P6

М.Г.Карповский, В.И.Руханский, Н.С.Черкасов

ИСПОЛЬЗОВАНИЕ КОРРЕКТИРУЮЩИХ КОДОВ ДЛЯ ОБНАРУЖЕНИЯ И ИСПРАВЛЕНИЯ ОШИБОК В КОНЕЧНЫХ АВТОМАТАХ

I. Синтез дискретных устройств с обнаружением и исправлением ошибок (подобно синтезу устройств без коррекции ошибок) целесообразно разбить на 3 этапа: абстрактный синтез, структурный синтез и комбинационный синтез.

На этапе абстрактного синтеза определяется число состояний устройства (объем памяти) и переходы между состояниями под действием входных сигналов.

На этапе структурного синтеза выбирается кодирование состояний и входных сигналов и строятся булевы функции, описывающие работу комбинационной части синтезируемого устройства.

На этапе комбинационного синтеза производится минимизация построенных булевых функций и строится структурная схема устройства.

Учет того обстоятельства, что синтезируемое устройство должно обладать способностью коррекции ошибок целесообразно производить уже на этапах абстрактного и структурного синтеза, так как при этом в общем случае удается снизить избыточность, требуемую для коррекции ошибок заданной кратности. Известные в настоящее время методы введения избыточности для коррекции ошибок как правило недостаточно полно учитывают конкретные особенности функций, реализуемых синтезируемым устройством. В данном сообщении описывается метод синтеза дискретных устройств с коррекцией ошибок, при использовании которого требуемая избыточность существенно учитывает функции, реализуемые устройством, что в ряде случаев позволяет понизить вводимую избыточность.

2. Будем считать, что входные сигналы для синтезируемого автомата не подвержены влиянию ошибок. В этом случае "истинное" состояние автомата восстанавливается по ошибочному состоянию и по входному сигналу.

Разобьем множество входных сигналов автомата на n_λ непересекающихся подмножеств $\lambda_1, \lambda_2, \lambda_{n_\lambda}$.

Закодируем состояния автомата двоичными наборами так, чтобы расстояние Хэмминга между любыми двумя наборами, соответствующими состояниям, в которых автомат может находиться при входных сигналах из одного и того же подмножества λ_i было бы не меньше $2\ell+1$. Тогда, как показано в /1/, в автомате может быть исправлена ℓ-кратная ошибка.

Описанное выше кодирование состояний автомата может быть осуществлено следующим образом. "Расщепим" каждое состояние исходного автомата a_j на $n_\lambda(a_j)$ эквивалентных состояний, где $n_\lambda(a_j)$ – число подмножеств λ_{js} $(s=1,2,\ldots,n_\lambda(a_j))$ таких, что исходный автомат может находиться в состоянии a_j при наличии на входе хотя бы одного из входных сигналов, принадлежащих каждому из подмножеств λ_{js}.

"Расщепление" будем производить таким образом, чтобы полученный после "расщепления" автомат мог находиться в каждом из своих состояний только при наличии на входе сигналов, принадлежащих одному и тому же подмножеству λ_{js}. Полученный после "расщепления" состояний автомат будет эквивалентен исходному автомату, и, кроме того, этот автомат будет обладать избыточностью в числе состояний по сравнению с исходным автоматом, которая в дальнейшем используется для коррекции ошибок.

Закодируем теперь состояния "расщепленного" автомата,

в которых этот автомат может находиться при входных сигналах из подмножества λ_i $(i=1,2,\ldots,n_\lambda)$ словами кодов $V \oplus \alpha_i$, где $V \oplus \alpha_i$. элементы- ошибок, полученный прибавлением по mod 2 вектора α_i код с коррекцией ко всем словам кода V с коррекцией ℓ ошибок, и вес α_i не превышает ℓ. (В том случае, когда код V групповой, $V \oplus \alpha_i$ являются смежными классами по коду V).

При этом расстояние Хэмминга между любыми двумя неэквивалентными состояниями "расщепленного" автомата, в которых этот автомат может находиться при одном и том же входном сигнале, не меньше $2\ell+1$, и, следовательно, в "расщепленном" автомате может быть исправлена ℓ-кратная ошибка.

