

ЛОГИЧЕСКОЕ ПРОЕКТИРОВАНИЕ ДИСКРЕТНЫХ АВТОМАТОВ

УДК 518.5

НАХОЖДЕНИЕ РЕЖИМА МАКСИМАЛЬНОГО ЭНЕРГОПОТРЕБЛЕНИЯ ЛОГИЧЕСКОЙ СХЕМЫ

А. Д. Закревский

*Объединённый институт проблем информатики НАН Беларуси, г. Минск, Беларусь***E-mail:** zakrevskij@tut.by

Известно, что оценка максимального энергопотребления логической схемы позволяет обеспечить её надёжность. Получение этой оценки облегчается предварительным нахождением такого режима работы схемы, при котором её энергопотребление оказывается максимальным. Предлагается метод нахождения этого режима для случая, когда рассматриваемая схема реализует конечный автомат.

Ключевые слова: логические схемы, режим максимального энергопотребления, логическое проектирование.

Введение

При проектировании логической схемы важно оценивать максимальную затрату энергии при её функционировании, поскольку чрезмерные затраты энергии могут приводить к выходу схемы из строя. В основном эта энергия тратится при смене входных наборов — комбинаций значений входных булевых переменных схемы. Затраты энергии для каждой пары входных наборов (предшествующего и последующего) можно находить экспериментально или оценивать числом переключаемых транзисторов [1, 2].

Оценивание энергопотребления логической схемы может проводиться более успешно при учёте назначения схемы и способа её проектирования. Это позволяет значительно сократить число рассматриваемых входных наборов и переходов между ними, облегчая тем самым решение данной задачи. Ещё больший эффект достигается при предварительном нахождении режима максимального энергопотребления — соответствующей последовательности наборов значений входных переменных схемы.

Ниже эта задача решается для случая, когда схема синхронна и реализует конечный автомат. Тогда она представляет собой комбинационную схему с одноканальной обратной связью, обеспечиваемой регистром. Значения переменных на отмеченных жирными линиями выходных полюсах в текущий момент времени становятся значениями соответствующих входных переменных в следующий такт. Пример такой схемы показан на рис. 1.

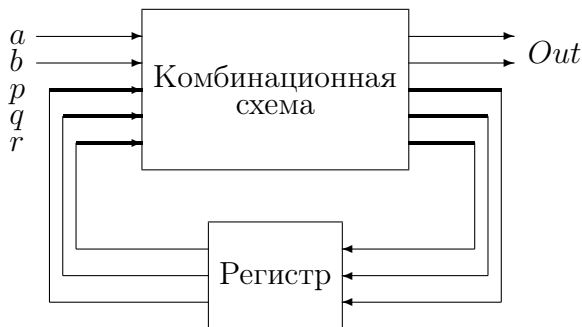


Рис. 1. Схемная реализация конечного автомата

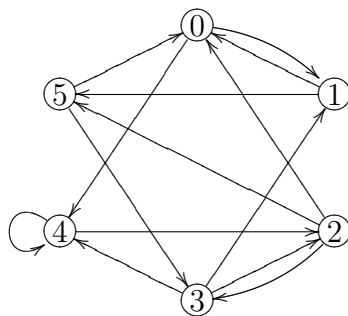
Положим, что комбинационная схема имеет n входных переменных и они делятся на две части: u свободных переменных, принимающих произвольные наборы значений, и v связанных переменных, наборы значений которых представляют коды возможных состояний автомата ($u + v = n$). В примере на рис. 1 свободными являются переменные a, b , а связанными — переменные p, q, r .

Число w входных наборов схемы в целом равно $s2^u$, где s — число состояний автомата. Эти наборы представляются соответствующими значениями n -компонентного булева вектора $\mathbf{x} = (\mathbf{u}, \mathbf{v})$. Вектор \mathbf{u} представляет набор значений свободных переменных и содержит u компонент. Вектором \mathbf{v} представляется набор значений связанных компонент. Его длина v определяется числом состояний автомата и способом их кодирования.

Предлагаемый в работе метод нахождения режима максимального энергопотребления логической схемы основан на предположении, что чем больше переменных меняют свои значения при очередной смене значения вектора \mathbf{x} , тем больше тратится энергии на эту смену.

1. Пример конечного автомата

Рассмотрим в качестве примера автомат с шестью состояниями $S_0, S_1, S_2, S_3, S_4, S_5$ и двумя свободными переменными a и b . Возможные переходы между состояниями отобразим ориентированным графом переходов G (рис. 2).

