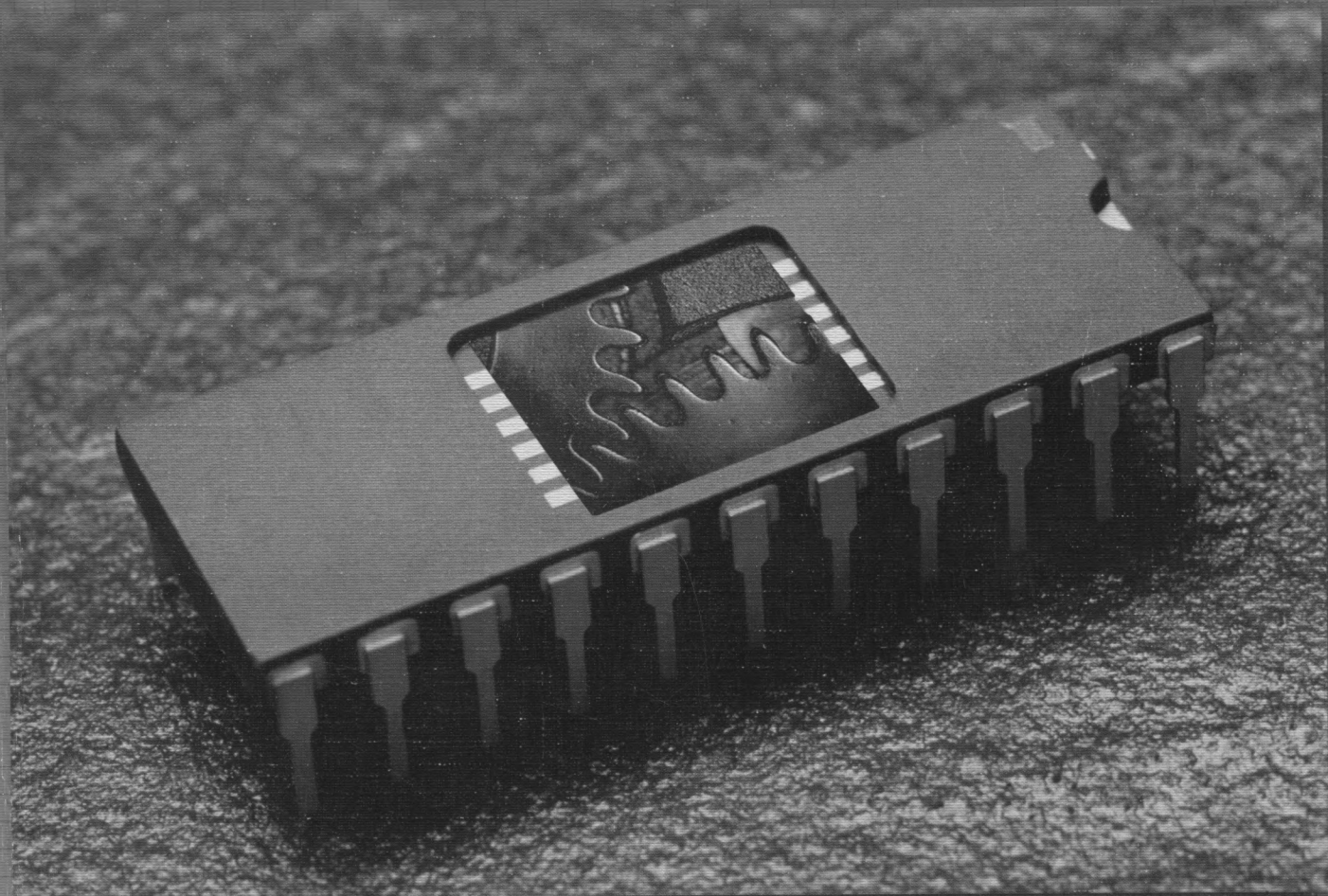


Министерство образования и науки Украины

АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ И ПРИБОРЫ АВТОМАТИКИ



'2002
119

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ

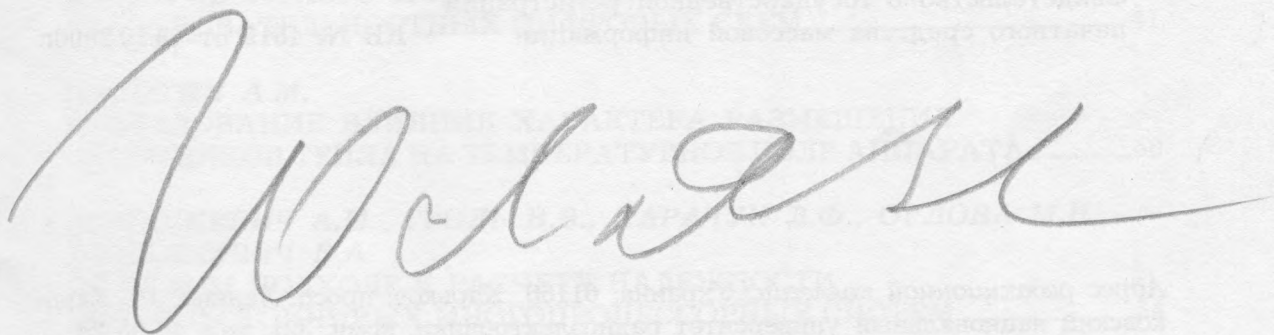
ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ
УНИВЕРСИТЕТ РАДИОЭЛЕКТРОНИКИ

АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ И ПРИБОРЫ АВТОМАТИКИ

Всеукраинский межведомственный
научно-технический сборник

Основан в 1965 г.

Выпуск 119

A large, stylized handwritten signature in black ink, likely belonging to the editor or publisher, is written across the middle of the page.

