УДК 510.62

Ю. П. ШАБАНОВ-КУШНАРЕНКО, д-р техн. наук

О ПЕРЕКЛЮЧАТЕЛЬНЫХ ЦЕПЯХ ТЕОРИИ ИНТЕЛЛЕКТА

Переключательные цепи, которые могут быть построены по формулам теории интеллекта [1], существенно отличаются от комбинационных схем, применяемых в современных вычислительных машинах. В комбинационных схемах входная и выходная информация имеет вид двоичных кодов, в переключательных цепях — представлена в форме слов, составленных из букв произвольного алфавита. При использовании переключательных цепей отпадает необходимость в кодировании входной и декодировании выходной информации, что неизбежно при применении комбинационных схем. В отличие от последних в переключательных цепях механизмы ввода и вывода информации включены в состав устройства обработки ее. Эти механизмы составляют единое целое со схемой, осуществляющей преобразование двоичных сигналов. В связи с этим появляется возможность производить минимизацию всего устройства в целом, а не только его центральной части, и за счет этого получать дополнительную экономию оборудования.

При конструировании всего устройства в целом появляются дополнительные возможности для его минимизации [1]. Так, число блоков узнавания во входной (первой) ступени цепи существенно зависит от способа соединения блоков вышележащих ступеней, преобразующих двоичные сигналы. В переключательной цепи, как и в комбинационной схеме, происходит преобразование двоичных сигналов в блоках совпадения и разделения. Однако интерпретация двоичных сигналов, циркулирующих в устройствах этих двух видов, различна. В комбинационных схемах мы рассматриваем сигналы 0 и 1 сами по себе без связи с буквенной информацией. В переключательных же цепях с каждым проводником, воспринимающим или выдающим двоичные сигналы, мы связываем определенное узнавание x^σ, где σ — некоторая буква, а x — какая-либо буквенная переменная. Двоичные сигналы, проходящие по такому проводнику, интерпретируем теперь как значения узнавания x^σ, соответствующего этому проводнику. Если по проводнику проходит сигнал 1, $x^\sigma = 1$, т. е. буквенная переменная x принимает значение σ. Если же по проводнику проходит сигнал 0, $x^\sigma = 0$ и значение переменной x не совпадает с буквой σ, т. е. $x \neq \sigma$.

Таким образом, с содержательной точки зрения переключательные цепи преобразуют не двоичные коды слов, а сами слова, составленные из букв. Следует ожидать, что с точки зрения человека — конструктора, для которого привычным является оперирование со словами, составленными из букв, а не с двоичными кодами этих слов, переключательные цепи будут выглядеть более естественным средством преобразования информации, чем комбинационные схемы. Важно подчеркнуть, что принимая число букв в алфавите алгебры конечных предикатов, равным двум, получаем возможность построить по формулам этой алгебры любую комбинационную схему. Таким образом, перейдя от алгебры логики к алгебре конечных предикатов и от комбинационных схем к переключательным цепям, разработчик вычислительных машин приобретает новые средства конструирования устройств преобразования информации, не теряя при этом ничего из арсенала средств, уже имеющихся в его распоряжении.

Вместе с тем, переключательные цепи, построенные по формулам алгебры конечных предикатов, имеют определенные преимущества также и в сравнении со схемами, которые строят по формулам многозначной логики. Эти схемы (назовем их многозначными) отчасти являются конкурентами переключательных цепей, поскольку они создаются в противовес двоичным комбинационным схемам с целью перехода к автоматической обработке непосредственно буквенной информации. Недостаток многозначных схем состоит в том, что для их построения нужны блоки, преобразующие сами буквы, а не двоичные знаки. Однако многолетние попытки создать такие блоки пока не привели к разработке элементов, которые могли бы успешно конкурировать с элементами, обрабатывающими двоичные сигналы (совпадения, разделения, инверторы и др.).

Переключательные же цепи соединяют в себе, казалось бы, несоединимое: обработку буквенной информации с аппаратурным преобразованием двоичных сигналов, что удобно человеку и приемлемо для вычислительной машины. Любопытно, что уже сегодня стихийно, еще не имея в своем распоряжении аппарата алгебры конечных предикатов, разработчики вычислительных машин используют приемы, которые выглядят гораздо естественнее в рамках алгебры конечных предикатов, чем алгебры логики. К числу таких приемов можно отнести использование дизъюнктивных и конъюнктивных нормальных форм, применение парафазных схем, использование пространственного принципа представления информации. Автор надеется, что переключательные цепи, являющиеся логическим следствием развития теории интеллекта, будут приняты разработчиками электронной техники как естественное средство при конструировании цифровых вычислительных машин новых поколений.