Рис. 2. Граф переходов G

Условия переходов между состояниями S_i и S_k представим соответствующими элементами таблицы. Например, автомат переходит из состояния S_1 в состояние S_0 только при одном наборе значений свободных переменных — когда $ab = 1$, а в состояние S_5

в трёх случаях — когда $\bar{a} \vee \bar{b} = 1$. Заметим, что пустые элементы таблицы соответствуют невозможным переходам.

Условия переходов в автомате

	S_0	S_1	S_2	S_3	S_4	S_5
S_0		a			\bar{a}	
S_1	ab					$\bar{a} \vee \bar{b}$
S_2	ab			$a\bar{b}$		\bar{a}
S_3		ab	\bar{b}		$\bar{a}\bar{b}$	
S_4			b		\bar{b}	
S_5	$a\bar{b}$			$\bar{a} \vee b$		

Проектирование логической схемы, реализующей конечный автомат, начинается с кодирования его состояний наборами значений булевых переменных. Практически эффективные алгоритмы решения этой задачи рассмотрены в [3]. В работе [4] данная задача решается путём такого отображения графа переходов в булев граф, при котором как можно больше рёбер графа переходов отображается на рёбра булева графа. В результате существенно уменьшается число переменных, меняющих свои значения на переходах, и тем самым снижается энергопотребление схемы, реализующей автомат. Состояния автомата кодируются булевыми векторами, соответствующими тем вершинам булева графа, на которые они отобразились.

Так, в данном примере находятся коды состояний автомата, компонентами которых служат соответствующие значения трёх булевых переменных p, q, r . Эти коды представлены столбцами следующей матрицы:

$$\begin{array}{cccccc}
 & S_0 & S_1 & S_2 & S_3 & S_4 & S_5 \\
 p & 1 & 0 & 1 & 0 & 1 & 0 \\
 q & 0 & 0 & 1 & 1 & 1 & 0 \\
 r & 0 & 0 & 0 & 1 & 1 & 1
 \end{array}$$

В полученном результате при большинстве переходов в данном автомате меняется значение лишь одной из кодирующих переменных p, q или r — соответствующие рёбра отмечены на рис. 3 одиночными линиями. При остальных переходах меняются значения двух либо трёх переменных — рёбра представлены двойными либо тройными линиями.

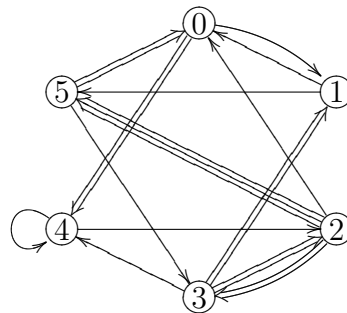


Рис. 3. Граф переходов с отмеченными рёбрами

Вместе с двумя входными переменными кодирующие переменные образуют пятикомпонентный входной булев вектор $\mathbf{x} = (a, b, p, q, r)$ комбинационной схемы, реализующей заданный автомат.

В данном примере автомата с шестью состояниями число возможных входных наборов схемы $w = 6 \cdot 2^2 = 24$, что несколько меньше числа всех наборов $2^5 = 32$. Значительно сильнее сокращается число переходов между наборами: из 1024 различных переходов оказываются возможными лишь $16 \cdot 14 = 224$. Действительно, число переходов в автомате равно 14 и при каждом из них значения переменных p, q, r меняются однозначно. Что же касается входных переменных a и b , то они могут принимать произвольные наборы значений в предшествующий и последующий такты.

2. Нахождение режима максимального энергопотребления схемы

Предлагаемый метод заключается в поиске такой циклической последовательности переходов, которая допустима в рамках используемой формальной модели (хотя и может быть маловероятной), может многократно повторяться и является наиболее энергоёмкой, приводя к максимальным энергозатратам. При решении этой задачи можно ограничиться рассмотрением *простых* (не самопересекающихся) циклов, поскольку любой цикл разлагается на простые и его энергоёмкость не может превысить максимальной энергоёмкости элементов разложения.

Например, простым является цикл из трёх переходов $0 \rightarrow 4, 4 \rightarrow 2$ и $2 \rightarrow 0$ (см. рис. 3), который проходит через вершины 0, 4, 2 и обозначается 042. Для нашего примера простых циклов 15: 4, 01, 015, 01532, 015342, 042, 04231, 042315, 0425, 153, 042531, 23, 234, 253, 2534.

Для входящих в простые циклы переходов подсчитываются их веса — числа кодирующих переменных, меняющих свои значения. Среднее значение такого числа на переход (обозначим это среднее через *mean_code*) определяет энергоёмкость цикла. Выбирается цикл, в котором оно максимально. Например, в цикле 042 на переходах $0 \rightarrow 4, 4 \rightarrow 2$ и $2 \rightarrow 0$ меняют свои значения соответственно две, одна и одна переменная; следовательно, $mean_code = (2 + 1 + 1)/3 = 4/3$.