Харьков
2002

СОДЕРЖАНИЕ

ВОРОБЬЕВА В.П., КРУГОЛЬ М.С., МАРТЫНОВ С.А., УСКОВ В.В., ХАЖМУРАДОВ М.А. МОДЕЛИРОВАНИЕ ПОВЕРХНОСТЕЙ СЛОЖНОЙ ПРОСТРАНСТВЕННОЙ ФОРМЫ СТЕЛЛАТОРНЫХ МАГНИТНЫХ ЛОВУШЕК.....	5
УДОВИЦКИЙ В.Г. НАПІВПРОВІДНИКОВІ СЕНСОРИ ХЛОРУ ДЛЯ АВТОМАТИЗОВАНИХ СИСТЕМ ЕКОЛОГІЧНОГО МОНІТОРИНГУ.....	10
ЛАПТА С.С. УЧЕТ ИНСУЛИННЕЗАВИСИМОЙ УТИЛИЗАЦИИ ГЛЮКОЗЫ В МАТЕМАТИЧЕСКОЙ МОДЕЛИ ВНУТРИВЕННОГО ТЕСТА ТОЛЕРАНТНОСТИ К ГЛЮКОЗЕ.....	13
СЛИПЧЕНКО Е.В. ИССЛЕДОВАНИЕ ВЛИЯНИЯ ЭКОНОМИЧЕСКИХ ФАКТОРОВ НА РЕНТАБЕЛЬНОСТЬ ДЕЯТЕЛЬНОСТИ БИЗНЕС-СТРУКТУР.....	16
АГАПОВА И.С. ПОСТРОЕНИЕ ОЦЕНКИ ИНТЕНСИВНОСТИ ВХОДЯЩЕГО ПОТОКА ЗАЯВОК ДЛЯ ОПРЕДЕЛЕННЫХ МНОГОЛИНЕЙНЫХ СИСТЕМ МАССОВОГО ОБСЛУЖИВАНИЯ.....	21
ЕВЛАНОВ М.В., АБУ САРХАН ЗАХИ А.М. КОНЦЕПЦИЯ МОДЕЛИРОВАНИЯ ПРОЦЕССОВ РЕИНЖИНИРИНГА КОРПОРАТИВНОЙ ИНФОРМАЦИОННОЙ СИСТЕМЫ.....	27
ГОЛЯН В.В., КАКУРИН Н.Я., МАКАРЕНКО А.Н. ПРОГРАММНЫЕ СРЕДСТВА ПОСТРОЕНИЯ ТАБЛИЦ ФУНКЦИОНИРОВАНИЯ ФОРМИРОВАТЕЛЕЙ ЭКВИВАЛЕНТОВ ДЛЯ МНОГОВЛОЧНЫХ ПРЕОБРАЗОВАТЕЛЕЙ КОДОВ ДРОБНЫХ ЧИСЕЛ.....	32
СЫСЕНКО И.Ю. МЕТОД ОБРАТНОГО МОДЕЛИРОВАНИЯ НЕИСПРАВНОСТЕЙ ПОСЛЕДОВАТЕЛЬНОСТНЫХ ЦИФРОВЫХ СХЕМ.....	41
СИНОТИН А.М. ИССЛЕДОВАНИЕ ВЛИЯНИЯ ХАРАКТЕРА РАЗМЕЩЕНИЯ ИСТОЧНИКОВ ТЕПЛА НА ТЕМПЕРАТУРНОЕ ПОЛЕ АППАРАТА.....	50
РОМАНКЕВИЧ А.М., ГРОЛЬ В.В., КАРАЧУН Л.Ф., ОРЛОВА М.Н., РОМАНКЕВИЧ В.А. ОБ ОДНОМ ПОДХОДЕ К РАСЧЕТУ НАДЕЖНОСТИ ОТКАЗОУСТОЙЧИВЫХ МНОГОПРОЦЕССОРНЫХ СИСТЕМ.....	54
ЗИНЧЕНКО Ю.Е., МИРОШНИКОВ А.С., РЫТОВ А.С., МУХАММАД ЯСИР КАДДАФИ ПРОЕКТИРОВАНИЕ ДИАГНОСТИЧЕСКОГО ОБЕСПЕЧЕНИЯ НА БАЗЕ FPGA-, HDL- И NES-ТЕХНОЛОГИЙ.....	59

ГРЕБЕННИК И.В., РОМАНОВА Т.Е. УЧЕТ ПОГРЕШНОСТЕЙ ПРИ ПОСТРОЕНИИ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ОПТИМИЗАЦИОННЫХ КОМБИНАТОРНЫХ ЗАДАЧ.....	64
КРАСНОПОЯСОВСКИЙ А.С. ОПТИМІЗАЦІЯ КОНТЕЙНЕРІВ КЛАСІВ РОЗПІЗНАВАННЯ ЗА МЕТОДОМ ФУНКЦІОНАЛЬНО-СТАТИСТИЧНИХ ВИПРОБУВАНЬ.....	69
ЛЕБЕДЕВ О.Г., ДУКИН Г.Ю., БАРБУЛАТ В.А. УНИФИЦИРОВАННАЯ МОДЕЛЬ ИСПОЛЬЗОВАНИЯ РАДИОЛОКАЦИОННОЙ ТЕХНИКИ В ОБЪЕДИНЕННОЙ СИСТЕМЕ ОРГАНИЗАЦИИ ВОЗДУШНОГО ДВИЖЕНИЯ.....	76
КОЛЯДИН В.Л. ОБ ОДНОМ ЗАБЛУЖДЕНИИ ОТНОСИТЕЛЬНО НЕСТАЦИОНАРНЫХ СЛУЧАЙНЫХ ПРОЦЕССОВ.....	84
ФЕДОРОВ Е.Е. СОЗДАНИЕ БЕСКОНФЛИКТНОЙ СИСТЕМЫ УПРАВЛЕНИЯ УСТРОЙСТВОМ ВВОДА-ВЫВОДА ЗВУКА.....	96
МАТОРИН В.С., МАТОРИН С.И., ПОЛУНИН Р.А., ПОПОВ А.С. UFO-ТЕХНОЛОГИЯ: ОБЪЕКТНО-ОРИЕНТИРОВАННЫЙ АНАЛИЗ И ПРОЕКТИРОВАНИЕ CASE-ИНСТРУМЕНТАРИЯ.....	101
ДИЧКА І.А., СУЛЕМА Є.С., ГОЛУБ В.І. ЗАСТОСУВАННЯ ШТРИХОВОГО КОДУВАННЯ ІНФОРМАЦІЇ В АВТОМАТИЗОВАНИХ СИСТЕМАХ УПРАВЛІННЯ.....	108
ТУРКОВСКИЙ О.С., ТОЛСТОХАТЬКО В.А., СИМОНОВ С.І. ВИКОРИСТАННЯ ГЕНЕТИЧНИХ АЛГОРИТМІВ ДЛЯ РОЗПОДІЛУ НАВЧАЛЬНОГО НАВАНТАЖЕННЯ ВИКЛАДАЧІВ КАФЕДРИ.....	115
РЕФЕРАТИ	120

