



Методы синтеза логических схем модульного контроля в натуральных двоичных кодах

(Балтийский государственный технический университет «Военмех» им. Д.Ф. Устинова, Санкт-Петербург)

Рассмотрены методы синтеза логических схем модульного контроля в натуральных двоичных кодах. Получены оценки сложности и быстродействия схем, синтезируемых различными методами.

1. Введение

Устройства модульного контроля параллельных двоичных кодов находят достаточно широкое применение [1-8] в современных цифровых системах управления, передачи и переработки дискретной информации, и, прежде всего, в системах специального назначения, к надежности, достоверности функционирования и контролепригодности которых предъявляются высокие требования. Они применяются в качестве средств аппаратного контроля в системах, использующих контроль по модулю [1-5], арифметические [6-8] и ряд других кодов, в устройствах, работающих в системе остаточных классов [4], в аппаратуре

кодирования и декодирования помехозащищенных кодов в системах передачи информации.

В современных цифровых системах используется как числовой, так и кодовый модульный контроль. В случае числового модульного контроля синтезируемая схема должна формировать остаток параллельного двоичного кода по модулю K с учетом весов его разрядов. При кодовом модульном контроле схема формирует остаток по модулю K числа единичных разрядов контролируемого кода, при этом веса всех его разрядов считаются единичными.

Известны устройства модульного контроля, работающие в унитарных позиционных и непозиционных двоичных кодах. В работах [9-11] исследованы методы их синтеза, получены оценки сложности и быстродействия схем, синтезируемых различными методами. Достоинством таких устройств является их высокое быстродействие, а основным недостатком - большая сложность, что ограничивает возможности их использования, в особенности при значениях модуля $K > 5$. В данной статье рассматриваются и исследуются методы синтеза логических схем модульного контроля, функционирующих в натуральных двоичных кодах. Несмотря на то, что такие устройства широко применяются на практике, а вопросам их проектирования посвящено большое количество работ [1-5, 12, 14, 27], проблема не получила достаточно полного теоретического решения.

2. Синтез логических схем формирования остатка двоичного кода по модулю K методом последовательной свертки

Анализ известных схемных реализаций устройств модульного контроля двоичных кодов показывает, что большинство из них может быть сведено [12] к обобщенной структуре, состоящей из двух последовательно соединенных блоков. Первый блок осуществляет операцию свертки входного n -разрядного кода $X = \{x_1, x_2, \dots, x_n\}$, в результате формируется $p \leq (2\lambda + 1)$ -разрядный код остатка Z , где λ - мощность множества значений остатков весов разрядов входного кода по модулю K . Второй блок формирует натуральный двоичный код остатка $Q = (Z) \bmod K = (X) \bmod K$. В общем случае его реализация требует специальных методов синтеза, рассматриваемых далее.

Введем понятие (m, k) -оператора C_m , где $k < m$, который по набору X формирует набор $X' = \{x'_1, x'_2, \dots, x'_k\}$ такой, что

$$(1) \quad \left(\sum_{i=1}^m w_i x_i \right) \bmod K = \left(\sum_{i=1}^k w_i x'_i \right) \bmod K .$$

Очевидно, что осуществляя последовательную свертку кода X с использованием оператора C_m , получим $p \leq \lambda + 1$ - разрядный код $Z = \{z_1, z_2, \dots, z_p\}$ остатка входного кода по модулю K .

Первый блок схемы формирования остатка двоичного кода по модулю K , синтезируемой методом последовательной свертки, выполняется в виде композиции блоков свертки, реализующих оператор C_m . Порядок их соединения может быть различным от параллельного до последовательного. При параллельном соединении синтезируемая схема имеет древовидную структуру и на каждом уровне группа блоков производит свертку разрядов, сформированных блоками предыдущих уровней, блоки первого уровня производят свертку входных разрядов. Число блоков свертки на каждом уровне зависит от мощности множества значений остатков весов разрядов входного кода по модулю K , его разрядности и способа свертки. Схемы такого типа имеют наибольшее быстродействие. При последовательном соединении на каждом из уровней схема содержит один блок свертки, производящий свертку разрядов, сформированных блоками предыдущих уровней, а также разрядов входного кода. Возможно большое число промежуточных вариантов соединения блоков свертки, приводящих к получению схем различной глубины и сложности.

Рассмотрим возможные варианты реализации оператора C_m .

2.1. Свертка равновесных разрядов кода блоками сложения по модулю два (C_m^1)

Блок сложения обеспечивает свертку по модулю два разрядов $x_i \in X_j$ с весами, удовлетворяющими условию $(w_i) \bmod K = w = \text{const}$. Он выполняется [12,25] на сумматорах, причем сумматоры соединяются входами с входами блока и выходами суммы сумматоров предыдущих уровней данного блока. Выход суммы последнего сумматора является выходом блока с весом w , а выходы переноса всех сумматоров являются выходами блока с весом $2w$. Возможны различные варианты соединения сумматоров в блоке сложения по модулю два. Наибольший интерес представляют два предельных варианта их соединения: последовательный и параллельный.

В первом варианте осуществляется последовательное соединение сумматоров, при котором первый сумматор соединен входами с входами блока, а каждый последующий сумматор соединен двумя входами с входами блока, а третьим входом - с выходом суммы предыдущего сумматора. При такой реализации блока получаем регулярную структуру в виде последовательно соединенной цепочки сумматоров. Ее недостатком является большая глубина схемы, равная $H = \lceil m/2 \rceil h_s^s$, где m - число входов блока, а h_s^s - глубина схемы сумматора по выходу суммы, и, следовательно, низкое быстродействие.

Во втором варианте осуществляется параллельное соединение сумматоров в виде древовидной структуры, при котором сумматоры первого уровня соединены входами с входами блока, а сумматоры каждого из последующих уровней - с выходами суммы сумматоров предыдущих уровней. При таком соединении глубина блока сложения по модулю два оказывается минимальной, равной $H = \lceil \log_3 m \rceil h_s^s$.

Возможно также большое число промежуточных вариантов реализации блока сложения по модулю два, отличающихся глубиной схемы.

Рассматриваемый вариант реализации оператора C_m приводит к формированию из входного m -разрядного набора $X_j \subset X$ выходного $k = \lceil m/2 \rceil + 1$ - разрядного кода $X' = \{x'_1, x'_2, \dots, x'_k\}$, удовлетворяющего условию (1), причем

$$(2) \quad \begin{cases} x'_1 = \bigoplus_{\{i\}_j} x_i, \\ x'_i = p_{i-1} \quad (i = 2, 3, \dots, k), \end{cases}$$

где $x_i \in X_j$, а p_{i-1} - функция, реализуемая на выходе переноса $(i-1)$ -го сумматора блока сложения по модулю два, $w'_i = (w_i) \bmod K = w$, а $w'_i = (2w) \bmod K$.

2.2. Свертка равновесных разрядов кода с использованием параллельного счетчика (C_m^2)

В данном случае блок свертки выполняется [12] в виде композиции параллельных счетчиков, реализующих оператор e_n .

Оператор e_n представляет собой (m,k) -оператор, где $k=\lceil \log_2(m+1) \rceil$, который по набору $X=\{x_1, x_2, \dots, x_m\}$ формирует набор $Z=\{z_1, z_2, \dots, z_k\}$, являющийся двоичной записью числа единиц в X . Вопросы реализации таких операторов подробно рассмотрены в [13], где приведены оценки сложности и глубины их схем.

Данный вариант реализации оператора C_m приводит к формированию из входного m -разрядного набора $X_j \subset X$, веса разрядов x_i которого удовлетворяют условию $(w_i) \bmod K = w = \text{const}$, выходного $k=\lceil \log_2(m+1) \rceil$ - разрядного кода $X'=\{x\zeta_1, x\zeta_2, \dots, x\zeta_k\}$, являющегося двоичной записью числа единиц в X_j , причем сформированный код удовлетворяет условию (1), а его i -й разряд $x\zeta_i$ имеет вес $w\zeta_i=(2^{i-1}w) \bmod K$.

