EESTI NSV TEADUSTE AKADEEMIA TOIMETISED. 23. KÖIDE
FÜÜSIKA * MATEMAATIKA. 1974, NR. 1

ИЗВЕСТИЯ АКАДЕМИИ НАУК ЭСТОНСКОЙ ССР. ТОМ 23.
ФИЗИКА * МАТЕМАТИКА. 1974, № 1

https://doi.org/10.3176/phys.math.1974.1.03

УДК 681.142.01

*А. СИЙМОН*

# О МЕТОДЕ ОБРАЗОВАНИЯ ВЕКТОРНО-ВРЕМЕННЫХ ПЕРЕКЛЮЧАТЕЛЬНЫХ ФУНКЦИЙ ОТ МИКРОПРОГРАММ

В работе [1] был разработан метод образования векторно-временных переключательных функций (ВП-функций) [2–8], но некоторые проблемы, поставленные и решенные там, допускают дальнейшие уточнения и расширения.

Сделаем одно замечание относительно исходных микропрограмм. В [1] исходные микропрограммы в регулярной записи были рассмотрены в форме, предложенной в [9]. Однако учет конечности регистров позволяет упростить эту форму записи микропрограмм, так как иногда некоторые микрооперации оказываются лишними из-за того, что результат их применения может быть получен как побочный эффект других микроопераций. Такая ситуация возникает, например, в микропрограмме [9]

$$Q=O_2\{(e\vee s_{12}p_3^{-1})\,l_1r_3\}_{\alpha_3\beta_3}O_1O_3, \tag{1}$$

которая представляет собой умножение целых неотрицательных чисел. В (1) применены следующие микрооперации и условия:

$e$ — тождественное преобразование;
$s_{12}$ — прибавление содержимого первого регистра ко второму регистру;
$p_3^{-1}$ — вычитание единицы из содержимого третьего регистра;
$l_1$ — левый сдвиг на один разряд в первом регистре;
$r_3$ — правый сдвиг на один разряд в третьем регистре;
$O_i$ — очистка (установка в нуль) $i$-го регистра;
$\alpha_3$ — условие равенства нулю содержимого третьего регистра;
$\beta_3$ — условие равенства нулю младшего разряда третьего регистра.

В данном случае, если количество разрядов во всех регистрах одинаково, как в работе [1], то в микропрограмме (1) лишней оказывается микрооперация $p_3^{-1}$, так как ее функции выполняет другая микрооперация $r_3$ за счет «ликвидации» младшего разряда за пределы регистра.

Назовем операцию удаления лишних микроопераций операцией приведения микропрограмм к регулярной записи, а форму полученных микропрограмм в регулярной записи — приведенной формой микропрограмм в регулярной записи. В данном случае приведенная форма микропрограммы (1) в регулярной записи будет иметь вид

$$Q=O_2\{(e\vee s_{12})\,l_1r_3\}_{\alpha_3\beta_3}O_1O_3. \tag{2}$$

Применение приведенных форм в регулярной записи микропрограмм, как исходных для образования ВП-функций, позволяет получить более экономичные схемы дискретных автоматов и иногда увеличить их быстродействие.

В [1] для определения временных координат существования установочных и управляющих сигналов был предложен графический метод — временная диаграмма. В данной работе приведем другой метод, который является модификацией известного в литературе метода образования таблицы переходов для конечных автоматов от соответствующих регулярных выражений [10]. С этой целью рассмотрим микрооперации микропрограммы приведенной формы в регулярной записи как буквы входного алфавита автомата [10] и в общем случае обозначим их через $m$. Если в одной и той же микропрограмме одна и та же микрооперация $m$ повторяется несколько раз, то для приведенной ниже процедуры целесообразно каждое вхождение этой микрооперации $m$ интерпретировать как появление новой микрооперации $m^{(\varepsilon)}$, где $\varepsilon$ — порядковый номер вхождения микрооперации $m$ в данную микропрограмму. Понятия мест — основных, предосновных, начальных и конечных — в микропрограмме регулярной записи определим так, как они определены в [10] для регулярных выражений. Применяемые в данной работе понятия основной и неосновной индекс также совпадают с соответствующими понятиями из [10].