ПРОГРАММНЫЕ СРЕДСТВА ПОСТРОЕНИЯ ТАБЛИЦ ФУНКЦИОНИРОВАНИЯ ФОРМИРОВАТЕЛЕЙ ЭКВИВАЛЕНТОВ ДЛЯ МНОГОБЛОЧНЫХ ПРЕОБРАЗОВАТЕЛЕЙ КОДОВ ДРОБНЫХ ЧИСЕЛ

Рассматривается принцип построения программы автоматизированного проектирования формирователей эквивалентов для многоблочных преобразователей кодов параллельного типа.

1. Постановка задачи

Основой развития современной науки, техники и в целом общества является бурный прогресс электронной вычислительной техники, широко используемой во всех областях научной, производственной и хозяйственной деятельности. Сегодня трудно себе представить деятельность человека без ЭВМ. Появившись около 50 лет назад, ЭВМ открыли новую страницу в истории человеческих знаний и возможностей, высвободили тысячи вычислителей, значительно облегчили труд ученых, дали возможность изучить сложнейшие процессы. Сейчас нет ни одной отрасли народного хозяйства, где нельзя было бы применить ЭВМ; более того, целые разделы науки и техники не смогут существовать без них.

Существует большое разнообразие способов и методов преобразования чисел. Одним из наиболее важных является аппаратный способ преобразования. Важнейшей областью применения преобразователей кодов (ПК) является преобразование двоичного кода в двоично-К-ичный и двоично-К-ичного в двоичный код, что используется в устройствах ЭВМ, так как половина всех ЭВМ работает в двоичной системе счисления, но исходные данные и результаты решения задач должны быть представлены в форме, удобной для их восприятия.

Актуальной задачей является разработка средств автоматизированного проектирования преобразователей, входящих в состав мультиплексорных и селекторных каналов ЭВМ и обеспечивающих наряду с другими блоками выполнение необходимых преобразований кодов при их передаче из периферийного устройства в канал и обратно. Одной из трудоемких задач на этапе проектирования ПК по методу накопления эквивалентов является построение таблиц функционирования формирователей эквивалентов (ФЭ) для многоразрядных ПК.

Программа FE_DROB, рассмотренная в этой работе, решает задачу разбиения многоразрядного ПК на заданное число параллельно функционирующих блоков (с небольшой разрядностью — 1...15), что дает возможность во много раз уменьшить аппаратные затраты и сократить сроки проектирования.

2. Алгоритм функционирования преобразователей кодов на базе счетчиков

При повышенных требованиях к быстродействию следует выбрать для разработки и проектирования трехшаговый ПК (рис.1). Он имеет группу разрядных счетчиков 1, генератор импульсов (блок управления) 2, содержащий прямой П, прямые задержанные ПЗ1 и ПЗ2 выходы, первую группу триггеров 3 состояния, вторую группу триггеров 4 состояния, третью группу триггеров 5 состояния, комбинационный двоичный сумматор 6 и регистр результата 7, образующие в совокупности накапливающий сумматор 8, пер-