Максимальное значение *mean_code*, равное 2, достигается в циклах 2504, 23 и 325. Заметим, что эти циклы, содержащие переходы с большими весами, можно выявить визуально (см. рис. 3), используя эвристический метод, иллюстрируемый следующим примером.

Берется переход 25 с максимальным весом (3), к нему подсоединяется соседний переход 50 с весом 2, затем переход 04, также с весом 2. В результате получается композиция трёх переходов, последовательно проходящих через вершины 2, 5, 0, 4. Её можно замкнуть переходом 42 с весом 1, получив таким образом цикл 2504. Аналогично находится цикл 325: в этом случае к переходу 25 подсоединяется соседний слева переход 32, затем цикл замыкается переходом 53.

Для каждого перехода в выбранном цикле находятся допустимые наборы значений свободных переменных, т. е. такие наборы, которые обеспечивают выполнение перехода. Например, для переходов $2 \rightarrow 5, 5 \rightarrow 0, 0 \rightarrow 4$ и $4 \rightarrow 2$ цикла 2504 такие наборы переменных a, b образуют соответственно множества $\{00, 01\}, \{10\}, \{00, 01\}$ и $\{01, 11\}$. Последовательность этих множеств представим выражением

$$free = 00, 01/10/00, 01/01, 11.$$

Из допустимых наборов выбираются такие, при которых по возможности максимизируется число переменных, меняющих свои значения на переходе. В данном примере такие наборы образуют следующую последовательность значений вектора \mathbf{u} : 01, 10, 01, 11. Объединяя полученные наборы свободных переменных с соответствующими наборами кодирующих переменных — последовательными значениями вектора \mathbf{v} : 110,

001, 100, 111, находим искомую циклическую последовательность наборов всех входных переменных схемы — последовательность соответствующих значений вектора \mathbf{x} : 01110, 10001, 01100, 11111.

В заключение подсчитываются числа входных переменных комбинационной схемы, меняющих свои значения на переходах, и находится их среднее значение:

$$mean_input = (5 + 4 + 3 + 2)/4 = 3,5.$$

Аналогично подсчитываются энергозатраты при двух других выбранных циклах:

$$\begin{aligned} 23 : \quad & free = 10/00, 10; \\ & \mathbf{u} : 10, 00; \quad \mathbf{v} : 110, 011; \quad \mathbf{x} : 10110, 00011; \\ & mean_input = (3 + 3)/2 = 3; \\ 325 : \quad & free = 00, 10/00, 01/00, 01, 11; \\ & \mathbf{u} : 00, 01, 11; \quad \mathbf{v} : 011, 110, 001; \quad \mathbf{x} : 00011, 01110, 11001; \\ & mean_input = (3 + 4 + 3)/3 = 3,33\dots \end{aligned}$$

Как видно, максимальной энергоёмкостью обладает цикл 2504. Следовательно, режим максимального энергопотребления рассматриваемой схемы заключается в периодическом повторении простого цикла 2504, который инициируется повторяющейся четвёркой (01, 10, 01, 11) наборов значений свободных переменных a и b .

Заключение

Задача определения максимального энергопотребления спроектированной логической схемы упрощается, когда известно назначение схемы — например, когда она реализует конечный автомат. Известные методы её решения, основанные на подсчёте числа переключаемых логических элементов, довольно трудоёмки, будучи связаны с рассмотрением многочисленных вариантов последовательной смены значений входного вектора. Объём вычислений, производимый при решении этой задачи, значительно сокращается предлагаемым в данной работе методом нахождения режима максимального энергопотребления — соответствующей циклической последовательности значений входного вектора.

ЛИТЕРАТУРА

1. Ghosh A., Devadas S., Keutzer K., and White J. Estimation of Average Switching Activity in Combinational and Sequential Circuits // Proc. 29th ACM/IEEE Design Automation Conf. Anaheim, CA, 1992. P. 253–259.
2. Marculesku D., Marculesku R., and Pedram M. Theoretical Bounds for Switching Activity Analysis in Finite-State Machines // Proc. 1998 international symposium on low power electronics and design. NY, USA: ACM, 1998. P. 36–41.
3. Закревский А. Д. Алгоритмы энергосберегающего кодирования состояний автомата // Информатика. 2011. № 1(29). С. 68–78.
4. Закревский А. Д. Алгоритм матричного отображения графа на булев куб // Вестник Томского госуниверситета. Управление, вычислительная техника и информатика. 2011. № 3(16). С. 94–99.