Для данного варианта свертки наиболее удобны структуры с параллельным соединением блоков свертки.

2.3. Свертка разрядов кода многоразрядным сумматором (C_m^3) .

В данном случае блок свертки выполняется [12,14] в виде композиции многоразрядных сумматоров двоичных чисел.

Использование для свертки кода многоразрядного сумматора приводит к формированию из входного $m=(2x+1)$ -разрядного набора $X_j \subset X$, имеющего веса разрядов, равные $w_i=w_{x+i}=(2^{i-1}w_1) \bmod K$, где w_1 - веса разрядов x_1, x_{x+1}, x_{2x+1} входного набора X , подаваемых на входы первого разряда x -разрядного сумматора, выходного $k=(x+1)$ - разрядного кода $X'=\{x\zeta_1, x\zeta_2, \dots, x\zeta_k\}$, удовлетворяющего условию (1), разряд $x\zeta_i$ которого имеет вес $w\zeta_i=(2^{i-1}w_1) \bmod K$. Схема устройства формирования остатка по модулю $K=7$, синтезированная данным методом приведена на рис.1.

2.4. Свертка разрядов входного кода блоками модульного сложения (C_m^4) .

В данном случае блок свертки выполняется [15,16] в виде композиции устройств сложения по модулю K двух двоичных кодов. Они отличаются от модульных сумматоров более широкими функциональными возможностями, обеспечивая суммирование по модулю K входных кодов X_j , $X_i > K-1$.

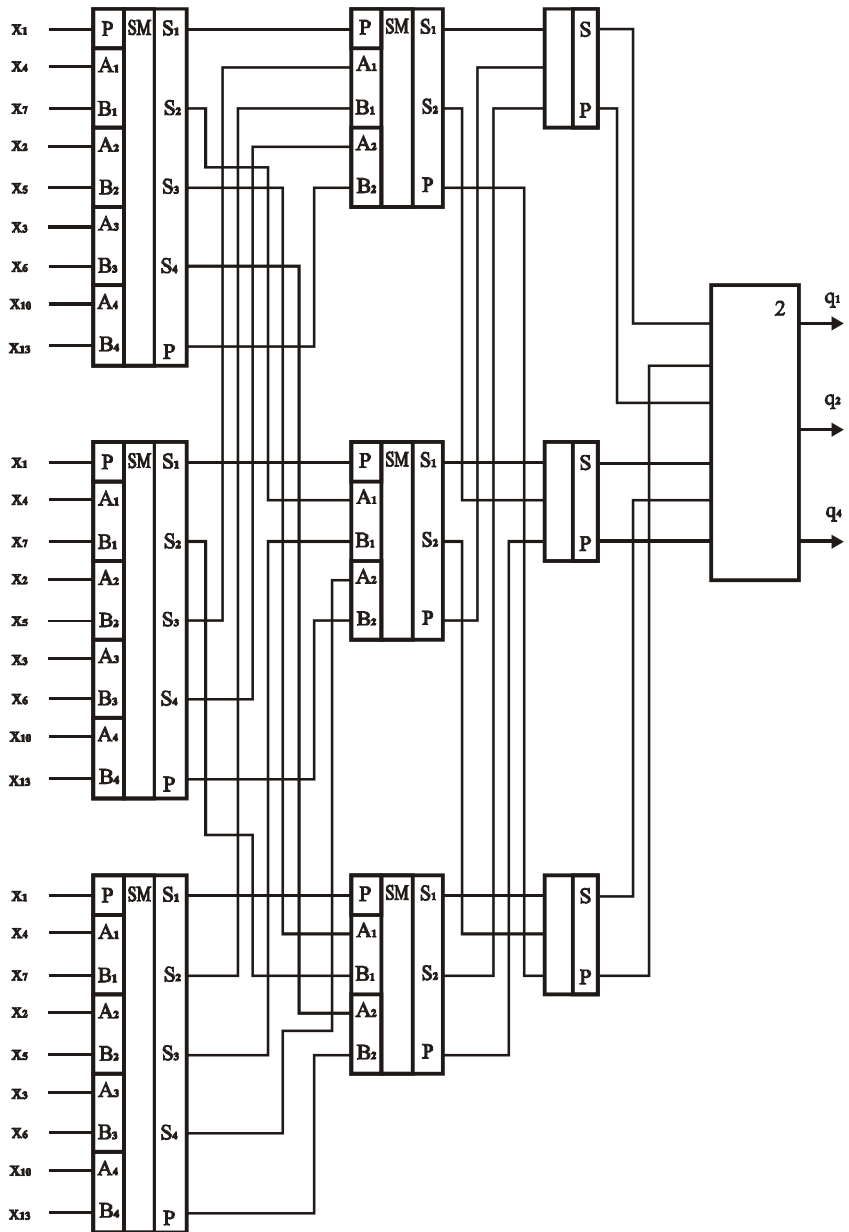


Рис. 1. Схема формирования остатка натурального двоичного 27-разрядного кода по модулю $K=7$, синтезированная методом свертки многоразрядными сумматорами

Использование для свертки кода устройства модульного сложения приводит к формированию из входного $m=2t$ -разрядного набора $X_j \in X$, где $t \in \lfloor \log_2 K \rfloor$, имеющего веса разрядов, равные $(2^{i-1} w_i) \bmod K$ ($i = 1, 2, \dots, t$), где w_i - веса разрядов входного набора X , подаваемых на входы первого разряда устройства, выходного $k \in \lfloor \log_2 K \rfloor$ -разрядного набора $X' = \{x_1, x_2, \dots, x_k\}$, удовлетворяющего условию (1), i -й разряд которого имеет вес $w'_i = (2^{i-1} w_i) \bmod K$. Если выходные коды блоков удовлетворяют условию $X' < K$, то их дальнейшая свертка может осуществляться модульными сумматорами.

Для данного способа свертки наиболее удобны структуры с параллельным соединением блоков модульного сложения. С целью упрощения синтезируемого устройства могут использоваться блоки модульного сложения, формирующие на ряде входных наборов не приведенный по модулю K код.

Рассмотренный способ свертки наиболее эффективен в случае $K=2^r-1$, при этом в качестве блоков свертки могут использоваться r -разрядные сумматоры с обратной связью с выхода переноса старшего разряда на вход переноса первого разряда. Необходимо отметить, что такие блоки свертки на ряде наборов входного кода X такого, что $(X) \bmod K = K$, дают не приведенный код остатка $X' = K$.

Возможна модификация данного метода свертки [17], заключающаяся в том, что сначала свертка осуществляется по модулю $K' = bK$, где $b = 2, 3, \dots, m$, а затем производится свертка сформированного кода по модулю K . Наиболее удобным является случай, когда $K' = 2^g - 1$, поскольку в этом случае блок свертки по модулю K может быть выполнен в виде многоразрядного сумматора с обратной связью. Следует отметить, что предварительная свертка по модулю K' практически не влияет на сложность синтезируемой схемы, но может приводить к существенному возрастанию ее глубины.

2.5. Свертка с использованием отрицательных весов разрядов входного кода $\binom{C_m^s}{m}$

Представим входной натуральный двоичный код $X = x_1 2^0 + x_2 2^1 + \dots + x_n 2^{n-1}$ в виде

$$(3) \quad X = X_0 W_0 + X_1 W_1 + \dots + X_r W_r,$$

где $X_i = x_{il+1}2^0 + x_{il+2}2^1 + \dots + x_{(i+1)l}2^{l-1}$, $r = \lfloor n/l \rfloor - 1$, W_i - целое число, равное весу $w_{il+1} = 2^{il}$ разряда x_{il+1} кода X , а $X = \bigcup_{i=0}^p X_i$.