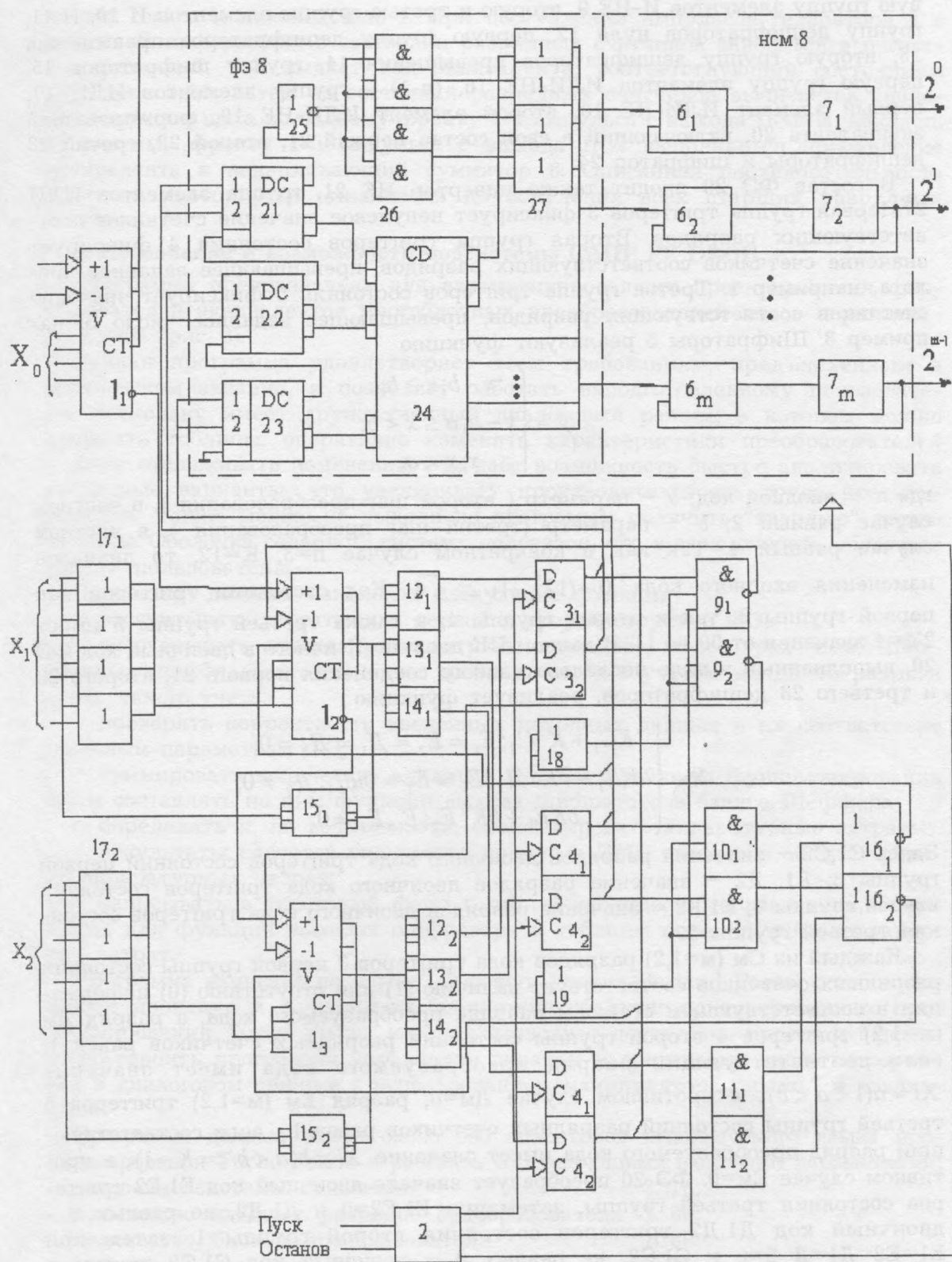


Рис. 1. Структурная схема трехшагового ПК

вую группу элементов И-НЕ 9, вторую и третью группы элементов И 10, И 11, группу дешифраторов нуля 12, первую группу дешифраторов превышения 13, вторую группу дешифраторов превышения 14, группу шифраторов 15, первую группу элементов ИЛИ-НЕ 16, (n-1) - группу элементов ИЛИ 17, первый элемент ИЛИ-НЕ 18, второй элемент ИЛИ-НЕ 19, формирователь эквивалента 20, включающий в свой состав первый 21, второй 22, третий 23 дешифраторы и шифратор 24.

В состав ФЭ 20 входит также инвертор НЕ 21, группа элементов ИЛИ 27. Первая группа триггеров 3 фиксирует ненулевое значение счетчиков соответствующих разрядов. Вторая группа триггеров состояния 4 фиксирует значение счетчиков соответствующих разрядов, превышающее заданное число а, например 1. Третья группа триггеров состояния 5 фиксирует значение счетчиков соответствующих разрядов, превышающее заданное число b, например 3. Шифраторы 5 реализуют функцию

$$y = \begin{cases} x - b; x \geq b; \\ x - a; a \leq x < b; \\ x; x < a, \end{cases}$$

где x - входной код; a - параметр (второй шаг преобразования), в частном случае равный 2; b - параметр (третий шаг преобразования), в частном случае равный 4. Так как в конкретном случае $n=3$; $K=12$, то диапазон изменения входного кода $0 - (12^3 - 1) \cdot 2^2 = 4$. Код состояния триггеров как первой группы 3, так и второй группы 4, а также третьей группы 5 имеет $2^2=4$ значения от 00 до 11. В данном ПК двоично-К-ичного в двоичный код ФЭ 20, выполненный в виде последовательного соединения первого 21, второго 22 и третьего 23 дешифраторов, реализует функцию

$$S = \begin{cases} KC_1 + K^2 C_2; E_1 = E_2 = D_1 = D_2 = 0; \\ aKD_1 + aK^2 D_2; E_1 = E_2 = 0; D_1, D_2 \neq 0; \\ bKE_1 + bK^2 E_2; E_1, E_2 \neq 0. \end{cases}$$

Здесь C_1, C_2 - значения разрядов двоичного кода триггеров состояний первой группы 3; D_1, D_2 - значение разрядов двоичного кода триггеров состояния второй группы 4; E_1, E_2 - значение разрядов двоичного кода триггеров состояния третьей группы 5.

Каждый из S_m ($m=1,2$) разрядов кода триггеров 3 первой группы состояний разрядных счетчиков соответствует наличию (1) или отсутствию (0) информации в соответствующем старшем разряде преобразуемого кода, а разряд D_m ($m=1,2$) триггеров 4 второй группы состояний разрядных счетчиков равен 1, если соответствующий разряд преобразуемого кода имеет значение $X_i = a$ ($1 < a < b$), в противном случае $D_m = 0$; разряд E_m ($m=1,2$) триггеров 5 третьей группы состояний разрядных счетчиков равен 1, если соответствующий разряд преобразуемого кода имеет значение $X_i = b$ ($a < b \leq K-1$), в противном случае $E_m = 0$. ФЭ 20 преобразует вначале двоичный код E_1, E_2 триггеров состояния третьей группы, затем при $E_1, E_2 = 0$ и D_1, D_2 , не равных 0 - двоичный код D_1, D_2 триггеров состояния второй группы 1, затем при $E_1 = E_2 = D_1 = D_2 = 0$ и C_1, C_2 , не равных 0 - двоичный код C_1, C_2 триггеров состояния первой группы и при $E_1 = E_2 = D_1 = D_2 = C_1 = C_2 = 0$ выполняет трансляцию двоично-К-ичного кода младшего разряда в двоичный код эквивалента на выходе.

Преобразование числа выполняется по принципу последовательного использования шагов преобразования: вначале третий шаг шифратора b, затем

второй шаг а и наконец шаг 1. При поступлении импульсов генератора 2 с выхода П информация о состоянии разрядных счетчиков заносится в регистры состояний, ФЭ выдает код эквивалента, соответствующий состояниям регистров 3,4,5, затем из состояний разрядных счетчиков вычитается значение текущего шага по заднему фронту 1-0 импульса с выхода ПЗ1. И, наконец, по заднему фронту 1-0 импульса с выхода ПЗ2 выполняется прибавление эквивалента в накапливающий сумматор 8. Описанная последовательность действий повторяется циклически до обнуления всех старших разрядных счетчиков.

3. Назначение и возможности подсистемы САПР FE DROB

Программа предназначена для автоматизированного проектирования формирователей эквивалентов многоблочных преобразователей кодов параллельного типа (рис. 2).