Обозначим условие, написанное под открывающейся итерационной скобкой в микропрограмме регулярной записи, через $\alpha_j$, где $j$ — порядковый номер соответствующего регистра. Обозначим условие, написанное под открывающейся обычной скобкой в микропрограмме регулярной записи, через $\beta_i$, где $i$ — порядковый номер соответствующего регистра, а сами эти обычные скобки назовем обычными скобками с условием.

Назовем в микропрограмме регулярной записи обобщенным термом всякое сочетание микроопераций, где самой внешней операцией является произведение, и назовем обобщенным многочленом всякое сочетание микроопераций, где самой внешней операцией является дизъюнкция. Например, микропрограммы (1) и (2) являются обобщенными термами.

На данном этапе синтеза целесообразно считать, что для каждой микропрограммы в регулярной записи синтезируется свой конечный подавтомат $A_Q$, который функционирует в своем собственном дискретном времени $t_k$, где $k = 0, 1, 2 \ldots$

Определим правила подчинения мест следующим образом:

1. Все начальные места обобщенного многочлена, точнее двучлена, заключенного в обычные скобки с условием, подчиняются месту, находящемуся непосредственно перед открывающейся скобкой. Этому же месту подчиняется начальное место первого обобщенного терма из данного обобщенного многочлена при выполнении условия $\beta_i$, прикрепленного к этой открывающейся скобке, и начальное место второго обобщенного терма из данного обобщенного многочлена — при выполнении условия $\overline{\beta}_i$.

2. Место, находящееся непосредственно справа от закрывающейся обычной скобки с условием, подчиняется всем конечным местам обобщенных термов из данного обобщенного многочлена, заключенного в эти обычные скобки с условием.

3. Начальное место обобщенного терма, заключенного в итерационные скобки, при выполнении условия $\overline{\alpha}_j$, где условие $\alpha_j$ прикреплено к

этим итерационным скобкам, подчиняется месту, находящемуся непосредственно слева от открывающейся итерационной скобки.

4. Начальное место обобщенного терма, заключенного в итерационные скобки, при выполнении условия $\overline{\alpha}_j$ подчиняется конечному месту данного обобщенного терма.

5. Место, находящееся непосредственно справа от закрывающейся итерационной скобки, при выполнении условия $\alpha_j$, прикрепленного к этим итерационным скобкам, подчиняется месту, находящемуся перед этой открывающейся итерационной скобкой.

6. Место, находящееся непосредственно перед открывающейся итерационной скобкой, при выполнении условия $\alpha_j$, прикрепленного к этим итерационным скобкам, подчиняется конечному месту обобщенного терма, заключенного в эти итерационные скобки.

7. Если место $\delta$ подчиняется месту $\varepsilon$ и место $\varepsilon$ подчиняется месту $\eta$, то и место $\delta$ подчиняется месту $\eta$.

8. Каждое место подчиняется самому себе.

9. Других правил подчинения мест не существует.

Приведем ниже алгоритм, который даст пару $m\tau^*_{t_k}$ с учетом максимального совмещения выполняемых микроопераций во времени, где $\tau^*_{t_k}$ — управляющий или установочный импульсный сигнал, поступающий из управляющей части автомата во время $t_k$. Кроме того, для потенциальной элементной структуры [11] $\tau^*_{t_k}$ является сигналом, формируемым серией тактных сигналов первого полупериода тактов.

Введем следующие обозначения:

$$\xi=\begin{cases}\tilde{\alpha}_j\tilde{\beta}_i\\ \tilde{\alpha}_j\\ \tilde{\beta}_i\end{cases}; \quad \tilde{\alpha}_j=\begin{cases}\alpha_j\\ \overline{\alpha}_j\end{cases}; \quad \tilde{\beta}_i=\begin{cases}\beta_i\\ \overline{\beta}_i\end{cases}.$$

Сам алгоритм определения пар $m\tau^*_{t_k}$ для микропрограммы в регулярной записи имеет следующий вид:

1. Отметить места в выражениях микропрограмм регулярной записи вертикальными черточками.