При этом

$$(4) \quad (X) \bmod K = \left(\sum_{i=0}^r X_i 2^{il} \right) \bmod K.$$

Для натурального двоичного кода X всегда существует такое $l \leq \lfloor K/2 \rfloor$, что

$$(5) \quad (W_i) \bmod K = (2^{il}) \bmod K = (2^l)^i \bmod K = (-1)^i.$$

С учетом свойства (5) получаем

$$(6) \quad (X) \bmod K = \left(\sum_{i=0}^r X_i (-1)^i \right) \bmod K = (X_0 - X_1 + X_2 - \dots - X_r) \bmod K$$

Наиболее эффективно такое представление при $K=2^r+1$, когда $l=r$.

Представление (6) позволяет реализовать несколько вариантов свертки.

Первый вариант [18] основан на следующем представлении уравнения (6):

$$(7) \quad (X) \bmod K = (S_1 - S_2) \bmod K = ((S_1) \bmod K - (S_2) \bmod K) \bmod K = \{(X_0 + X_2 + \dots + X_{2\lfloor r/2 \rfloor}) \bmod K - (X_1 + X_3 + \dots + X_{2\lfloor r/2 \rfloor - 1}) \bmod K\} \bmod K$$

Суммирование (свертка) кодов X_i в каждой группе представления (7) может осуществляться любым из рассмотренных выше методов. В результате суммирования формируются два кода остатков сумм S_1 и S_2 , после чего вычисляется остаток по модулю K их разности. Вычисление разности целесообразно реализовать с использованием инверсного кода.

Данный вариант свертки можно рассматривать как частный случай рассмотренного ранее варианта свертки C_m^3 при дополнительных ограничениях на порядок свертки.

Второй вариант [19] использует последовательную свертку входного кода X многоразрядными сумматорами в соответствии со следующим представлением:

$$(8) \quad \begin{cases} S_i^j = (S_g^{j-1} + S_{g+2}^{j-1} + p_i^j) \bmod 2^l, \\ p_i^j = \lfloor (S_g^{j-1} + S_{g+2}^{j-1} + p_i^j - S_i^j) / 2^l \rfloor \end{cases}$$

где $j=1, \dots, \lceil \log_2 \Gamma \rceil - 1$ – номер сумматора; $i=1, \dots, \lfloor n/2 \rfloor$ – номер группы выходов j -го сумматора;

$$\gamma = \begin{cases} 2i - 1 & \text{при } i = 2\lfloor i/2 \rfloor, \\ 2i - 2 & \text{при } i \neq 2\lfloor i/2 \rfloor, \end{cases}$$

$S_i^j = \{s_{i1}^j, \dots, s_{(i-1)\gamma+1}^j\}$ – i -я группа выходов суммы j -го сумматора;

p_i^j – сигнал переноса li -го разряда j -го сумматора, причем $p_0^1 = 0$,

а $p_0^i = p_{\lfloor n/2 \rfloor}^{i-1}$;

$$S_i^0 = X_{i-1}.$$

При таком порядке свертки вес сигнала переноса $w(p_i^j) = (-1)^j$ равен весу разрядов s_{li+1}^{j-1} и $s_{(l+1)i+1}^{j-1}$, подаваемых на входы i -го разряда сумматора.

В результате свертки получаем $(\lambda+1)$ -разрядный код $S = \{s_{2l+1}, s_{2l}, \dots, s_1\} = \{p_2^t, S_2^t, S_1^t\}$, где $t = \lceil \log_2 \Gamma \rceil - 1$, а $\lambda = 2l$, разряды которого имеют веса $W = \{2^{2l}, \dots, 2^1, 2^0\}$, причем их остатки по модулю K равны $W' = \{1, -2^l, \dots, -1, 2^l, \dots, 1\}$. Последующая свертка до получения $(l+1)$ -разрядного кода может быть проведена путем вычитания $(S_1 - S_2 + P_2)$, реализуемого на l -разрядном сумматоре с использованием инверсного кода \bar{S}_2 .

Данный вариант свертки является частным случаем рассмотренного выше варианта свертки C_m^3 .

Третий вариант [20], который можно назвать конвейерной сверткой, реализуется на основе следующего представления

$$(9) \quad (X) \bmod K = \left(\dots \left((X_0 - X_1) \bmod 2^l + X_2 + p_1 \right) \bmod 2^l \right) - \dots - X_r + p_{r-1} \bmod K,$$

где p_r – сигнал переноса, формируемый при выполнении операции $(X_{i-1} \pm X_i)$.

Вычисление разности выполняется с использованием инверсного кода

$$(10) \quad (X) \bmod K =$$

$$\left(\dots \left(\left(\overline{X_0 + X_1} \right) \bmod 2^l + X_2 + p_1 \right) \bmod 2^l \right) + \dots + X_r + p_{r-1} \bmod K$$

и реализуется следующим образом

$$(11) \quad \begin{cases} S_j = (\overline{S_{j-1}} + X_j + p_{j-1}) \bmod 2^l, \\ p_j = \left\lfloor \frac{\overline{S_{j-1}} + X_j + p_{j-1}}{2^l} \right\rfloor \end{cases}$$

где $j=1,2,\dots,r$, $S_0=X_0$, а $p_0=0$.

Сигналы переноса p_i имеют вес, равный весу переменных младшего разряда следующего сумматора $|(W_i) \bmod K|=1$. Получаемая структура состоит из r последовательно соединенных l -разрядных сумматоров. Выходные сигналы всех сумматоров инвертируются. Это позволяет избежать необходимости введения поправок, обусловленных использованием для реализации вычитания инверсного кода, так как поправки компенсируются. В результате свертки формируется $(l+1)$ - разрядный код Z . При формировании окончательного результата необходимо учитывать его знак, так как при четном r результат свертки формируется в инверсном коде. Схема формирования остатка по модулю $K=9$, синтезированная данным методом, приведена на рис.2.

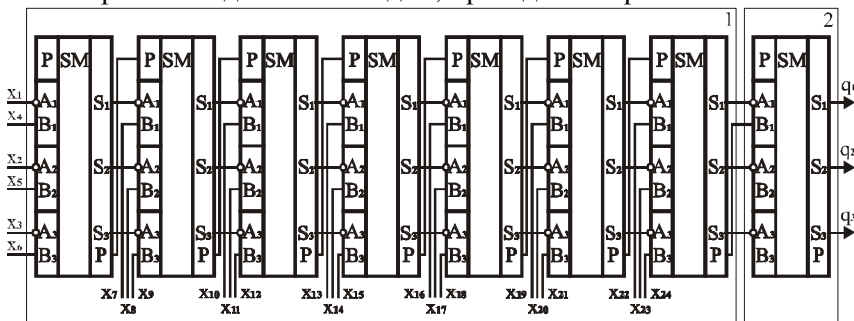


Рис. 2. Схема формирования остатка натурального двоичного 24-разрядного кода по модулю $K=9$, синтезированная методом конвейерной свертки

Четвертый вариант [21] заключается в последовательной свертке с использованием полных сумматоров разрядов входного кода и сигналов переноса, формируемых при суммировании, имеющих одинаковые значения $|(W_i) \bmod K|$, выполняемой в следующем порядке

$$(12) \quad \begin{cases} s_i^1 = x_i \oplus \bar{x}_{l+i} \oplus x_{2l+1}, \\ p_i^1 = F_3^2(x_i, \bar{x}_{l+i}, x_{2l+1}), \end{cases}$$

$$(13) \quad \begin{cases} s_1^j = \bar{s}_1^{j-1} \oplus p_l^{j-1} \oplus x_{(j+1)l+i}, \\ p_1^j = F_3^2(\bar{s}_1^{j-1}, p_l^{j-1}, x_{(j+1)l+i}), \\ s_i^j = \bar{s}_i^{j-1} \oplus \bar{p}_{i-1}^{j-1} \oplus x_{(j+1)l+i}, \\ p_i^j = F_3^2(\bar{s}_i^{j-1}, \bar{p}_{i-1}^{j-1}, x_{(j+1)l+i}), \end{cases}$$

где $j=2, \dots, r-2$, $i=1, \dots, r$,

s_i^j, p_i^j - сигналы на выходах суммы и переноса i -го сумматора j -й группы,

$F_3^2(z_1, z_2, z_3)$ - пороговая равновесная функция трех переменных с порогом $a=2$.