Данная программа удовлетворяет всем требованиям, предъявленным в техническом задании, и позволяет работать неподготовленному пользователю, поскольку имеет дружественный диалоговый режим, в котором можно создавать таблицы, оперативно изменять характеристики преобразователей и сразу отслеживать изменения. Это даёт возможность быстро анализировать различные варианты, что увеличивает производительность труда проектировщика. Также можно не выходя из программы получить "твердую" копию таблицы. Оболочка содержит систему подсказок, что в свою очередь облегчает работу пользователя.

Программа должна выполнять следующие функции:

- рассчитывать затраты для первой и второй структур ФЭ;
- рассчитывать аппаратурные затраты для одно-, двух-, трех- и четырехшаговых преобразователей кодов дробных чисел с учетом младшего разряда и без такого учета;
- проверять корректность введенных исходных данных и их соответствие заданным параметрам ФЭ;
- суммировать количество единиц в столбцах таблицы функционирования ФЭ и составлять по ним функции выхода шифратора в базисе Шеффера;
- определять и, по возможности, оптимизировать аппаратурные затраты;
- результаты расчетов количества корпусов ИМС сводить в общую таблицу аппаратурных затрат;
- записывать в текстовый файл result.txt таблицы преобразования, выражения для функций выходов шифратора и таблицы определения аппаратурных затрат.

Программа написана на языке высокого уровня Borland Pascal с использованием процедур и функций из библиотеки объектно-ориентированного программирования Turbo Vision, которая позволяет достаточно просто и эффективно строить программы управления экраном и средствами ввода информации в диалоговом режиме с использованием манипулятора "мышь" и клавиатуры.

Программа должна иметь около 500 операторов используемого языка программирования Turbo Pascal и работать со следующими входными параметрами:

основание системы счисления	3...15;
количество входных разрядов преобразователя	1...80;
количество блоков преобразователя	1...80;
количество входных разрядов блока	1...15;
количество шагов преобразования	
(но не более значения основания счисления)	1...20;
количество выходных разрядов	1...250.

Для нормальной работы данной программы необходимы следующие технические средства : IBM-совместимый компьютер не менее 80x286, монитор типа CGA, EGA, VGA, SVGA, RAM 1 Mb, мышшь, дисковод HDD (необходимая

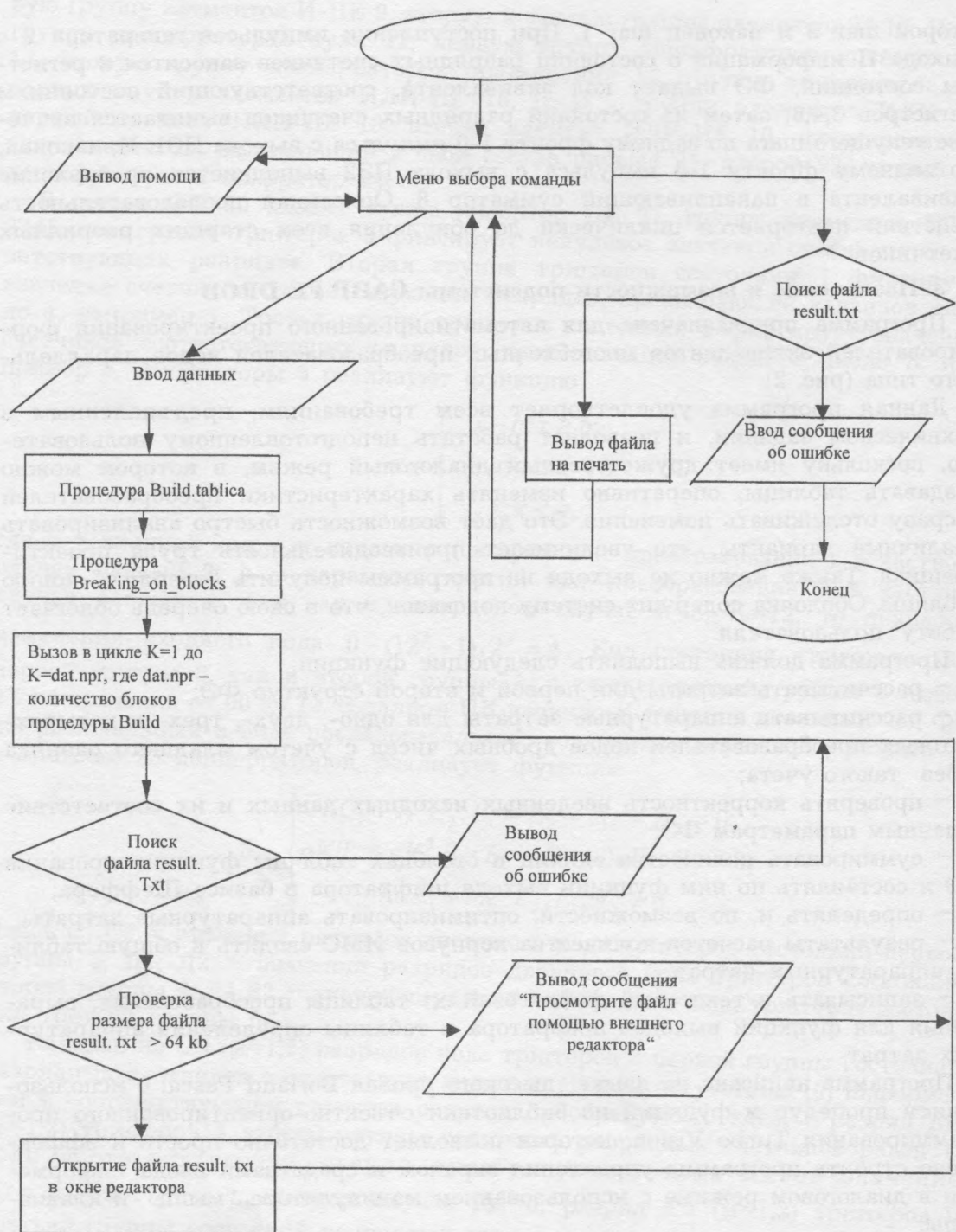


Рис. 2. Общая структура программы

емкость свободного пространства определяется исходя из сложности решаемой задачи, но составляет не менее 300 kb), математический сопроцессор. Используемыми программными средствами являются: ОС MS DOS версии 3.0 и выше.