2. Присвоить этим местам, слева направо, порядковые номера $\iota$, где $\iota = 0, 1, 2, \ldots$, записав их в качестве основных индексов под вертикальными черточками обозначения мест. Начальному месту присвоить порядковый номер $\iota = 0$.

3. К основному индексу начального места первого обобщенного терма обобщенного двучлена, заключенного в обычные скобки с условием $\beta_i$, приписать $\overline{\beta}_i$, а к основному индексу начального места второго обобщенного терма обобщенного двучлена, заключенного в обычные скобки с условием $\beta_i$, приписать $\beta_i$.

4. К основному индексу места, находящегося непосредственно справа от закрывающейся итерационной скобки, приписать условие $\alpha_j$, прикрепленное к данным итерационным скобкам.

5. Определить подчинение указанных мест по соответствующим правилам и написать основные индексы мест, которым данное место подчиняется, в качестве неосновных индексов под вертикальной черточкой обозначения места, которое подчиняется данному месту. Если подчинение мест происходит с условием, то и обозначение условия приписать к этому неосновному индексу.

6. К неосновному индексу $\iota_1$ места, находящегося непосредственно справа от закрывающейся обычной скобки с условием, приписать $\overline{\beta'}_i$, где $\iota_1$ — основной индекс конечного места первого обобщенного терма обобщенного двучлена, заключенного в обычные скобки с условием $\beta_i$. К неосновному индексу $\iota_2$ места, находящегося непосредственно справа от закрывающейся обычной скобки, приписать $\beta'_i$, где $\iota_2$ — основной индекс конечного места второго обобщенного терма обобщенного двучлена, заключенного в обычные скобки с условием $\beta_i$. Условия $\overline{\beta'}_i$ и $\beta'_i$ являются соответственно значениями $\overline{\beta}_i$ и $\beta_i$ перед заходом в эти обычные скобки с условием $\beta_i$.

7. Обвести индексы (как основные, так и неосновные) предосновных мест общей рамкой.

8. Взять $k = 0$.

9. Взять $\iota = 0$.

10. Найти предосновное место с основным или неосновным индексом $\iota$ или $\xi\iota$ (в последнем случае должно выполняться условие $\xi$) и перейти к п. 11. Если такого предосновного места не найдется, то перейти к п. 17.

11. Если найденному предосновному месту с индексом $\iota$ или $\xi\iota$ непосредственно следует микрооперация $m \neq e$, то перейти к п. 12. В противном случае перейти к п. 15.

12. Если во время $t_k$ при какой-то микрооперации $m_1$ совершается какое-то действие над тем регистром, над которым совершается микрооперация $m$, или для какого-то условия $\alpha_j$ или $\beta_i$ (но не $\overline{\beta'}_i$ и $\beta'_i$) снимается информация с данного регистра, то перейти к п. 13. В противном случае перейти к п. 14.

13. Увеличить значение $k$ на «1».

14. Выписать пару $m\tau^*_{t_k}$ и перейти к п. 16.

15. Если найденному месту с индексом $\iota$ или $\xi\iota$ непосредственно следует микрооперация $m = e$, то пару $m\tau^*_{t_k}$ не выписывать.

16. Если микрооперации $m$ соответствует основное место с основным индексом $\iota_3$ или $\xi\iota_3$, то взять $\iota = \iota_3$ и перейти к п. 10.

17. Увеличить значение $\iota$ на «1».

18. Если существует место с основным индексом $\iota$, то перейти к п. 10. В противном случае перейти к п. 19.

19. Конец данного алгоритма.

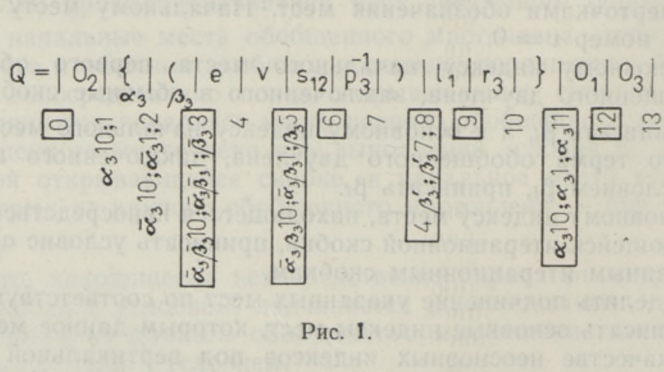


Рис. 1.

