переменную `step` заносится значение максимального шага из массива значений шагов. Переменная `equiv` хранит состояние накапливающего сумматора. Она инициализируется нулем на начальном этапе преобразования. После этого инициализируется нулевая строка таблицы. Первая, третья и четвертая строка заполняется нулями, так как на нулевом такте преобразования все счетчики обнуляются, а во вторую заносится исходное число.

Далее в цикле `while` выполняется преобразование числа:

```

while(true) do
begin
// подбираем максимально возможный шаг преобразования
if w_index<=l_kol_steps-2 then
begin
while(true) do
begin
for i:=0 to l_n-1 do
if l_counter[i]-step>=0 then goto l1;
w_index:=w_index+1;
step:=l_weights[w_index];
end;
end;
l1:
sum:=0;
j:=l_n-1;
for i:=0 to l_n-1 do
begin
if l_counter[i]-step>=0 then
begin
l_counter[i]:=l_counter[i]-step;
sum:=sum+step*power(l_k,j);
end;
end;
j:=j-1;
end;
equiv:=equiv+sum;
Child.StringGrid1.RowCount:=Child.StringGrid1.RowCount+1;
Child.StringGrid1.Cells[0,number+1]:=IntToStr(number);
for i:=0 to l_n-1 do
Child.StringGrid1.Cells[1,number+1]:=Child.StringGrid1.Cells[1,number+1]+' '+IntToStr(l_counter[i]);
Child.StringGrid1.Cells[2,number+1]:=FloatToStr(sum);
Child.StringGrid1.Cells[3,number+1]:=FloatToStr(equiv);
number:=number+1;
// условие завершения цикла
for i:=0 to l_n-1 do
if l_counter[i]>0 then goto l2;
break;
l2:
end.
end.

```

Структурная схема алгоритма для режима статистики представлена на рис. 4.

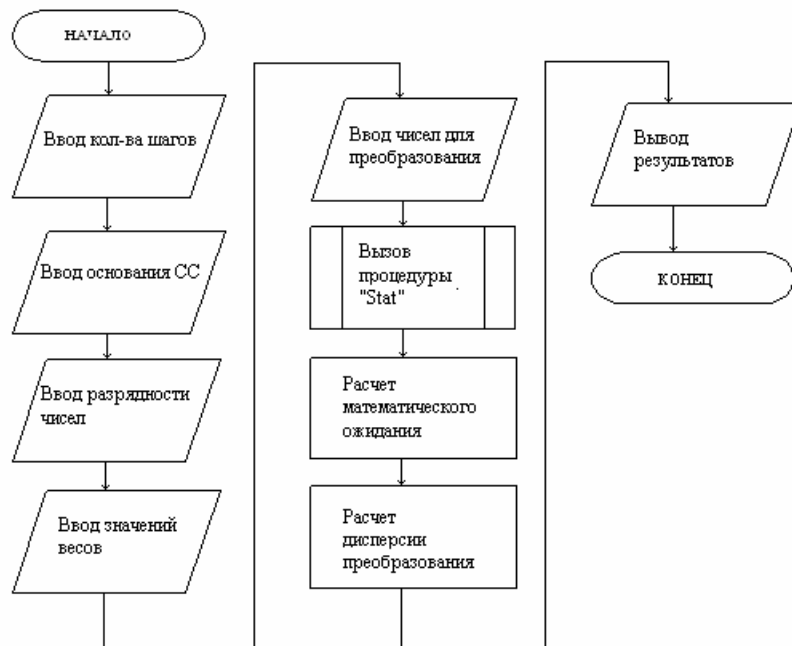


Рис. 4

После этого на экране монитора появится окно для ввода исходных данных в режиме статистики (рис.5).

В этом окне имеются четыре текстовых поля для ввода данных, три из которых снабжены кнопками инкремента. С помощью этих элементов управления необходимо задать исходные данные для режима статистики: количество шагов преобразователя, основание системы счисления на входе, разрядность чисел и значения весов преобразователя. Программа содержит ряд проверок, которые предотвращают ввод некорректных данных. Если же пользователь по каким-либо причинам желает отказаться от расчета, то следует нажать кнопку Отмена, после чего программа, не сохраняя данные, вернется в исходное состояние.

После того, как пользователь введет все необходимые данные на этом этапе работы программы, следует нажать кнопку ОК. После этого приложение выведет на экран еще одно окошко, в котором пользователь сможет задать необходимый список чисел для преобразования (рис. 6).

После ввода всех данных программа в режиме статистики осуществляет полный перебор всех чисел и вывод результатов расчета как в виде гистограммы, так и в виде чисел в специальных окнах. Например, для $K=10$; $n=3$ и одного шага преобразования получены следующие расчетные данные количества чисел N_i за i тактов: за 0 тактов – 1 число; за 1 – 7 чисел; за 2 – 19; за 3 – 37 чисел; за 4 – 61 число; за 5 – 91 число; за 6 – 127 чисел; за 7 – 169 чисел; за 8 – 217 чисел; за 9 – 271 число. Математическое ожидание числа тактов преобразования $M=6,97$, а дисперсия $D=3,71$.