Для своей работы программа FE_DROB использует такие файлы :

Fe_drob.exe -выполняемая программа;

Result.txt -файл результатов расчета преобразователя;

RRR.txt -вспомогательный файл с таблицей функционирования блока;

Time.txt - файл отчета с временем обработки последней задачи и общим количеством ИМС в ней.

Перечень стандартных процедур:

Assign, Read, ReadLn, Write, WriteLn, Reset, Rewrite, Close – процедуры работы с файлами;

INC – инкремент;

DEC – декремент;

STR – перевод числового значения в строковое;

VAL – перевод строкового значения в числовое;

FRAC – взятие дробной части числа;

TRUNC – перевод вещественного значения в целое с дробной частью;

ROUND – перевод вещественного значения в целое с округлением до ближайшего целого;

LN – натуральный логарифм.

Процедуры библиотеки Turbo Vision :

Execute Dialog – метод объекта Tprogram для раскрытия диалогового окна и обмена данными между ним и программой;

NewStatusKey, NewItem, NewSItem, MessageBox, InputBox, InsertWindow, GetExtent.

Перечень оригинальных процедур:

Процедура **RW** служит для подсчета количества строк в таблице функционирования ФЭ и вычисления количества анализируемых строк таблицы в зависимости от числа шагов преобразования.

Процедура **YZnew** формирует и выводит в файл результатов функции выходов ФЭ с одновременным подсчетом в массив SUM количества единиц в каждом столбце таблицы.

Функция **Equal** сравнивает два столбца таблицы. Имеет два входных параметра – col1 и col2, обозначающих номера сравниваемых столбцов по счету. В случае полного совпадения столбцов функция принимает значение true.

Процедура **Optimize** используется для оптимизации аппаратных затрат ФЭ методом разбиения таблицы функционирования на классы эквивалентных столбцов. Процедура просматривает массив SUM и, находя столбцы с одинаковым количеством единиц, сравнивает их с помощью функции Equal. Если столбцы окажутся одинаковыми, то для одного из них значение количества единиц в массиве SUM обнуляется (для него не нужно формировать функцию выхода, так как она уже сформирована для другого выхода, эквивалентного ему). Таким образом, после выполнения процедуры Optimize массив SUM будет выглядеть так, чтобы в дальнейшем затраты рассчитывались только для классов столбцов, а не по каждому в отдельности.

Процедура **IMS** вычисляет количество элементов 1x8 И-НЕ, 2x4 И-НЕ, 3x3 И-НЕ, 4x2 И-НЕ, 4x2 ИЛИ и 1x6 НЕ в долях корпусов каждой из ИМС, необходимое для реализации функций выхода. Результаты расчетов заносятся в следующие переменные :

R8[col,1] – количество элементов 1x8 И-НЕ для каждого из столбцов;

R8[col,2] – количество элементов 2x4 И-НЕ для каждого из столбцов;

R8[col,3] – количество элементов 3x3 И-НЕ для каждого из столбцов;

R8[col,4] – количество элементов 4x2 И-НЕ для каждого из столбцов;

AR8 – общее количество элементов 1x8 И-НЕ;

AR4 – общее количество элементов 2x4 И-НЕ;

AR3 – общее количество элементов 3x3 И-НЕ;

AR2 – общее количество элементов 4x2 И-НЕ;

IlI[col] – количество элементов 4x2 ИЛИ для каждого из столбцов;

HeI[col] – количество элементов 1x6 НЕ для каждого из столбцов;

allI – общее количество элементов 4x2 ИЛИ.

Процедура **DC_MX** подсчитывает затраты построения входного дешифратора, а также затраты на построение для второй структурной реализации ФЭ, включающие входные мультиплексоры и сдвигатель. Результаты расчетов представляются в следующем виде:

He – количество инверторов;

dc2 – количество дешифраторов типа “DC 2 в 4”;

dc3 – количество дешифраторов типа “DC 3 в 8”;

dc4 – количество дешифраторов типа “DC 4 в 16”;

mx – количество мультиплексоров типа “MS 2 в 1” либо “MS 4 в 1”.

Процедура **TABnew** выполняет построение таблицы аппаратурных затрат для первой или второй структурной реализации ФЭ. Она использует для своей работы 4 локальных процедуры : St, St0, St1 и CASE_FR, которые имеют следующее назначение :

St – выводит верхнюю строку таблицы результатов;

St0 – выводит строку-разделитель между строками таблицы;

St1 – выводит пустую строку таблицы.

Процедура **CASE_FR(n)** – выводит в столбец “Всего” общее количество корпусов и вентилях для корпусов, имеющих 2, 3, 4 или 6 вентилях, и в столбец “ИМС” общее количество корпусов данного типа. Также ведет подсчет общего количества корпусов в переменную allims.

Функция **ReadNFromStr** служит для прочтения чисел строки, введенных значений весов шагов преобразования.

Функция **Inccod** – возвращает значение true, если в таблице посчитаны эквиваленты для всех значений триггеров состояния.

Функция **GradeI** – возвращает число основания системы счисления в степени заданной переменной grade:integer. Использует i:integer, y:real – рабочие переменные, применяющиеся для промежуточных подсчетов.

Процедура **Breaking_on_blocs** – производит разбивку преобразователя на блоки. Использует переменные k1,k3,k4: k1 – количество блоков с разрядностью +1, k3 – разрядность всех блоков без +1, k2 – рабочая переменная.

Размер программы:

размер исходного текста программы на языке Turbo Pascal – файла FE_LION.pas составляет 25505 байт;

размер исполнимого модуля - файла FE_LION.exe равен 114768 байт;

размер вспомогательных файлов: time.txt – до 58 байт; rrr.txt – до 5Мбайт и более; result.txt – до 30Мбайт и более.

Размер файлов rrr.txt и result.txt зависит от решаемой задачи и резко возрастает при увеличении числа входных разрядов блока.

После разработки и отладки программы были произведены некоторые экспериментальные исследования ее работы. Эксперименты проводились на ЭВМ IBM Pentium-133 под управлением ОС WINDOWS 95 и с объемом ОЗУ, равным 32 Мбайт. В частности, эксперименты проводились с целью определить время построения таблиц функционирования и аппаратурных затрат для преобразователей кодов, которые рассматриваются только с теоретической точки зрения. Для преобразователей кодов, реализуемых на практике, время построения таблиц не превышает нескольких секунд (табл. 1–4).