Конкретный вид переключательной цепи, реализующей заданный алфавитный оператор, определяется видом этого оператора, рядом требований, диктуемых практикой. Сформулируем некоторые из этих требований и на конкретном примере рассмотрим, как меняется вид цепи в зависимости от характера требований. Предположим, что требуется построить цепь, преобразующую двоичные цифры x_1, x_2, x_3 в их сумму, записанную смешанным кодом типа $(k_1, k_2, \ldots, k_n)$ [3], причем числа k_1, k_2, $\ldots$, k_n и число n выбираются по усмотрению разработчика цепи. В данном примере возможные следующие три варианта типа кода: (2,2), (2,3), (4); сложение цифр x_1, x_2, x_3 может быть задано тремя различными алфавитными операторами: $y_1 y_2 = F_1(x_1 x_2 x_3)$; $z_1 z_2 = F_2(x_1 x_2 x_3)$; $u = F_3(x_1 x_2 x_3)$. Здесь y_1, y_2, z_1 — двоичные цифры, z_2 — троичная и u — четверичная цифры. Операторы F_1, F_2, F_3 задаем соответственно табл. 1—3.

Таблица 1

x_1	0	0	0	0	1	1	1	1
x_2	0	0	1	1	0	0	1	1
x_3	0	1	0	1	0	1	0	1
y_1	0	0	0	1	0	1	1	1
y_2	0	1	1	0	1	0	0	1

Таблица 2

x_1	0	0	0	0	1	1	1	1
x_2	0	0	1	1	0	0	1	1
x_3	0	1	0	1	0	1	0	1
z_1	0	0	0	0	0	0	0	1
z_2	0	1	1	2	1	2	2	0

Таблица 3

x_1	0	0	0	0	1	1	1	1
x_2	0	0	1	1	0	0	1	1
x_3	0	1	0	1	0	1	0	1
u	0	1	1	2	1	2	2	3

Способом, описанным в работе [1], переходим от таблиц к уравнениям, явно задающим оператор F_1:

$$x_1^0 x_2^0 x_3^0 \vee x_1^0 x_2^0 x_3^1 \vee x_1^0 x_2^1 x_3^0 \vee x_1^1 x_2^0 x_3^0 = y_1^0; \;\; x_1^0 x_2^1 x_3^1 \vee x_1^1 x_2^0 x_3^1 \vee x_1^1 x_2^1 x_3^0 \vee$$
$$\vee\, x_1^1 x_2^1 x_3^1 = y_1^1; \;\; x_1^0 x_2^0 x_3^0 \vee x_1^0 x_2^1 x_3^1 \vee x_1^1 x_2^0 x_3^1 \vee x_1^1 x_2^1 x_3^0 = y_2^0; \;\; x_1^0 x_2^0 x_3^1 \vee$$

$$\vee\ x_1^0x_2^1x_3^0 \vee x_1^1x_2^0x_3^0 \vee x_1^1x_2^1x_3^1 = y_2^1; \tag{1}$$

оператор F_2:

$$x_1^0x_2^0x_3^0 \vee x_1^0x_2^0x_3^1 \vee x_1^0x_2^1x_3^0 \vee x_1^1x_2^0x_3^0 \vee x_1^0x_2^1x_3^1 \vee x_1^1x_2^0x_3^1 \vee x_1^1x_2^1x_3^0 = z_1^0;$$

$$x_1^1x_2^1x_3^1 = z_1^1;\ x_1^0x_2^0x_3^0 \vee x_1^1x_2^1x_3^1 = z_2^0;\ x_1^0x_2^0x_3^1 \vee x_1^0x_2^1x_3^0 \vee x_1^1x_2^0x_3^0 = z_2^1;$$

$$x_1^0x_2^1x_3^1 \vee x_1^1x_2^0x_3^1 \vee x_1^1x_2^1x_3^1 = z_2^2; \tag{2}$$

оператор F_3:

$$x_1^0x_2^0x_3^0 = u^0;\ x_1^0x_2^0x_3^1 \vee x_1^0x_2^1x_3^0 \vee x_1^1x_2^0x_3^0 = u^1;$$

$$x_1^0x_2^1x_3^1 \vee x_1^1x_2^0x_3^1 \vee x_1^1x_2^1x_3^0 = u^2;\ x_1^1x_2^1x_3^1 = u^3. \tag{3}$$