Оценим требуемую при использовании описанного метода длину кода (а, следовательно, и требуемый объем памяти избыточного автомата). Обозначим $n_a(\lambda_i)$ – число состояний, в которых исходный автомат может находиться при наличии на входе хотя бы одного из сигналов подмножества λ_i. Тогда, как показано в /1/, число элементов памяти, необходимых для исправления описанным методом ℓ-кратной ошибки или обнаружения 2ℓ-кратной ошибки в том случае, если код V групповой, определяется неравенствами:

$$\begin{cases} \max n_a(\lambda_i) \le B(m, 2\ell+1) \\ ]\log_2 n_\lambda[+]\log_2 \max_i n_a(\lambda_i)[\le m \end{cases} \qquad (I)$$

(Здесь $]N[$ – означает ближайшее большее N целое число; $B(m,2\ell+1)$ – число слов m-разрядного группового кода с исправлением ℓ ошибок). Оценки функции $B(m,2\ell+1)$ приведены в /2/.

Как видно из (I), объем памяти m, необходимый для коррекции ошибок описанным методом, существенно зависит от разбиения множества входных сигналов на непересекающиеся подмножества $\lambda_1, \lambda_2, \ldots, \lambda_{n_\lambda}$. В частности, если в качестве такого разбиения выбрано разбиение, при котором $n_\lambda=1$, то описанный выше метод коррекции ошибок совпадает с методом, предложенным в /3/. Множество всевозможных разбиений множества входных сигналов образуют конечную струтуру /4/. Нижнюю границу для объема памяти m, определяемую неравенствами (I), можно рассматривать как числовой оператор на этой структуре. Наивные методы нахождения минимума этого оператора и оценки требуемого при этом перебора рассмотрены в работе /5/.

3. В качестве примера построим автомат с исправлением одиночной ошибки, заданный таблицей переходов:

табл. I

Вход / Состояние	x_1	x_2	x_3	x_4	x_5	x_6
a_1	a_2	a_3	a_1	a_4	a_6	a_4
a_2	a_4	a_6	a_5	a_2	a_3	a_1
a_3	a_2	a_3	a_5	a_4	a_6	a_4
a_4	a_2	a_6	a_5	a_4	a_3	a_1
a_5	a_2	a_3	a_5	a_4	a_6	a_1
a_6	a_4	a_6	a_5	a_2	a_3	a_4

Как видно из таблицы I, оптимальным разбиением множества входных сигналов в данном случае является разбиение, при котором: $\lambda_1 = \{x_1, x_4\}$, $\lambda_2 = \{x_2, x_5\}$,

$$\lambda_3 = \{x_3\},\ \lambda_4 = \{x_6\};\ n_a(\lambda_1) = n_a(\lambda_2) = n_a(\lambda_3) = n_a(\lambda_4) = 2 \qquad n_\lambda = 4$$

и из (I) $m = 3$.

Так как для данного разбиения $n_\lambda(a_j) = 1$ $(j = 2, 3, 5, 6)$, $n_\lambda(a_j) = 2$ $(j = 1, 4)$, то "расщепленный" автомат может быть получен "расщеплением" состояний a_1 и a_4 на 2 состояния, которые обозначим a_{11}, a_{12} и a_{41}, a_{42}.

Положим $V = \{000, 111\}$; $d_1 = 000$, $d_2 = 001$,

$d_3 = 010$, $d_4 = 100$. Тогда для исправления одиночной ошибки состояния "расщепленного" автомата можно закодировать следующим образом: a_{11}-010, a_{12}-100, a_2-111, a_3-001, a_{41}-000, a_{42}-011, a_5-101, a_6-110.

Таким образом, в данном случае для исправления одиночной ошибки не потребовалось избыточных элементов памяти, в то время как при использовании метода, описанного в /3/, потребовалось бы 3 избыточных элемента памяти. Сокращения избыточности удалось достигнуть за счет использования конкретных особенностей таблицы переходов синтезируемого автомата (табл. I).

4. Избыточность, требуемая для коррекции ошибок заданной кратности описанным выше методом в ряде случаев может быть несколько сокращена за счет использования информации о распределении ошибок на множестве элементов памяти автомата.

Пусть в автомате можно выделить множество из $m - m'$ элементов памяти, в которых могут возникать ошибки произвольной кратности (например, вследствие общности для них элементов цепей питания или синхронизации или наличия в комбинационной части устройства элементов, ошибки в которых приводят к многократным ошибкам в элементах памяти). Пусть кроме того ошибки в остальных m' элементах памяти носят независимый характер и имеют некоторую кратность ℓ'. Тогда общая кратность ошибок $\ell = m - m' + \ell'$.

