

Рис. 2.

Однако эти регистры построены так же, как и регистры, предназначенные для хранения и сдвига информации, применяемые в арифметических устройствах (АУ) ЭЦВМ, что приводит к их неоправданному усложнению. На рис. 3 предлагается схема сдвигающего регистра для сдвига двоичных чисел, содержащих всего одну единицу, которая требует сравнительно меньше элементов. Вначале триггер  $T_1$  устанавливается в единичное положение, а триггеры  $T_2 \div T_n$  — в нулевое положение. При поступлении импульсов по каналу  $K_0$  триггер  $T_y$  и логические элементы  $I_0, I'_0$  обеспечивают поочередное появление этих импульсов в каналах «установка нуля» триггеров четных разрядов  $K'_0$  и «установка нуля» триггеров нечетных разрядов  $K''_0$ . Благодаря этому от каждого импульса единица сдвигается на один разряд влево и на единичных выходах триггеров  $T_1 \div T_n$  поочередно появляется управляющий потенциал.

Анализ работы вышеописанных схем представляет определенный интерес. Известно, что время задержки сигнала переноса в схеме одного разряда сумматора  $\tau_{3П}$  состоит из двух величин: времени задержки сигнала на триггере  $\tau_3$ ; времени задержки сигнала на линии задержки  $\tau_{ЛЗ}$ .

В схемах на рис. 1, 2 при работе устройств, т. е. при поступлении импульсов на счетные входы триггеров четных или нечетных разрядов, импульс переноса от каждого четного или нечетного разряда в следующий четный или нечетный разряд соответственно попадает через время  $2\tau_3$ . Если принять  $\tau_3 \approx (0,6 \div 0,8) \tau_{ПП}$  ( $\tau_{ПП}$  — время протекания переходных процессов), то триггеры находятся в таком режиме, когда на их входы подаются импульсы с частотой

$$f \approx \frac{1}{2\tau_3} < \frac{1}{\tau_{ПП}}, \quad (1)$$

что обеспечивает надежную работу устройств. Время задержки сигнала на ЛЗ определяется из условия

$$\tau_{ЛЗ} \geq (n-1) \tau_3. \quad (2)$$

Следовательно, для суммирования двух  $n$ -разрядных чисел без учета знакового разряда в предложенном сумматоре потребуется время

$$T_{\Sigma} = \tau_{ПП} + [(n-2) + (n-1)] \tau_3 = \tau_{ПП} + (2n-3) \tau_3, \quad (3)$$

которое меньше времени суммирования в существующих схемах аналогичных сумматоров\*.

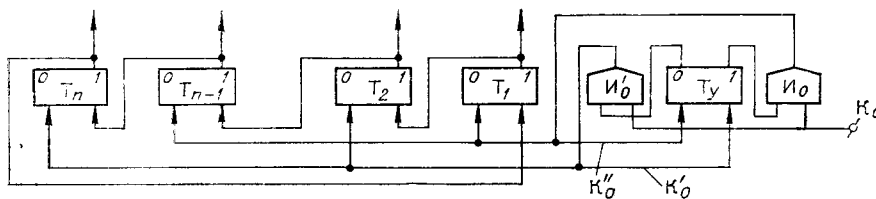


Рис. 3.

\* Е. А. Дроздов, В. А. Комарницкий, А. П. Пятибратов. Электронные цифровые вычислительные машины. М., Воениздат, 1968.

Предельная частота следования импульсов  $f_v$  по каналу вычитания в предложенном реверсивном счетчике определяется исходя из условия

$$f \leq \frac{1}{(2n-3)\tau_3} \quad (4)$$

Выражения (2), (3), (4) позволяют определять без большого труда  $\tau_{ЛЗ}$ ,  $T_{\Sigma}$  и  $f_v$  для заданного  $\tau_3$  и  $n$ . Так, например, при  $\tau_3 = 0,1$  мкс и  $n = 10$   $\tau_{ЛЗ} = 0,9$  мкс,  $T_{\Sigma} \approx 1,8$  мкс,  $f_v \approx 588$  кГц.

