

читающего двоичного счета. При циклическом выполнении этих строк значения переменных идентификатора изменяются аналогично разрядам 4-разрядного счетчика. Строка 8 реализует логическую операцию конъюнкции согласно (1). Операция конъюнкции выполняется над переменными 1—4, а результат записывается как значение первой переменной из области определения, т. е. переменной 2. Строка 9 реализует дизъюнкцию над переменными 4, 1 согласно (2) совместно для T1 и T2. Строка 10 формирует значение переменных (1—4) как результат поразрядного сравнения T2 и T3 согласно (3).

Для записи результатов по разрядам (независимо от указанных в первой области определения) предусмотрена запись с использованием конструкций, содержащих знак (·) при логических операциях и вторую область определения. Например, выражение ЗАДАТЬ T1 =  $\neg \cdot [1/2][3/4]$  инвертирует значения переменных (1, 2) T1, а результат записывается по разрядам (3, 4) T1.

Решение ряда задач контроля электронной аппаратуры на языке ТЕСТ требует программного изменения области определения (замена переменных по условию). Для таких случаев предусмотрено использование в качестве переменных области определения идентификаторов числа. Заменяя в предыдущем примере переменные (1—4) на идентификаторы числа A1—A4, получим выражение ЗАДАТЬ T1 =  $\neg \cdot [A1/A2][A3/A4]$ . Это выражение в зависимости от значений идентификаторов числа теперь применимо в различных блоках рабочих программ контроля.

В заключение отметим, что определенный набор логических операций и разработанные средства программирования реализованы на ЭВМ «Электроника-100И» и используются в АСК электронной аппаратуры. Эти программные средства описания процессов генерирования позиционных кодов управления могут найти также применение в различных АСУ ТП.

#### ЛИТЕРАТУРА

1. Кудрицкий В. Д., Сеница М. А., Чинаев П. И. Автоматизация контроля РЭА. М., «Сов. радио», 1977.
2. Поспелов Д. А. Логические методы анализа и синтеза схем. М., «Энергия», 1974.
3. Буровцев В. А., Ибрагимов К. Ш., Подзин А. Е. Система аппаратно-функционального контроля и диагностики цифровых схем, управляемая малой ЭВМ.— «Упр. сист. и маш.», 1976, № 3, с. 44—48.
4. Ибрагимов К. Ш., Корня И. Х., Подзин А. Е. Возможности математического обеспечения автоматической системы диагностики цифровых узлов.— В кн.: Вопросы электроники. Кишинев, «Штиинца», 1977.

Поступило в редакцию 15 декабря 1977 г.

УДК 681.3.06

Б. З. КИРИЛЕНКО  
(Кишинев)

#### РЕАЛИЗАЦИЯ АЛГОРИТМОВ ФОРМИРОВАНИЯ ПОЗИЦИОННЫХ КОДОВ ПРОИЗВОЛЬНОЙ РАЗРЯДНОСТИ

Функциональные возможности и быстродействие системного программного обеспечения автоматизированных систем контроля (АСК) электронной аппаратуры в значительной степени зависят от применяемых алгоритмов генерирования позиционных кодов управления (ПКУ). В то же время вопросам разработки и реализации эффективных алгоритмов генерирования ПКУ в литературе уделяется мало внимания.

В настоящей работе предлагается один из возможных алгоритмов формирования ПКУ, который обеспечивает достаточно высокое быстродействие при экономном использовании памяти ЭВМ. Программный модуль, реализующий данный алгоритм, — один из модулей интерпретатора с языка ТЕСТ [1]. Рассматриваемый алгоритм формирования позиционных кодов реализован в виде единой управляющей программы, имеющей  $n$  разветвлений для вычисления заданных операций, приведенных в работе [2]. Список реализуемых операций включает следующие операции: «Константа 0», «Константа 1», тождественность, инверсия, дизъюнкция, конъюнкция, циклические сдвиги вправо и влево на 1 разряд, суммирующий и вычитающий двоичный счет и др.

Исходные данные для управляющей программы — описание позиционного управления на языке ТЕСТ, включающее логическую операцию и обрабатываемый операнд [2, 3].