При применении ПК параллельного типа с последовательным использованием шагов преобразования с разрядностью в блоке до 6, а лучше до 4, затраты и экономические, и материальные во много раз меньше, чем в ПК последовательного типа с такими же характеристиками.

Пример работы программы:

Вычисление двоичных кодов для формирователя эквивалентов.

Основание системы счисления: 10.

Число шагов: 2.

Шаги: 21.

Количество 10-ичных разрядов входного кода: 6.

Количество блоков преобразователя: 2.

Количество 2-ичных разрядов выходного кода: 24.

Числа дробные.

Тип преобразователя: параллельный.

Блок номер: 1.

Таблица 1

№ строки	Состояние триггеров	Значение эквивалента	
		Десятичное	Двоичное
1	000 000	0.000000000000000E+0000	0000 0000 0000 0000 0000 0000
2	000 001	1.000000000000000E-0003	0000 0000 0100 0001 1000 1001
3	000 010	1.000000000000000E-0002	0000 0010 1000 1111 0101 1100
4	000 011	1.100000000000000E-0002	0000 0010 1101 0000 1110 0101
5	000 100	1.000000000000000E-0001	0001 1001 1001 1001 1001 1001
6	000 101	1.010000000000000E-0001	0001 1001 1101 1011 0010 0010
7	000 110	1.100000000000000E-0001	0001 1100 0010 1000 1111 0101
8	000 111	1.110000000000000E-0001	0001 1100 0110 1010 0111 1110
9	001 XXI	2.000000000000000E-0003	0000 0000 1000 0011 0001 0010
10	010 XIX	2.000000000000000E-0002	0000 0101 0001 1110 1011 1000
11	011 XII	2.200000000000000E-0002	0000 0101 1010 0001 1100 1010
12	100 IXX	2.000000000000000E-0001	0011 0011 0011 0011 0011 0011
13	101 IXI	2.020000000000000E-0001	0011 0011 1011 0110 0100 0101
14	110 IIX	2.200000000000000E-0001	0011 1000 0101 0001 1110 1011
15	111 III	2.220000000000000E-0001	0011 1000 1101 0100 1111 1101

Количество 10-ичных разрядов входного кода : 3.

Разряды с 1-го по 3-й.

$$\begin{aligned}
 Y_1 &= \overline{Z_2} | \overline{Z_4} | \overline{Z_6} | \overline{Z_8} | \overline{Z_9} | \overline{Z_{10}} | \overline{Z_{11}}, Y_2 = \overline{Z_2} | \overline{Z_5} | \overline{Z_7} | \overline{Z_9} | \overline{Z_{11}} | \overline{Z_{13}}, \\
 Y_3 &= \overline{Z_2} | \overline{Z_4} | \overline{Z_5} | \overline{Z_6} | \overline{Z_7} | \overline{Z_9} | \overline{Z_{12}} | \overline{Z_{14}}, Y_4 = \overline{Z_3} | \overline{Z_4} | \overline{Z_5} | \overline{Z_9} | \overline{Z_{11}} | \overline{Z_{12}} | \overline{Z_{13}} | \overline{Z_{14}}, \\
 Y_5 &= \overline{Z_1} | \overline{Z_3} | \overline{Z_5} | \overline{Z_6} | \overline{Z_{10}} | \overline{Z_{11}} | \overline{Z_{12}}, Y_6 = \overline{Z_2} | \overline{Z_3} | \overline{Z_6} | \overline{Z_8} | \overline{Z_{10}} | \overline{Z_{12}} | \overline{Z_{13}}, \\
 Y_7 &= \overline{Z_2} | \overline{Z_3} | \overline{Z_6} | \overline{Z_7} | \overline{Z_{10}} | \overline{Z_{12}} | \overline{Z_{13}} | \overline{Z_{14}}, Y_7 = \overline{Z_2} | \overline{Z_3} | \overline{Z_6} | \overline{Z_7} | \overline{Z_{10}} | \overline{Z_{12}} | \overline{Z_{13}} | \overline{Z_{14}}, \\
 Y_8 &= \overline{Z_7} | \overline{Z_9} | \overline{Z_{10}} | \overline{Z_{13}}, Y_9 = \overline{Z_6} | \overline{Z_7} | \overline{Z_9} | \overline{Z_{10}} | \overline{Z_{11}} | \overline{Z_{12}}, \\
 Y_{10} &= \overline{Z_4} | \overline{Z_5} | \overline{Z_6} | \overline{Z_7} | \overline{Z_{13}} | \overline{Z_{14}}, Y_{11} = \overline{Z_4} | \overline{Z_5} | \overline{Z_6} | \overline{Z_7} | \overline{Z_{11}} | \overline{Z_{12}} | \overline{Z_{13}} | \overline{Z_{14}}, \\
 Y_{12} &= \overline{Z_{11}} | \overline{Z_{12}} | \overline{Z_{13}} | \overline{Z_{14}}, Y_{13} = 0, Y_{14} = 0, Y_{15} = 0, Y_{16} = 0, Y_{17} = 0, Y_{18} = 0, Y_{19} = 0, \\
 Y_{20} &= 0, Y_{21} = 0, Y_{22} = 0, Y_{23} = 0, Y_{24} = 0.
 \end{aligned}$$

Таблица 2

Тип корпуса	y24-y23	y22	y21-y20	y19-y18	y17-y12	y11	y10-y1	Всего	ИМС
1*8 И-НЕ			1		1		1	18	18
2*4 И-НЕ		1/2		1/2		1/2	1/4	3/4	1
2*DC 3 В 8								2	2
Итого									22

Блок номер : 2.