Автомат с исправлением такого рода ошибок может быть построен методом последовательной декомпозиции /6/ в виде последовательного соединения 2-х автоматов, содержащих соответственно $m - m'$ и m' элементов памяти. Число m элементов памяти избыточного автомата при этом определяется неравенствами:

$$\max n_a(\lambda_i) \leq B(m', 2\ell'+1) \tag{2}$$

$$]\log_2 \max_i n_a(\lambda_i)[\ \leq m -]\log_2 n_\lambda[\ \leq m'$$

– (где, как и ранее, $\{\lambda_1, \lambda_2, \ldots \lambda n_\lambda\}$ – выбранное разбиение множества входных сигналов).

Неравенства (2) являются обобщением неравенств (I) (при $m - m' = 0$) и позволяют в общем случае понизить требуемый объем памяти m по сравнению с неравенствами (I).

Из неравенств (2) следует, что, если m' есть минимальное число элементов памяти, необходимое для коррекции в заданном автомате ошибки кратности ℓ, и если число элементов памяти m автомата больше m', то в оставшихся $m - m'$ элементах памяти могут быть исправлены ошибки произвольной кратности.

Кроме того, из неравенств (2) при $\ell' = 0$ следует, что для исправления ошибки произвольной кратности в автомате достаточно иметь $]\log_2 \max_i n_a(\lambda_i)[$ абсолютно-надежных элементов памяти; при этом общее число элементов памяти равно $m =]\log_2 \max_i n_a(\lambda_i)[+]\log_2 n_\lambda[$

В качестве разбиения $\{\lambda_1, \lambda_2 \ldots \lambda n_\lambda\}$ целесообразно выбрать разбиение, при котором n_λ равно числу входных сигналов, так как при этом разбиении величина $]\log_2 \max_i n_a(\lambda_i)[$ достигает минимума.

Наличие весьма небольшого числа абсолютно надежных элементов памяти в ряде случаев может привести к весьма резкому снижению избыточности для коррекции ошибок заданной кратности. Так для автомата, задаваемого таблицей переходов табл. I, при описанном в п.3 разбиении множества входных сигналов как видно из (2), то есть коррекция 2-х кратных ошибок не требует избыточных элементов памяти при наличии I абсолютно-надежного элемента памяти. (Без абсолютно надежного элемента памяти потребовалось бы в данном случае для коррекции 2-х кратной ошибки 2 избыточных элемента памяти).

5. Описанные выше методы построения конечных автоматов с исправлением ошибок, использующие корректирующие коды становятся эффективными при достаточно больших длинах кодов, то есть при достаточно большом числе состояний исходного автомата. В связи с этим целесообразно запрограммировать эти методы на ЦВМ. В /5/ описаны программы для ЦВМ М-20 выбора оптимального разбиения множества входных сигналов, "расщепления" состояний исходного автомата и кодирования состояний "расщепленного" автомата для коррекции ошибок методом, рассмотренным в п. 3. Эти программы содержат соответственно 1150, 178, 422 команд. Программы были проверены для автоматов с числом состояний, не превышающим 100, и числом входов, не превышающих 10. Время реализации этих программ на ЦВМ М-20 для автомата с 100 состояниями и 10 входами в режиме исправления одиночной ошибки составляет соответственно 560 сек, 110 сек, 90 сек.

Литература

1. Карповский М.Г. "Конечные автоматы с обнаружением и исправлением ошибок". "Методы вычислений", вып. 5, 1967, изд. ЛГУ.
2. Питерсон. "Коды, исправляющие ошибки". МИР, 1964.
3. Гаврилов М.А. "Структурная избыточность и надежность релейных устройств". Труды I-го Международного конгресса ИФАК том 3, АН СССР, 1961.
4. Биркгоф С. "Теория структур". ИЛ, М, 1952.
5. Карповский М.Г., Левит И.В., Руханский В.И. "Некоторые машинные алгоритмы кодирования и минимизации автоматов с коррекцией ошибок". "Методы вычислений", вып. 5, 1967, изд. ЛГУ.
6. Yoely "A Cascade Decomposition of Sequential Mashines" Trans IRE, EC-10, N4, 1961.