В результате свертки получаем $l=2r$ -разрядный код Z . При формировании кода остатка необходимо учитывать в прямом или инверсном виде он получен. Схема формирования остатка по модулю $K=9$, синтезированная данным методом, приведена на рис.3.

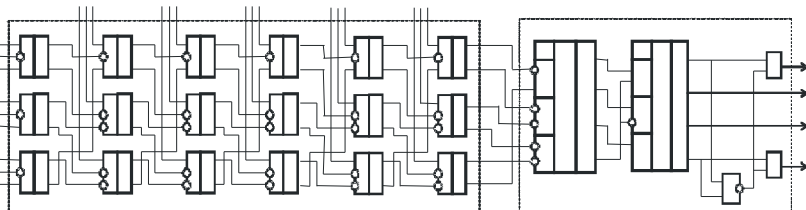


Рис. 3. Схема формирования остатка натурального двоичного 24-разрядного кода по модулю $K=9$, синтезированная методом свертки полными сумматорами

Возможно одновременное использование рассмотренных вариантов свертки с целью минимизации объема оборудования или глубины схемы за счет оптимального использования номенклатуры имеющихся многоразрядных сумматоров двоичных чисел, приводящее к получению комбинированных структур.

3. Синтез схем формирования кода остатка по модулю K

В том случае, когда входной код X является натуральным двоичным кодом, а $K=2^r-1$, любой из рассмотренных вариантов свертки $C_m^1, C_m^2, C_m^3, C_m^4$ приводит к получению двух r -разрядных кодов Z_1 и Z_2 ($Z=Z_1 \cup Z_2$) из которых суммированием по модулю K формируется код $Q=\{q_1, q_2, \dots, q_r\}$ остатка входного кода X по модулю K . Т.е. в этом случае второй блок осуществляет суммирование по модулю K двух r -разрядных кодов и при $Z_1, Z_2 < K$ выполняется в виде модульного сумматора.

Если $Z_1, Z_2 \geq K$, либо входной код X не является натуральным двоичным кодом, либо $K \neq 2^r-1$, то в результате свертки входного кода получаем $p \leq (\lambda+1)$ -разрядный код Z , дальнейшая свертка которого с использованием рассмотренных выше способов свертки оказывается невозможной. Для решения последней задачи требуется использованием специальных способов свертки.

3.1. Синтез блока формирования кода остатка по модулю K методом последовательного расширения и свертки.

Процесс синтеза данным методом [22] заключается в последовательном увеличении разрядности кода $Z=\{z_1, z_2, \dots, z_{l+1}\}$ за счет представления его разрядов z_i с весами w_i , удовлетворяющими условию $(w_i) \bmod K \neq 2^e$, в виде $z_i = \{z_i^1, z_i^2, \dots, z_i^r\}$, где r - число единичных разрядов в двоичном представлении остатка по модулю K веса w_i причем $w_i^j = 2^{e_j} < 2^t$, где $t = \lceil \log_2 K \rceil$, а

$\sum_{j=1}^r w_i^j = (w_i) \bmod K$, и последующей свертки сформированного

кода с использованием C_m оператора. Проведение операций последовательного расширения кода Z и его свертки позволяет каждый раз сокращать разрядность кода и мощность множества значений остатков весов разрядов по модулю K и получить $\xi \leq 2t$ - разрядный код $Z\zeta$ который можно представить в виде $Z' = Z_1 \cup Z_2$, где $Z_j = \{z_1^j, z_2^j, \dots, z_l^j\}$ с весами разрядов $w_i = 2^{i-1}$.

Возможен различный порядок проведения свертки. Наибольший интерес представляют варианты последовательной и параллельной свертки. В первом случае свертка осуществляется

многоразрядными сумматорами, соединенными последовательно. При этом расширение кода производится перед каждым суммированием в объеме, обеспечивающем использование всех входов очередного сумматора.

Во втором случае все разряды z_i кода Z с весами $w_i \neq 2^e < 2^t$ представляются набором $\{z_i^1, z_i^2, \dots, z_i^r\}$ разрядов с весами $w_i^j = 2^{e_j}$, после чего производится их свертка любым из рассмотренных способов. При получении в результате суммирования разрядов с весом $w_i > 2^t$, для которых $(w_i) \bmod K \neq 2^e < 2^t$, производится их расширение, а затем свертка сформированного кода.

Свертка производится до получения кода $Z' \leq 2K-2$ разрядностью не более $2t$ из которого сумматором по модулю K формируется код $Q = \{q_1, q_2, \dots, q_t\}$ остатка входного кода X .

3.2. Синтез блока формирования кода остатка по модулю K методом прямого формирования вычетов

Пусть Z -натуральный двоичный код, формируемый в результате свертки. При этом

$$(14) \quad Q(Z) = (\Psi(Z) + W(Z)) \bmod 2^{\lfloor \log_2 K \rfloor},$$

где $Z = \{z_1, z_2, \dots, z_p\}$, $\Psi(Z) = \{z_1, z_2, \dots, z_t\} = (Z) \bmod 2^{\lfloor \log_2 K \rfloor}$,

$W(Z) = \{w_1, w_2, \dots, w_t\}$, причем

$$W(Z) = (2^p - \lfloor Z / K \rfloor K) \bmod 2^{\lfloor \log_2 K \rfloor}, \text{ а } t = \lfloor \log_2 K \rfloor.$$

Код $\Psi(Z)$ может быть сформирован схемой, состоящей из трех последовательно соединенных блоков, первый из которых реализует монотонный (p, r) -оператор $M(p, \{K, 2K, \dots, rK\})$ для кода Z ($r = \lfloor Z_{\max} / K \rfloor = \lfloor (2^p - 1) / K \rfloor$), а второй реализует систему функций

$E = \{E_i\}$, где $E_i = M(p, iK) \overline{M(p, (i+1)K)}$ при $i = K, 2K, \dots, rK$. Третий блок является шифратором $P(E)$, осуществляющим преобразование $E \rightarrow W$ вида $w_j = \bigvee_{\{i\}_j} E_i$, где $\{i\}_j$ - множество индексов функций E_i ,

для которых $E_i = 1$ при $w_j = 1$, и выполняется в общем случае в виде t элементов ИЛИ. Монотонный оператор $M(p, \{K, 2K, \dots, rK\})$

реализует систему p пороговых функций с порогами $a = K, 2K, \dots, rK$ и весами входных переменных $w_i = 2^i$. Вопросы синтеза таких схем подробно рассмотрены в [23].

Реализация устройства может быть несколько упрощена, если формировать код $W(Z)$ непосредственно по коду, формируемому монотонным оператором $M(p, \{jK\})$, на основе следующего преобразования

$$(15) \quad \bigvee_{i=a}^b E_{iK} = M(p, aK) \overline{M(p, (b+1)K)}.$$

Упрощение достигается за счет исключения формирования ряда функций E_i , которые становятся избыточными.

Пример такой реализации блока формирования остатка приведен на рис. 4 для модуля $K=5$.