Когда требуются цепи с максимальным быстродействием, целесообразно применять трехступенчатые переключательные цепи со многими выходами. Поскольку эти цепи имеют наименьшее возможное число ступеней, время, необходимое для прохождения сигналов по ним, будет минимальным. Построив по полученным формулам методом, изложенным в работе [1], трехступенчатые цепи, мы получили бы для каждой из них в первой ступени по 6 блоков узнавания, во второй — по 8 блоков совпадения с суммарным числом входов 24. В третьей ступени цепи для оператора F_1 мы имели бы 4 блока разделения с 16 входами, для оператора F_2 — 4 блока разделения с 15 входами, для оператора F_3 — 2 блока разделения с 6 входами.

Как видим, сложность переключательной цепи, реализующей некоторое преобразование информации, существенно зависит от способа кодирования выходных слов этого преобразования. В нашем примере при коде типа (2,2) для построения цепи требуются 40 диодов, при коде типа (2,3) — 39, при коде типа (4) — 30. Поэтому варьируя тип кода выходных слов, в ряде случаев можно получить дополнительные упрощения в схеме переключательной цепи. Интересной задачей, ждущей своего решения, является получение асимптотических оценок сложности цепей в зависимости от выбора типа кодирования информации. Можно предположить, что при уменьшении числа n будут, как правило, получаться более экономные цепи. Однако при этом может наблюдаться усложнение выходных устройств, преобразующих значения узнаваний букв в сами буквы.

Рассмотрим некоторые из вариантов переключательных цепей, которые могут быть получены в обсуждаемом примере при типе кода выходных слов (2,2). Аналогичные рассмотрения могут быть проведены также и для типов кода (2,3) и (4). В некоторых практических приложениях более важным может оказаться требование минимальности стоимости цепи, достигаемой за счет снижения ее быстродействия. В этом случае целесообразно остановить выбор на многоступенчатой переключательной цепи. Отправляясь от уравнений (1) и пользуясь описанными методами построения многоступенчатых безынверторных цепей и цепей со многими выходами [1], приходим к пятиступенчатой схеме. В ней число блоков узнавания осталось таким же, как и в трехступенчатой схеме (6), число же диодов несколько уменьшилось (34 вместо 40). Экономия в числе диодов достигнута ценой увеличения числа ступеней (5 вместо 3), т. е. в конечном итоге за счет снижения быстродействия цепи.

Важно обратить внимание на то, что обсуждаемая переключательная цепь сама формирует область $\{0,1\}$ изменения своих входных сигналов x_1, x_2, x_3, поэтому по входным каналам цепи разрешается подавать не только символы **0** и **1**, но и любые другие буквы. Если хотя бы по одному из трех входных каналов поступит буква, отличная от **0** или **1**, цепь в ответ на это на всех своих выходных каналах $\overset{0}{y_1}$, $\overset{1}{y_1}$, $\overset{0}{y_2}$, $\overset{1}{y_2}$ выработает нулевые значения, сигнализируя о неприемлемости входной информации.

Если заранее известно, что на входы цепи будут подаваться только допустимые буквы (в нашем примере — только **0** и **1**), переключательную цепь можно дополнительно упростить за счет использования усеченных законов истинности. Для нашего примера имеют место следующие усеченные законы истинности:

$$\overset{0}{x_1} \vee \overset{1}{x_1} = 1, \; \overset{0}{x_2} \vee \overset{1}{x_2} = 1, \; \overset{0}{x_3} \vee \overset{1}{x_3} = 1. \tag{4}$$

С учетом равенств (4), вводя промежуточные переменные $t_1 \div t_{10}$, систему уравнений (1) преобразуем к виду

$$t_1 = \overset{0}{x_1}\overset{0}{x_2}; \; t_2 = \overset{0}{x_1}\overset{1}{x_2}; \; t_3 = \overset{1}{x_1}\overset{0}{x_2}; \; t_4 = \overset{1}{x_1}\overset{1}{x_2}; \; t_5 = t_2 \vee t_3;$$

$$t_6 = t_1 \vee t_4; \; t_7 = \overset{0}{x_3}t_5; \; t_8 = \overset{1}{x_3}t_5; \; t_9 = \overset{0}{x_3}t_6; \; t_{10} = \overset{1}{x_3}t_6; \; \overset{0}{y_1} = t_1 \vee t_7;$$

$$\overset{1}{y_1} = t_4 \vee t_8; \; \overset{0}{y_2} = t_8 \vee t_9; \; \overset{1}{y_2} = t_7 \vee t_{10}. \tag{5}$$

По уравнениям (5) строим переключательную цепь (рис. 1). Она, как и предыдущая, имеет 5 ступеней и 6 блоков узнавания, однако для ее построения требуется меньшее число диодов (28 вместо 34).