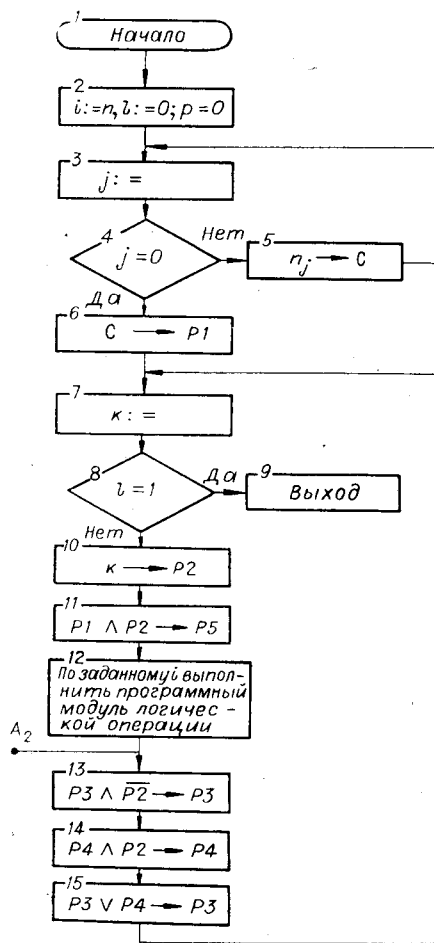


Рис. 1.

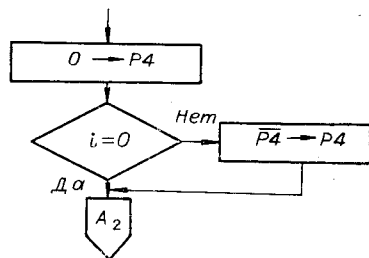


Рис. 2.

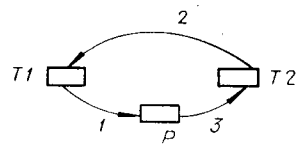


Рис. 3.

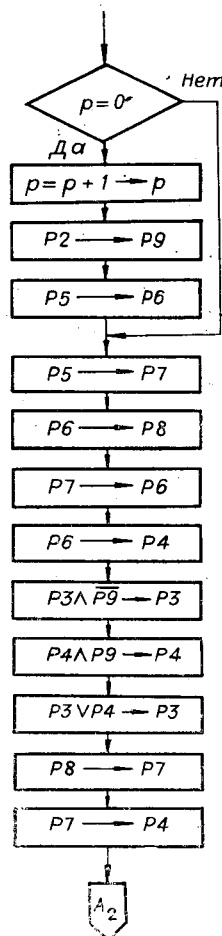


Рис. 4.

Одна из основных особенностей реализации алгоритмов заключается в том, что выполнение логических операций производится одновременно с проведением синтаксического и семантического анализа выполняемой программы. Блок-схема алгоритма представлена на рис. 1. В него введены следующие обозначения:  $i$  — номер логической операции;  $l$  — признак конца области определения;  $j$  — номер операнда;  $k$  — номер обрабатываемого разряда;  $P1$  — регистр операнда;  $P2$  — регистр позиционного кода номера разряда унитарного;  $P3$  — регистр результата;  $P4$  — рабочий регистр;  $P5$  — регистр исходного значения обрабатываемого разряда;  $C$  — стек;  $p$  — номер прохода.

Рассмотрим работу алгоритма. Блок 2 формирует код  $i$  выполняемой логической операции, по которому устанавливается адрес передачи для выполнения программного модуля соответствующей операции.