Сложность рассмотренного преобразователя быстро возрастает с увеличением разрядности p кода Z , в основном из-за сложности реализации монотонного оператора $M(p, \{jK\})$.

3.3. Синтез блока формирования кода остатка по модулю K с использованием отрицательных весов разрядов

Синтез блока формирования кода остатка данным методом возможен при значении модуля $K=2^r+1$, когда получаемый в результате свертки $(\lambda+1)$ -разрядный код $Z = \{z_1, z_2, \dots, z_{\lambda+1}\}$, где $\lambda=2r$, может быть преобразован в код остатка Q следующим образом

$$(16) \quad Q = (Z_1 + 2^r Z_2 + 2^{2r} z_{\lambda+1}) \bmod K = (Z_1 - Z_2 + z_{\lambda+1}) \bmod K,$$

где $Z = Z_1 \cup Z_2 \cup z_{\lambda+1}$, $Z_1 = \{z_1, z_2, \dots, z_r\}$, а $Z_2 = \{z_{r+1}, z_{r+2}, \dots, z_{\lambda}\}$.

Операция вычитания выполняется с использованием инверсного кода, при этом

$$(17) \quad Q = (Z_1 - Z_2 + z_{\lambda+1}) \bmod K = (Z_1 + \bar{Z}_2 - 2^r + 1 + z_{\lambda+1}) \bmod K = \\ \{ (Z_1 + \bar{Z}_2 + z_{\lambda+1}) \bmod K - (2^r - 1) \bmod K \} \bmod K.$$

Откуда для $K=2^r+1$ получаем

$$(18) \quad Q = \{ (Z_1 + \bar{Z}_2 + z_{\lambda+1}) \bmod K + 2 \} \bmod K.$$

Следовательно, при $Z_1 \geq Z_2$

$$(19) \quad \{ (Z_1 + \bar{Z}_2 + z_{\lambda+1}) \bmod K + 2 \} \bmod K = (Z_1 + \bar{Z}_2 + z_{\lambda+1}) \bmod 2^r + 1,$$

а при $Z_1 < Z_2$

$$(20) \quad \{ (Z_1 + \bar{Z}_2 + z_{\lambda+1}) \bmod K + 2 \} \bmod K = (Z_1 + \bar{Z}_2 + z_{\lambda+1}) \bmod 2^r + 2.$$

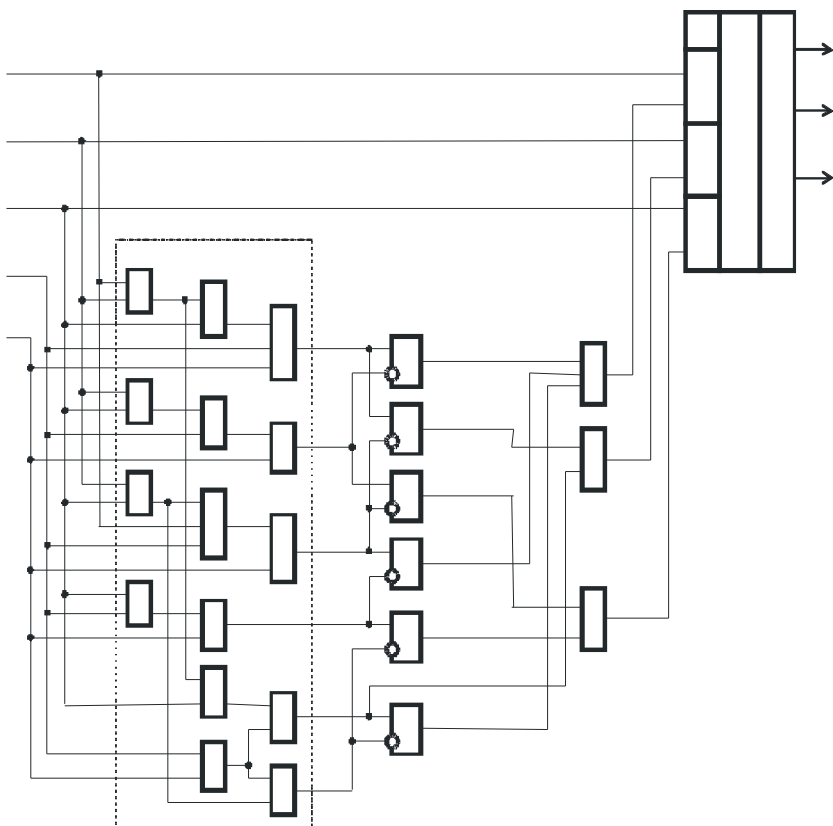


Рис. 4. Схема свертки пятиразрядного двоичного кода по модулю пять, синтезированная методом прямого формирования вычетов

Описанная процедура формирования кода остатка может быть реализована на двух последовательно соединенных r -разрядных сумматорах со схемой коррекции. Сумматоры выполняют операцию

$$(21) \quad Q' = (Z_1 + \bar{Z}_2 + z_{l+1}) \bmod 2^r + p + 2\bar{p} ,$$

где p - сигнал на выходе переноса первого сумматора.

Схема коррекции инвертирует единичные разряды результата суммирования в случае его равенства модулю K :

$$(22) \quad q_i = q'_i \left(\bigwedge_{\{j\}} q'_j \right),$$

где $\{j\}$ - множество номеров единичных разрядов в двоичном представлении модуля K . Такое инвертирование необходимо только при нулевом входном коде Z .

При отсутствии схемы коррекции в ряде случаев будет формироваться не приведенный по модулю K результат суммирования, если он равен K .

3.4. Синтез блока формирования кода остатка по модулю K комбинированными методами

На основе рассмотренных выше могут быть сформированы [24] два комбинированных метода преобразования кода Z в код остатка Q . Первый использует комбинацию метода последовательного расширения и свертки и метода формирования вычетов и заключается в следующем. Сначала первым из методов осуществляется свертка $p \leq \lambda + 1$ - разрядного кода Z в $(t+1)$ -разрядный код Z' , а затем он преобразуется в код остатка Q с использованием метода прямого формирования вычета в описанном выше порядке.

Экономичные схемы преобразования кода Z в код остатка Q синтезируются при одновременном использовании всех трех рассмотренных методов свертки.

$(\lambda+1)$ -разрядный код Z может быть представлен в виде

$$(23) \quad Z = \{Z_1; Z_2; z_{l+1}\} = \{(z_1, z_2, \dots, z_l); (z_{l+1}, z_{l+2}, \dots, z_{2l}); z_{l+1}\},$$

где $l = \lambda/2$, а

$$(24) \quad w_i = \begin{cases} w_i = (2^{i-1}) \bmod K & \text{при } i=1, 2, \dots, l, l+1, \\ w_i - K = -(2^{i-l-1}) \bmod K & \text{при } i=l+1, \dots, l. \end{cases}$$

При этом на первом шаге свертки кода Z формируется код

$$(25) \quad Z' = Z_1 - Z_2 = Z_1 + \bar{Z}_2 + 1 = \{z'_1, z'_2, \dots, z'_{l+1}\},$$

где $z'_{l+1} = \bar{p}$, $z'_i = s_i$, а s_i, p - сигналы, формируемые на i -м выходе суммы и выходе переноса сумматора.

Далее сформированный код Z' и разряд z_{l+1} методом последовательного расширения и свертки преобразуются в $(t+1)$ -разрядный код Z'' , из которого методом прямого формирования вычетов получается код остатка Q . Схема такого преобразователя для случая $K=11$, приведена на рис. 5.