В заключение отметим, что на основе описанных схем (см. рис. 1, 2, 3) построены лабораторные образцы, испытания которых показали правильность предложенных принципов. Применение этих узлов в измерительно-информационных системах, в цифровых измерительных устройствах и в малых цифровых вычислительных машинах целесообразно и представляет определенный практический интерес.

Поступило в редакцию 29 мая 1972 г.;  
окончательный вариант — 10 января 1973 г.

УДК 681.3.053

Я. Л. ЛИБЕРМАН  
(Свердловск)

### ОБ ОДНОМ АЛГОРИТМЕ СИНТЕЗА КОДОВЫХ КОЛЕЦ

В последние годы в аналого-цифровых преобразователях различного назначения все шире используются кодовые кольца,  $n$ -членные отрезки которых представляют собой комбинации, принадлежащие некоторому числопозиционному коду [1, 2]. Использование их позволяет существенно упростить кодирующие устройства и сократить объем оборудования. В работе анализируется один из наиболее простых алгоритмов синтеза таких колец, предложенный автором настоящей статьи, который состоит в следующем:

1) формируется исходная числовая последовательность  $T_1 = \{t\}$  вида

$$\underbrace{00 \dots 0}_{n-1} \underbrace{100 \dots 0}_{n-1} \underbrace{200 \dots 0}_{n-1} \underbrace{30 \dots 0}_{n-1} \dots \underbrace{0(m-3)00 \dots 0}_{n-1} \underbrace{00 \dots 0}_{n-1} \underbrace{0(m-2)00 \dots 0}_{n-1} \underbrace{0(m-1)}_{n-1},$$

где  $m$  — основание кода, реализуемого кольцом;

2) последовательность  $T_1$  преобразуется в последовательность  $T_2$  отбрасыванием членов  $t_1, t_2, t_3, \dots, t_n$ ;

3) производится проверка равенства длины последовательности  $T_2$  единице (если равенство не выполняется, то производится операция по п. 4, если выполняется — по п. 6);

4) из последовательности  $T_2$  путем увеличения на единицу членов  $t_{kn}, t_{kn}$  и  $t_{kn-1}$ ;  $t_{kn}, t_{kn-1}$  и  $t_{kn-2}$ ;  $\dots$ ;  $t_{kn}, t_{kn-1}, t_{kn-2}, \dots, t_{kn-(n-2)}$ , где  $k=1, 2, 3, \dots$ , формируются последовательности  $T_3, T_4, T_5, \dots, T_{n+1}$ ;

5) последовательность  $T_{n+1}$  принимается за  $T_1$  и производится переход к п. 2;

6) последовательность  $T_2$  преобразуется в  $(n-1)$ -членную последовательность  $T'_2$  вида

$$(m-1)(m-1) \dots (m-1);$$

7) последовательности, полученные по п. п. 1, 4, 6, выписываются друг за другом в порядке формирования.

Пусть, например, требуется синтезировать кодовое кольцо, реализующее комбинации числопозиционного кода с  $n=3$  и  $m=4$ . Тогда в соответствии с п. 1 исходная последовательность примет вид 0001002003. Далее по п. 2 получится последовательность 1002003, а из нее по п. 4 — последовательности 1012013 и 1112113. После этого, согласно пп. 5 и 2, найдется последовательность 2113, из которой по п. 4 получатся последовательности 2123 и 2223. И наконец, в соответствии с пп. 5, 2 и 3 найдется последовательность, состоящая лишь из члена 3, повторение которого по п. 6 образует последовательность 33. Если теперь по п. 7 выписать друг за другом последовательности, сформированные по пп. 1, 4 и 6, то получится кодовое кольцо

—000100200310120131112132123222333

Достоинства описанного алгоритма очевидны (он пригоден как для машинной, так и для «ручной» реализации; от кольца, синтезированного с его помощью, всегда можно