Таблица 3

№ строки	Состояние триггеров	Значение эквивалента	
		Десятичное	Двоичное
1	000 000	0.000000000000000E+0000	0000 0000 0000 0000 0000 0000
2	000 001	1.000000000000000E-0006	0000 0000 0000 0000 0001 0000
3	000 010	1.000000000000000E-0005	0000 0000 0000 0000 1010 0111
4	000 011	1.100000000000000E-0005	0000 0000 0000 0000 1011 1000
5	000 100	1.000000000000000E-0004	0000 0000 0000 0110 1000 1101
6	000 101	1.010000000000000E-0004	0000 0000 0000 0110 1001 1110
7	000 110	1.100000000000000E-0004	0000 0000 0000 0111 0011 0101
8	000 111	1.110000000000000E-0004	0000 0000 0000 0111 0100 0110
9	001 XXI	2.000000000000000E-0006	0000 0000 0000 0000 0010 0001
10	010 XIX	2.000000000000000E-0005	0000 0000 0000 0001 0100 1111
11	011 XII	2.200000000000000E-0005	0000 0000 0000 0001 0111 0001
12	100 IXX	2.000000000000000E-0004	0000 0000 0000 1101 0001 1011
13	101 IXI	2.020000000000000E-0004	0000 0000 0000 1101 0011 1100
14	110 IIX	2.200000000000000E-0004	0000 0000 0000 1110 0110 1010
15	111 III	2.220000000000000E-0004	0000 0000 0000 1110 1000 1100

Количество 10-ичных разрядов входного кода : 3.
Разряды с 4-го по 6-й.

$$\begin{aligned} Y1 &= \overline{Z2} | \overline{Z4} | \overline{Z6} | \overline{Z8} | \overline{Z9} | \overline{Z10} | \overline{Z11}, Y2 = \overline{Z2} | \overline{Z5} | \overline{Z7} | \overline{Z9} | \overline{Z11} | \overline{Z13}, \\ Y3 &= \overline{Z2} | \overline{Z4} | \overline{Z5} | \overline{Z6} | \overline{Z7} | \overline{Z9} | \overline{Z12} | \overline{Z14}, Y4 = \overline{Z3} | \overline{Z4} | \overline{Z5} | \overline{Z9} | \overline{Z11} | \overline{Z12} | \overline{Z13} | \overline{Z14}, \\ Y5 &= \overline{Z1} | \overline{Z3} | \overline{Z5} | \overline{Z6} | \overline{Z10} | \overline{Z11} | \overline{Z12}, Y6 = \overline{Z2} | \overline{Z3} | \overline{Z6} | \overline{Z8} | \overline{Z10} | \overline{Z12} | \overline{Z13}, \\ Y7 &= \overline{Z2} | \overline{Z3} | \overline{Z6} | \overline{Z7} | \overline{Z10} | \overline{Z12} | \overline{Z13} | \overline{Z14}, Y7 = \overline{Z2} | \overline{Z3} | \overline{Z6} | \overline{Z7} | \overline{Z10} | \overline{Z12} | \overline{Z13} | \overline{Z14}, \\ Y8 &= \overline{Z7} | \overline{Z9} | \overline{Z10} | \overline{Z13}, Y9 = \overline{Z6} | \overline{Z7} | \overline{Z9} | \overline{Z10} | \overline{Z11} | \overline{Z12},, \\ Y20 &= 0, Y21 = 0, Y22 = 0, Y23 = 0, Y24 = 0. \end{aligned}$$

Таблица 4

Тип корпуса	y24-y13	y12	y11-y8	y7	y6-y1	Всего	ИМС
1*8 И-НЕ			1		1	10	10
2*4 И-НЕ		1/2		1/2		1	1
2*DC 3 В 8						2	2
Итого							13

В данной работе разработан программный продукт, позволяющий анализировать аппаратурные затраты для первой и второй структурных реализаций ФЭ для многоступенчатых ПК дробных чисел с учетом и без учета младшего разряда. Основания чисел входного кода в диапазоне от 3 до 15 и количество разрядов от 1 до 80, число шагов преобразования от 1 до 4. Для выполнения поставленной задачи на базе программ FEPPSOPR и COL IMS была создана программа на языке TURBO Pascal FE DROB, которая позволяет пользователю, выбрав пункт меню “ДАННЫЕ”, ввести исходные данные и получить на экране монитора таблицу функционирования ФЭ, выражения для функций выходов шифратора и таблицы анализа аппаратурных затрат для первой или второй структурных реализаций ФЭ. Таким образом, разработанный программный продукт позволяет существенно облегчить работу по анализу и оптимизации аппаратурных затрат ФЭ и в считанные секунды провести исследование структуры ФЭ, на которые до появления программ COL_IMS, FEPP, FEPPSOPR, FE_DROB и им подобных требовалось несколько десятков часов вычислений вручную.

Список литературы: 1. Довгань С.И., Сбитнев А.И. Персональные ЭВМ. Турбо Паскаль V7.0. // Объектное программирование. К.: “Довгань”, 1995. 320с. 2. Зуев Е.А. Язык программирования Turbo Pascal 6.0, 7.0 // Радио и связь. М.: Веста, 1993. 384 с. 3. Фаронов В.В. Турбо Паскаль (в 3-х книгах) // Книга 2-я. Библиотека Turbo Vision. М.: Учебно-инженерный центр “МВТУ-ФЕСТО ДИДАКТИК”. 1994. 429 с.

Поступила в редколлегию 22.04.2002

Голян Вера Владимировна, инженер 1к., н.с. ПНИИ ХНУРЭ. Научные интересы: информатика, автоматизация проектирования цифровых устройств. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 30-28-14.

Какурин Николай Яковлевич, канд. техн. наук, профессор кафедры АПВТ ХНУРЭ. Научные интересы: прикладная теория цифровых автоматов, автоматизация проектирования цифровых устройств. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 40-93-26.

Макаренко Анна Николаевна, канд. техн. наук, доцент кафедры АПВТ ХНУРЭ. Научные интересы: информатика, автоматизация проектирования цифровых устройств. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 40-93-26.

Для преподавателей университетов,
научных работников, аспирантов

Тематика:

- **Компьютерная инженерия**
- **Математическое моделирование**
- **Оптимизация и процессы управления**
- **Автоматизация проектирования и диагностика**
- **Информационные интеллектуальные системы**
- **Проектирование интегральных схем и микросистем**
- **Компьютерные технологии в образовании**

Адрес редакции: 61166, Харьков, просп. Ленина, 14,
ХНУРЭ, тел. 40-93-26, e-mail: ri@kture.kharkov.ua