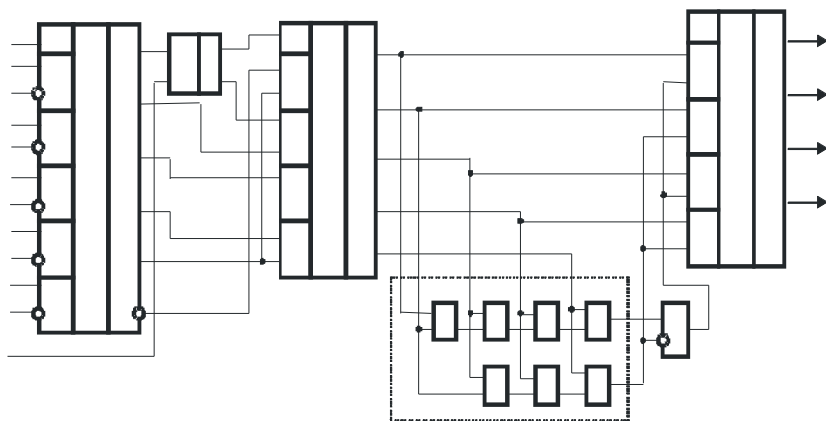


Рис. 5. Схема блока формирования кода остатка по модулю $K=11$, синтезированная комбинированным методом

4. Синтез логических схем подсчета количества единиц двоичного кода по модулю K методом промежуточного преобразования

Схема, формирующая натуральный двоичный код G остатка по модулю K числа единиц входного кода X , может быть синтезирована путем промежуточного преобразования числа единиц кода X в $t \geq 1$ натуральных двоичных кода Y_i числа единиц в X_i , где $i=1,2,\dots, t$, а $X=X_1 \cup X_2 \cup \dots \cup X_t$ и $X_i \cap X_j = \emptyset$ при $i \neq j$. Каждый из преобразователей кода X_i в код Y_i выполняется в виде (n_i, k_i) -оператора e_{n_i} , где $k_i = \lceil \log_2(n_i + 1) \rceil$, который по набору X_i формирует набор Y_i , являющийся двоичной записью числа единиц в X_i . В результате такого преобразования на выходах параллельных счетчиков, реализующих операторы e_{n_i} , получаем параллельный двоичный m -разрядный код $Y = Y_1 \cup Y_2 \cup \dots \cup Y_t = \{y_1, y_2, \dots, y_m\}$, где $m = k_1 + k_2 + \dots + k_t$. Далее для кода Y в описанном выше порядке осуществляется синтез схемы формирования остатка Q по модулю K .

Сложность синтезируемой таким способом схемы составляет

$$(26) \quad L(G(n)) = \sum_{i=1}^t L(e_{n_i}) + L(D(Y)) ,$$

где $L(e_{n_i})$ -сложность реализации оператора e_{n_i} , а $L(D(Y))$ -сложность схемы формирования остатка двоичного кода Y по модулю K .

Выбор параметра t , а также мощности подмножеств переменных X_i позволяет оптимизировать синтезируемую схему по сложности и быстродействию. Решение задачи оптимизации в аналитической форме затруднено из-за рекуррентности оценок сложности реализации операторов e_{n_i} . Результаты проектирования и оптимизации схем на ЭВМ путем перебора возможных значений параметра t и мощности подмножеств переменных X_i показали, что при формировании результата в натуральном двоичном коде выбор $t=1, 2$ позволяет получать наиболее экономичные схемы.

5. Оценки сложности и быстродействия схем модульного контроля в натуральных двоичных кодах

Оценку сложности схем модульного контроля будем проводить для классического базиса И, ИЛИ, НЕ, а оценку быстродействия – по глубине схемы (максимальному числу последовательно соединенных элементов от входа к выходу).

Сложность и глубина схем числового модульного контроля, синтезируемых рассмотренными методами свертки, составляют

$$(27) \quad L(Q) = L(X) + L(Z) ,$$

$$(28) \quad H(Q) = H(X) + H(Z) ,$$

где $L(X), H(X)$ - сложность и глубина блока свертки осуществляющего преобразование $X \rightarrow Z$, а $L(Z), H(Z)$ - сложность и глубина блока формирования кода остатка, осуществляющего преобразование $Z \rightarrow Q$.

В случае натурального двоичного кода и модуля $K=2^r-1$ блок формирования кода остатка выполняется в виде устройства суммирования по модулю K двух r -разрядных кодов. В настоящее время известен ряд вариантов реализации таких устройств. Одним из простейших является реализация в виде r -разрядного сумматора двоичных чисел с обратной связью с выхода переноса r -го разряда на вход переноса и схемой коррекции, инвертирующей выходной код в случае его равенства модулю K . Сложность такой реализации блока равна

$$(29) \quad L(s_K) = rl_s + 2r ,$$

а глубина

$$(30) \quad H(s_K) = rh_s^p + h_s^s + 3 .$$

где h_s^s, h_s^p - глубина полного сумматора по выходам суммы и переноса.

Оценим сложность и быстродействие блоков свертки для рассмотренных в разделе 2 вариантов его реализации. Приводимые оценки глубины схем соответствуют вариантам реализации блока с максимальным быстродействием.

При свертке с использованием сумматоров по модулю два

$$(31) \quad L_1 \leq rl_s \sum_{i=1}^g \lceil n_{i-1} / 2 \rceil ,$$

$$(32) \quad H_1 \leq h_s^s \sum_{i=1}^g \lceil \log_3 n_{i-1} \rceil ,$$

где $n_0 = \lceil n / r \rceil$, $n_i = \lceil n_{i-1} / 2 \rceil + 1$, l_s - сложность полного сумматора, h_s^s, h_s^p - глубина полного сумматора по выходам суммы и переноса, S_i, P ,

$$\mathbf{g} \leq \sum_{i=1}^{a+1} Sgn(n_{i-1} - 2), \quad Sgn(x) = \begin{cases} 1 & \text{при } x > 0, \\ 0 & \text{при } x \leq 0. \end{cases}$$

При свертке с использованием параллельных счетчиков

$$(33) \quad L_2 \approx r \sum_{i=1}^g L(e_{n_i}) = rl_s \sum_{i=1}^g \sum_{j=0}^{\lceil \log_2 m_i \rceil} \lceil n_i / 2^{j+1} \rceil ,$$

$$(34) \quad H_2 \approx \sum_{i=1}^g (\lceil \log_2 n_{i-1} \rceil (h_s^s + h_s^p) - h_s^p) - (g-1) (h_s^s - h_s^p) Sgn(g-2) ,$$

где $n_0 = \lceil n / r \rceil$, $n_i = \log_2 \lceil n_{i-1} \rceil + 1$,

$$a = \lceil \log_2 \rceil n / r \lceil \mathbf{g} \leq \sum_{i=1}^a Sgn(n_{i-1} - 2) .$$

При свертке с использованием многоразрядных сумматоров

$$(35) \quad L_3 \approx rl_s \sum_{i=1}^g \lceil n_{i-1} / 2r \rceil ,$$

$$(36) \quad H_3 \approx h_s^p (\lceil n / r \rceil - 1) + gh_s^s ,$$

где $n_0 = n$, $n_i = n_{i-1} - (r-1) \lfloor n_{i-1}/2r \rfloor + 1 - \lfloor n_{i-1}/2r \rfloor$,

$$g = \sum_{i=1}^a \text{Sgn}(n_{i-1} - 2r), \quad a = \lfloor \log_2 \lfloor n/r \rfloor \rfloor.$$

При свертке с использованием блоков модульного сложения, выполняемых на r -разрядном сумматоре с обратной связью

$$(37) \quad L_4 \leq r l_s \lfloor n/r \rfloor - 2,$$

$$(38) \quad H_4 \leq (r h_s^p + h_s^s) \lfloor \log_2 \lfloor n/r \rfloor \rfloor - 1.$$

В таблице 1 приведены оценки сложности и глубины схем числового модульного контроля, синтезируемых рассмотренными методами свертки, полученные по результатам проектирования.