В этом окошке (см. рис.6) внизу имеется таблица для ввода чисел. Чтобы добавить число в список, пользователь должен ввести его в данную таблицу и нажать кнопку с изображением знака "+". После этого введенное значение появится в верхней таблице, которая содержит все заданные числа на текущий момент. Если возникнет необходимость удалить последнее значение из списка, то необходимо нажать кнопку с изображением знака "-".

Если нет необходимости проводить полную статистику преобразования по всему диапазону чисел, то следует деактивизировать опцию «Полный перебор всех комбинаций».

Когда все значения будут введены в окно, следует нажать кнопку ОК для запуска преобразователя. После того, как программа произведет расчет, на экране высветится окошко с результатами статистики преобразования чисел (рис.7, а, б).

В этом окошке появится график в виде гистограммы, который и отображает зависимость количества преобразованных чисел от числа тактов преобразования.

Выводы

Научная новизна выполненного исследования заключается в следующем:

1. Разработано программное обеспечение для получения достоверных оценок числа тактов преобразований как для ограниченного подмножества заданных чисел, так и в случае полного перебора всех чисел заданной разрядности.

2. Рассмотрена структура программного пакета и его функционирование в двух основных режимах: преобразование числа и получение статистических оценок результатов преобразования.

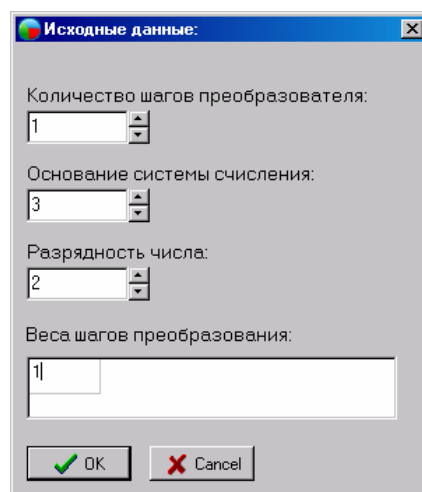


Рис. 5

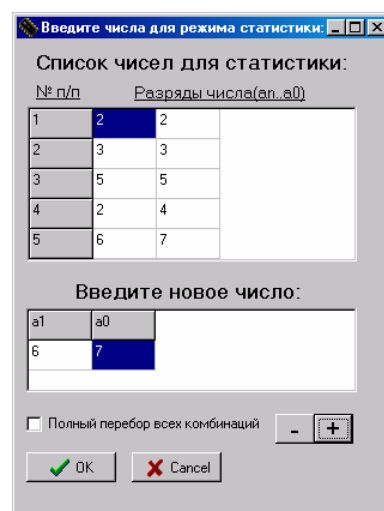


Рис. 6



а



б

Рис. 7

3. Результаты апробации программного пакета показали, что требования по быстродействию к преобразователям кодов могут быть существенно снижены (на 25-30%).

Практическая значимость исследования состоит в возможности использования программного пакета для проектирования преобразователей кодов с улучшенными характеристиками.

Список литературы: 1. А.с. 1126946 5G06F 5/02. Преобразователь двоично-К-ичного кода в двоичный код /А.Н. Слобожанин // Открытия, изобретения. 1984. №44. С.250. 2. А.с. 1647908 5H03M 7/12. Преобразователь двоично-К-ичного кода в двоичный код /Н.Я.Какурин, Ю.К. Кирияков, А.Н. Макаренко // Там же. 1991. №17. С. 262-263. 3. Какурин Н.Я. Выбор величин шага преобразования в преобразователях код-код /АСУ и приборы автоматики. 1991. Вып.97. С. 14-17. 4. Голян В.В., Какурин Н.Я., Макаренко А.Н., Замалеев Ю.С., Николаев А.А. Двухкритериальный системный синтез многоблочных преобразователей кодов по методу накопления эквивалентов // АСУ и приборы автоматики. 2005. Вып. 133. С.102-107.

Поступила в редколлегию 17.09.2007

Какурин Николай Яковлевич, канд. техн. наук, профессор кафедры автоматизации проектирования вычислительной техники ХНУРЭ. Научные интересы: прикладная теория цифровых автоматов, автоматизация проектирования цифровых устройств. Адрес: Украина, 61166, Харьков, пр.Ленина, 14, тел. 70-21-326.

Лопухин Юрий Владимирович, ст. преподаватель кафедры АПВТ ХНУРЭ. Научные интересы: проектирование программного обеспечения, автоматизации проектирования цифровых устройств. Адрес: Украина, 61166, Харьков, пр.Ленина, 14, тел. 70-21-326.

Быкова Надежда Николаевна, студентка группы КСМ 04-2 факультета компьютерной инженерии и управления ХНУРЭ. Научные интересы: прикладная теория цифровых автоматов, автоматизация проектирования цифровых устройств. Адрес: Украина, 61166, Харьков, пр.Ленина, 14, тел. 70-21-326.