Во всех трех рассмотренных случаях информация о выходных сигналах y_1 и y_2 формируется переключательными цепями с избытком. Очевидно, что достаточно было бы сформировать на выходах цепей лишь сигналы $\overset{1}{y_1}$, $\overset{1}{y_2}$ (или $\overset{0}{y_1}$, $\overset{0}{y_2}$, или $\overset{0}{y_1}$, $\overset{1}{y_2}$, или $\overset{1}{y_1}$, $\overset{0}{y_2}$), чтобы получить полную информацию о значениях переменных y_1 и y_2. Такая избыточность для некоторых применений может оказаться полезной, так как повышает надежность работы цепи, позволяя обнаруживать сбои в ее работе. В наборах сигналов $(\overset{0}{y_1}, \overset{1}{y_1})$ и $(\overset{0}{y_2}, \overset{1}{y_2})$ при нормальной работе цепи всегда должны быть как единица, так и нуль. Если же в ней произошел сбой, исказивший один из выходных сигналов, это приведет к появлению недопустимых наборов значений (**0**, **0**) или (**1**, **1**) на выходах цепи. В том случае, когда требования, предъявляемые к цепи таковы, что избыточность формируемой ею информации не нужна, возможны дополнительные упрощения переключательной цепи. В нашем примере, сохраняя в схеме

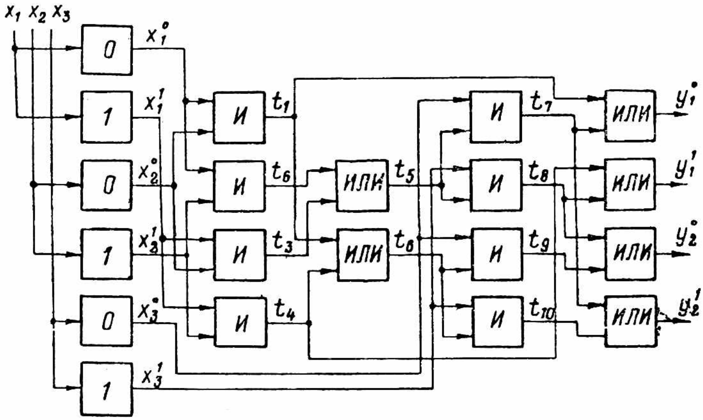

Рис. 1

лишь два выхода: y_1^1 и y_2^1, получаем пятиступенчатую переключательную цепь, содержащую 6 блоков узнавания и 22 диода (рис. 2). Цепь построена по следующим уравнениям:

$$t_1 = x_1^0 x_2^0;\ \ t_2 = x_1^0 x_2^1;\ \ t_3 = x_1^1 x_2^0;\ \ t_4 = x_1^1 x_2^1;\ \ t_5 = t_2 \vee t_3;\ \ t_6 = t_1 \vee t_4;\ \ t_7 = x_3^0 t_5;\ \ t_8 = x_3^1 t_5;\ \ t_9 = x_3^1 t_6;\ \ y_1^1 = t_4 \vee t_8;\ \ y_2^1 = t_7 \vee t_9. \tag{6}$$

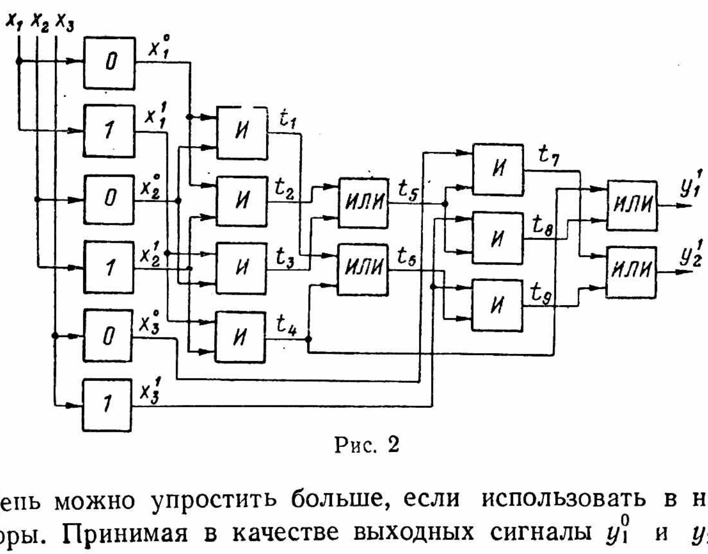

Рис. 2

Цепь можно упростить больше, если использовать в ней инверторы. Принимая в качестве выходных сигналы y_1^0 и y_2^1, получаем схему, содержащую 6 ступеней, 6 блоков узнавания, 14 диодов и 1 инвертор (рис. 3) и описываемую уравнениями:

$$t_1 = x_1^0 x_2^0; \quad t_2 = x_1^1 x_2^1; \quad t_3 = t_1 \vee t_2; \quad t_4 = x_3^0 \overline{t_3};$$
$$y_1^0 = t_1 \vee t_4; \quad y_2^1 = x_3^1 t_3 \vee t_4. \tag{7}$$

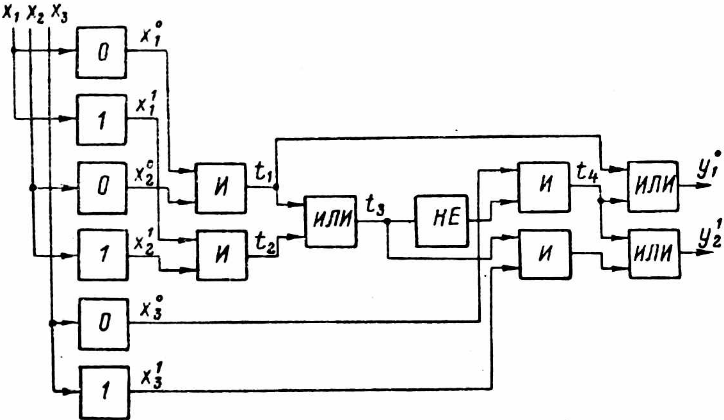

Рис. 3

Другой вариант экономной цепи получаем, выбирая в качестве выходных сигналы y_1^1 и y_2^1. В ней используются 7 ступеней, 3 блока узнавания, 16 диодов и 1 инвертор (рис. 4). Цепь построена по следующим уравнениям:

$$t_1 = x_1^0 x_2^0; \quad t_2 = x_1^1 x_2^1; \quad y_2^1 = x_3^1 t_1 \vee t_2; \quad y_1^1 = (x_3^1 \vee t_1)\, y_2^1 \vee x_3^1 t_2. \tag{8}$$

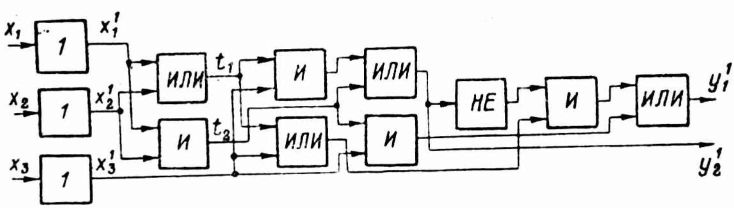

Рис. 4

Рассмотрены примеры переключательных цепей различной сложности, реализующих одну и ту же операцию сложения двоичных цифр. Эти примеры могут служить иллюстрацией того общего положения, что сложность цепи определяется не только видом заданного преобразования информации, но также и различными практическими требованиями, предъявляемыми к переключательной цепи. К числу таких требований относятся: максимальное быстродействие цепи; минимальная сложность ее; максимальная надежность работы; необходимость селекции допустимых букв из множества всевозможных букв, поступающих на входы цепи. В распоряжении разработчика имеется ряд приемов конструирования переключательных цепей, с помощью которых он может в той или иной степени удовлетворить упомянутым требованиям. К числу подобных приемов можно отнести следующие: варьирование типом кода выходных слов, числом ступеней цепи, блоков узнавания и диодов; введение в цепь инверторов и варьирование их числом; использование минимальных дизъюнктивных и конъюнктивных нормальных форм, скобочных форм, приемов построения цепей со многими выходами; введение или устранение избыточности при передаче информации по цепи.

Список литературы: 1. *Шабанов-Кушнаренко Ю. П.* Аналитический метод явного задания конечных алфавитных операторов.— В кн.: Проблемы бионики, Харьков. 1980, вып. 25, с. 25—33. 2. *Шабанов-Кушнаренко Ю. П.* Об универсальной алгебре конечных предикатов. — В кн.: АСУ и приборы автоматики. Харьков, 1980, вып. 55, с. 69—74.

Поступила 26 марта 1979 г.