Анализ полученных оценок а также результатов оптимизации на ЭВМ синтезируемых схем показывает, что сложность рассмотренных вариантов реализации схем числового модульного контроля отличается незначительно и весьма точно определяется оценкой (37). Глубина схем для первых трех вариантов реализации блока свертки в случае древовидных структур при параллельном соединении блоков свертки также отличается незначительно и меньше глубины схем, синтезируемых с использованием свертки блоками модульного сложения, что обусловлено наличием в них обратной связи. При использовании более быстродействующих вариантов реализации блоков модульного сложения, например, предложенных в [28, 29], быстродействие схем модульного контроля может быть существенно повышено. Особо необходимо отметить, что сложность схем числового модульного контроля практически не зависит от значения модуля $K=2^r-1$.

Для случая $K=2^r+1$ сложность и глубина схем формирования остатка для рассмотренных методов свертки C_m^5 с использованием отрицательных весов разрядов кода определяются следующими оценками.

Для варианта конвейерной свертки

$$(39) \quad L_5^3 \leq m r \lfloor l_s + 1 \rfloor + r \lfloor \lfloor m/2 \rfloor - \lfloor m/2 \rfloor \rfloor \lfloor l_s + 1 \rfloor,$$

$$(40) \quad H_5^3 \leq 3 h_s^p \lfloor m + \lfloor m/2 \rfloor - \lfloor m/2 \rfloor \rfloor + h_s^s - h_s^p,$$

где $m = \lfloor n/r \rfloor$.

Таблица 1

Зависимость сложности схем числового модульного контроля от разрядности входного кода n

n	Сложность L / Глубина H															
	$K=7$								$K=15$							
	C_n^1	C_n^2	C_n^3	C_n^4	C_n^1	C_n^2	C_n^3	C_n^4	C_n^1	C_n^2	C_n^3	C_n^4	C_n^1	C_n^2	C_n^3	C_n^4
15	<u>114</u> 36	<u>234</u> 21	<u>126</u> 36	<u>246</u> 23	<u>114</u> 33	<u>234</u> 20	<u>114</u> 48	<u>234</u> 30	<u>111</u> 33	<u>221</u> 20	<u>111</u> 33	<u>221</u> 20	<u>123</u> 34	<u>223</u> 21	<u>111</u> 39	<u>221</u> 25
21	<u>168</u> 39	<u>348</u> 23	<u>168</u> 39	<u>349</u> 23	<u>168</u> 39	<u>348</u> 24	<u>168</u> 48	<u>348</u> 30	<u>173</u> 42	<u>343</u> 27	<u>177</u> 40	<u>347</u> 24	<u>173</u> 39	<u>343</u> 26	<u>173</u> 57	<u>343</u> 36
27	<u>222</u> 42	<u>462</u> 24	<u>222</u> 42	<u>462</u> 25	<u>222</u> 45	<u>462</u> 27	<u>222</u> 63	<u>462</u> 39	<u>219</u> 42	<u>449</u> 25	<u>219</u> 42	<u>449</u> 25	<u>219</u> 49	<u>449</u> 33	<u>219</u> 57	<u>449</u> 36
33	<u>276</u> 48	<u>576</u> 27	<u>324</u> 49	<u>624</u> 29	<u>276</u> 48	<u>576</u> 30	<u>276</u> 63	<u>576</u> 39	<u>281</u> 45	<u>571</u> 26	<u>273</u> 48	<u>563</u> 30	<u>281</u> 48	<u>571</u> 30	<u>281</u> 69	<u>571</u> 44
45	<u>384</u> 54	<u>804</u> 30	<u>384</u> 54	<u>804</u> 31	<u>384</u> 60	<u>804</u> 37	<u>384</u> 63	<u>804</u> 37	<u>389</u> 57	<u>799</u> 33	<u>389</u> 57	<u>799</u> 33	<u>389</u> 57	<u>799</u> 36	<u>389</u> 105	<u>799</u> 66

Примечания:

1. В числителе дроби приведены оценки сложности схем, а в знаменателе - их глубины.
2. Оценки приведены для случаев реализации схем на сумматорах минимальной сложности $l_s=9$, имеющих глубину по выходу суммы $h_s^s = 6$ и по выходу переноса $h_s^p = 3$, и реализации схем на сумматорах минимальной глубины ($h_s^s = 3$, $h_s^p = 2$), имеющих сложность $l_s=19$.

Для варианта последовательной свертки с использованием полных сумматоров

$$(41) \quad L_K^4 \leq (m-2)rl_s + (m-2)(2r-1_s) - (r-1) + L_K,$$

$$(42) \quad H_K^4 \leq (h_s^s + 1)(m-1) + H_K,$$

где L_K , H_K - сложность и глубина блока сложения по модулю K , определяемые следующими оценками

$$(43) \quad L_K = \begin{cases} r(l_s + l_p) + 2r + 4 & \text{при } m = 2[m/2], \\ 2(r-1)l_s + (r+4)l_p + 4r & \text{при } m \neq 2[m/2], \end{cases}$$

$$(44) \quad H_K = \begin{cases} r(h_s^p + h_p^p) + 3 & \text{при } m = 2[m/2], \\ 4h_s^p + 5h_p^p + 2h_s^s + 2 & \text{при } m \neq 2[m/2], \end{cases}$$

Для варианта свертки на многоразрядных сумматорах

$$(45) \quad L_S^2 \leq l_S \sum_{i=1}^g (2ra_i + 2rb_i + (1 - b_i)c_i) + L_K ,$$

$$(46) \quad H_S^2 \leq h_S^p \sum_{i=1}^g (2ra_i + 2rb_i + (1 - b_i)c_i) + H_K ,$$

где $g = \lceil \log_2 \rceil (n - 1) / 2r \lceil \lceil , \quad a_i = \lceil (n_i - 1) / 4r \rceil ,$

$$b_i = \lceil (n_i - 4ra_i - 1) / 2r \rceil , \quad c_i = n_i - 4ra_i - 1 - 2rb_i ,$$

$$n_1 = n , \quad n_{i+1} = 2ra_i + 2rb_i + (1 - b_i)c_i + 1 ,$$

L_K , H_K - сложность и глубина блока сложения по модулю K , выполняемого в соответствии с (21), определяемые следующими оценками

$$(47) \quad L_K = (r + 1)l_S + (r - 1)l_P + r + 5 ,$$

$$(48) \quad H_K = (r + 1)h_S^p + (r - 1)h_P^p + 3 .$$

Схемы, синтезированные на основе представления (7), имеют большую сложность и глубину, чем у рассмотренных выше вариантов свертки C_m^5 .

Сложность и глубина схем числового модульного контроля при $K \neq 2^t \pm 1$ определяются прежде всего характеристиками блока формирования кода остатка, поскольку сложность блоков свертки, синтезируемых с использованием рассмотренных методов свертки, отличается незначительно, в особенности в представляющем практический интерес диапазоне числа переменных n . Получение аналитических оценок сложности и глубины блоков формирования кода остатка для произвольного K в этом случае не представляется возможным.

В таблице 2 приведены оценки сложности и глубины блоков формирования кода остатка, синтезированных рассмотренными выше методами.

Из приведенных оценок видно, что наиболее экономичные схемы блоков формирования остатка для $K = 2^t + 1$ получаются при синтезе методом свертки, использующим представление разрядов кода Z с отрицательными весами. При $K \neq 2^t \pm 1$ наилучшие характеристики имеют схемы, синтезированные комбинированными методами. Следует отметить, что в большинстве случаев минимум сложности и глубины достигается для различных схемных реализаций блока.