Блоки 3—5 осуществляют запись значений идентификаторов операнда, указанных в рабочей программе, в стек. Блок 6 записывает в регистр операнда  $P1$  обрабатываемый

операнд, представляющий собой ПКУ. Регистр операнда в зависимости от числа разрядов обрабатываемого операнда может включать требуемое число слов ЭВМ. Блок 7 выделяет очередной номер обрабатываемого разряда операнда. Блоки 8, 9 осуществляют выход из модуля (конец алгоритма), если обработана вся область определения. Преобразование значения номера разряда в позиционный унитарный код и запись его в регистр позиционного кода номера разряда  $P_2$  выполняет блок 10. Несмотря на произвольное число разрядов в унитарном коде номера разряда, при реализации алгоритма в качестве регистра  $P_2$  используется всего одно слово ЭВМ. Это стало возможным в результате применения программного счетчика числа разрядов, кратного числу разрядов в слове ЭВМ, описывающем регистр операнда. Блок 11 осуществляет логическое выделение содержимого по обрабатываемому разряду и запись его в регистр  $P_5$ . Выделение содержимого обрабатываемого разряда в отдельном регистре позволяет производить над ним функциональные преобразования, оставляя при этом информацию по другим разрядам операнда без изменения. Блок 12 вычисляет значение разряда позиционного кода управления в соответствии с заданной логической операцией  $i$ . Запись результата вычисления значения информации по обрабатываемому разряду в регистр результата осуществляют блоки 13—15. При выполнении записи результата вычисления информация по остальным разрядам сохраняется.

В качестве примера реализации алгоритма вычисления ПКУ по обрабатываемому разряду операциями «Константа 0» и «Константа 1» приведем блок-схему программного модуля (рис. 2). Более сложным примером является реализация логической операции циклического сдвига на один разряд вправо. Данная операция находит широкое применение в АСК для синтеза контролируемых наборов функциональных узлов. Суть его состоит в том, что последовательно производится обмен информацией между первым и текущим разрядами области определения (рис. 3). На рисунке введены следующие обозначения:  $T_1$  — триггер разряда ПКУ, указанного в области определения первым;  $T_2$  — триггер текущего разряда;  $P$  — рабочий регистр. Стрелками указаны такты выполнения операций обмена. Первоначально в триггеры  $T_1$  и  $T_2$  записывается значение первого разряда операнда. В первом такте рабочий регистр  $P$  принимает состояние триггера  $T_1$ , во втором — триггер  $T_1$  принимает состояние триггера  $T_2$  и результат записывается по первому разряду операнда, в третьем — информация из регистра  $P$  переписывается в триггер  $T_2$  и результат из триггера  $T_2$  записывается по текущему разряду операнда. При вычислении последующих разрядов информация из следующих разрядов регистра операнда поступает в  $T_2$  и далее выполняются описанные такты. Таким образом, содержимое первого разряда операнда обновляется каждый раз из триггера  $T_1$ , а содержимое текущего разряда — из триггера  $T_2$ .

На рис. 4 приводится блок-схема описанного алгоритма формирования ПКУ. На рисунке введены, кроме перечисленных ранее, следующие обозначения:  $P_6$  — регистр разряда ПКУ, указанного в области определения первым;  $P_7$  — регистр текущего разряда;  $P_8$  — рабочий регистр;  $P_9$  — регистр позиционного унитарного кода номера разряда, указанного в области определения первым.

Реализация алгоритма формирования ПКУ выполнена в программном обеспечении АСК на базе мини-ЭВМ «Электроника-100И» и имеет следующие характеристики: скорость вычисления одного разряда ПКУ — около 400 мкс; объем оперативной памяти ЭВМ, занимаемый соответствующим программным модулем, составляет около 1 К слов, а полный объем интерпретирующей программы с языка ТЕСТ занимает 3,5 К слов памяти ЭВМ.

## ЛИТЕРАТУРА

1. Ибрагимов К. Ш., Клисторин И. Ф., Подзин А. Е. О реализации математического обеспечения системы контроля БИС ЗУ.— «Электронная техника», 1977, сер. 8, с. 36—42.
2. Корня И. Х. Программные средства генерирования позиционных кодов управления в автоматизированной системе контроля электронной аппаратуры.— «Автоматрия», 1978, № 4, с. 127—129.
3. Буровцев В. А., Ибрагимов К. Ш., Подзин А. Е. Управляемая малой ЭВМ система аппаратно-функционального контроля цифровых схем.— «Упр. сист. и маш.», 1976, № 3, с. 44—48.

Поступило в редакцию 15 декабря 1977 г.