Оценки сложности блоков формирования остатка входного $p=(\lambda+1)$ -разрядного код Z по модулю K

Метод синтеза	$K=5$		$K=9$		$K=11$		$K=13$		$K=17$	
Последовательное расширение и свертка	$\frac{59}{23}$	$\frac{99}{15}$	$\frac{134}{42}$	$\frac{244}{31}$	$\frac{194}{52}$	$\frac{374}{36}$	$\frac{134}{42}$	$\frac{244}{31}$	$\frac{134}{42}$	$\frac{244}{31}$
Прямое формирование вычета	$\frac{55}{13}$	$\frac{70}{12}$	$\frac{146}{22}$	$\frac{171}{17}$	*	*	*	*	*	*
Комбинированный 1	$\frac{51}{23}$	$\frac{86}{17}$	$\frac{100}{35}$	$\frac{175}{23}$	$\frac{186}{45}$	$\frac{321}{28}$	$\frac{228}{50}$	$\frac{458}{31}$	$\frac{164}{47}$	$\frac{309}{30}$
Использование отрицательных весов разрядов	$\frac{38}{14}$	$\frac{63}{11}$	$\frac{48}{18}$	$\frac{88}{14}$	-	-	-	-	$\frac{66}{24}$	$\frac{116}{19}$
Комбинированный 2	-	-	-	-	$\frac{126}{48}$	$\frac{231}{33}$	$\frac{144}{35}$	$\frac{259}{26}$	-	-

Примечания:

1. В числителе дроби приведены оценки сложности, а в знаменателе - глубины схем.
2. Оценки приведены для случаев реализации схем на сумматорах минимальной сложности $l_s=9$, имеющих глубину по выходу суммы $h_s^s = 6$, а по выходу переноса $h_s^p = 3$, и сумматорах минимальной глубины, имеющих сложность $l_s=19$, $h_s^s = 3$ и $h_s^p = 2$.
3. «*» - реализация не целесообразна из-за большой сложности схемы.
4. «-» - данный вариант реализации невозможен.
5. Метод «Комбинированный 1» использует методы последовательного расширения и свертки и метод прямого формирования вычета.
6. Метод «Комбинированный 2» использует представление разрядов входного кода с отрицательными весами, метод последовательного расширения и свертки и метод прямого формирования вычета.

6. Заключение

В статье исследованы вопросы синтеза логических схем модульного контроля в натуральных двоичных кодах. Показано, что большинство из них может быть сведено к обобщенной

структуре, состоящей из двух последовательно соединенных блоков. Первый блок осуществляет операцию свертки входного кода, а второй формирует натуральный двоичный код остатка. Рассмотрены возможные варианты реализации обоих блоков. Получены оценки сложности и быстродействия синтезируемых рассмотренными методами схем. Они показывают, что характеристики синтезируемых схем определяются прежде всего сложностью и глубиной используемых при их реализации сумматоров. Использование сумматоров минимальной сложности позволяет получать более экономичные схемы по сравнению со схемами модульного контроля в унитарных параллельных позиционных и непозиционных кодах, причем по мере увеличения модуля K выигрыш в сложности возрастает. Вместе с тем их быстродействие оказывается значительно меньшим.

Приведенные в данной статье результаты формируют достаточно полный теоретический подход к проблеме синтеза логических схем модульного контроля в натуральных двоичных кодах, позволяют определить области их возможного использования, а также точно оценить на ранних стадиях проектирования параметры синтезируемых схем и решить вопрос о целесообразности их применения.

ЛИТЕРАТУРА

1. *Путинцев Н.Д.* Аппаратный контроль управляющих цифровых вычислительных машин. М.: Сов. Радио, 1966.
2. *Гаврилов Ю.В., Пучко А.М.* Арифметические устройства быстродействующих ЭВМ.-М.: Сов. Радио, 1970.
3. *Журавлев Ю.П., Кателюк Л.А., Циклинский Н.И.* Надежность и контроль ЭВМ. М.: Радио и связь, 1978.
4. *Долгов А.И.* Диагностика устройств, функционирующих в системе остаточных классов. М.: Сов. Радио, 1982.
5. *Селлерс Ф.* Методы обнаружения ошибок в работе ЭЦВМ. М.: Мир, 1972.
6. *Питерсон У., Уэлдон Э.* Коды, исправляющие ошибки. М.: Мир, 1976.
7. *Касами Т., Токура Н., Ивадари Ё., Инагаки Я.* Теория кодирования. М.: Мир, 1978.

8. *Дадаев Ю.Г.* Теория арифметических кодов. М.: Радио и связь, 1981.
9. *Музыченко О.Н.* Синтез логических схем модульного контроля в унитарных позиционных двоичных кодах. // Автоматика и телемеханика. 2001. № 3. С. 158 - 173.
10. *Музыченко О.Н.* Устройство для контроля параллельного двоичного кода по модулю К. А.с. 1361557 СССР // Б. И. 1987. № 47. С. 202.
11. *Музыченко О.Н.* Устройство для контроля параллельного двоичного кода по модулю К. А.с. 1425676 СССР // Б. И. 1988. № 35. С. 210.
12. *Музыченко О.Н.* Преобразователь двоичного кода. А.с. 1476614 СССР // Б. И. 1989. № 16. С. 250.
13. *Музыченко О.Н.* Упрощение пороговых схем, синтезируемых методом промежуточного преобразования. // Автоматика и телемеханика. 1990. № 12. С. 164 - 170.
14. *Черкасский Н.В.* Устройство для формирования остатков по модулю. А.с. 1305684 СССР// Б. И. 1987. № 15. С. 250.
15. *Черкасский Н.В., Митьков В.С., Аксарин Л.Л.* Устройство для формирования остатка по модулю три. А.с. 1084799 СССР// Б. И. 1984. № 13. С. 177.
16. *Gajski D.D.* Pat. 4187549 USA Modular modulo 3 module / Filed 17.11.77.
17. *Самойлов А.Л.* Пирамидальная свертка по модулю три. А.с. 1105896 СССР// Б. И. 1984. № 28. С. 145.
18. *Черкасский Н.В.* Устройство для формирования остатков по модулю. А.с. 1736006 СССР// Б. И. 1992. № 19. С. 228-229.
19. *Черкасский Н.В., Титков В.Н.* Устройство для формирования остатка по модулю $m=2^K+1$. А.с. 1339566 СССР// Б. И. 1987. № 35. С. 173-174.
20. *Черкасский Н.В.* Устройство для формирования остатков по модулю. А.с. 1387201 СССР// Б. И. 1988. № 13. С. 259-260.
21. *Черкасский Н.В.* Устройство для формирования остатков по модулю. А.с. 1401610 СССР// Б. И. 1988. № 21. С. 256-257.
22. *Музыченко О.Н.* Устройство для свертки двоичного кода в код по модулю К. А.с. 1425845 СССР // Б. И. 1988. № 35. С. 268.

23. *Музыченко О.Н.* Преобразователь двоичного кода в код по модулю К. А.с. 1429322 СССР // Б. И. 1988. № 37. С. 245.
24. *Музыченко О.Н.* Преобразователь двоичного кода в код по модулю К. А.с. 1732472 СССР // Б. И. 1992. № 17. С. 227.
25. *Черкасский Н.В.* Устройство для формирования остатков по модулю. А.с. 1449986 СССР// Б. И. 1989. № 1. С. 211.
26. *Черкасский Н.В.* Устройство для формирования остатков по модулю. А.с. 1444774 СССР// Б. И. 1988. № 46. С. 220.
27. *Балюк В.В., Выжиковски Р., Каневский Ю.С.* Преобразователь n-разрядного двоичного кода в его представление по модулю М. А.с. 1076899 СССР// Б.И. 1984, № 8. С. 159.
28. *Музыченко О.Н.* Сумматор по модулю семь. А.с. 1603371 СССР // Б. И. 1990. № 40. С. 210.
29. *Музыченко О.Н.* Сумматор по модулю пятнадцать. А.с. 1603375 СССР // Б. И. 1990. № 40. С. 212-215.