Для демонстрации приведенного выше алгоритма рассмотрим микропрограммы вида (1) и (2), изображенные соответственно на рис. 1 и 2.

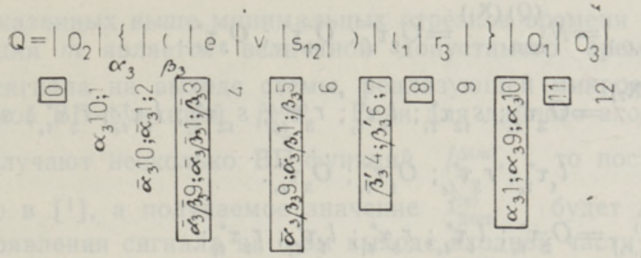


Рис. 2.

Ради простоты предполагаем, что все регистры в данном примере 3-разрядные.

Назовем последовательность пар $m\tau^*_{t_k}$, упорядоченных по приведенному выше алгоритму, — путем.

Назовем путь минимальным относительно множества микроопераций $\mathfrak{R}$, если при его образовании опускается максимальное количество микроопераций (соответствующим выбором значений $\alpha_j$ и $\beta_i$). Обозначим минимальный путь символом $N_{\min}^{(Q)(\mathfrak{R})}$.

Назовем путь максимальным относительно $\mathfrak{R}$, если при его образовании охвачено максимальное количество микроопераций $m \in \mathfrak{R}$ (соответствующим выбором значений $\alpha_j$ и $\beta_i$). Обозначим максимальный путь символом $N_{\max}^{(Q)(\mathfrak{R})}$.

Назовем путь укороченным относительно микрооперации $m$ при следующих условиях его образования:

1) если микрооперация $m$ находится в итерационных скобках, то пара $m\tau^*_{t_k}$ повторяется два раза, чтобы количество остальных пар $m_4\tau^*_{t_k}$ перед парой $m\tau^*_{t_k}$ было минимальным ($m_4$ какая-то микрооперация, не равная $m$);

2) если микрооперация $m$ находится в обычных скобках с условием $\beta_i$ или вне скобок, то пара $m\tau^*_{t_k}$ повторяется один раз, чтобы количество остальных пар $m_4\tau^*_{t_k}$ перед парой $m\tau^*_{t_k}$ было минимальным.

Укороченный путь кончается последней парой $m\tau^*_{t_k}$ (просто больше пар $m\tau^*_{t_k}$ не образуем, так как они будут циклически повторяться при сохранении значений $\alpha_j$ и $\beta_i$). Обозначим укороченный путь символом $N^{(Q)}_{\{m\}}$, где скобки $\{\ldots\}$ определяют множество.

Для микропрограммы вида (1) получим

$$N^{(Q)}_{\{O_1,O_3\}} = N_{\min}^{(Q)(\mathfrak{R}_1)} = O_2\tau^*_{t_0};\ O_1\tau^*_{t_1};\ O_3\tau^*_{t_1}.$$

$$N_{\max}^{(Q)(\mathfrak{R}_2)} = O_2\tau^*_{t_0};\ s_{12}\tau^*_{t_1};\ p_3^{-1}\tau^*_{t_2};\ l_1\tau^*_{t_3};\ r_3\tau^*_{t_3};\ s_{12}\tau^*_{t_4};\ p_3^{-1}\tau^*_{t_5};\ l_1\tau^*_{t_6};$$

$$r_3\tau^*_{t_6};\ s_{12}\tau^*_{t_7};\ p_3^{-1}\tau^*_{t_8};\ l_1\tau^*_{t_9};\ r_3\tau^*_{t_9};\ O_1\tau^*_{t_{10}};\ O_3\tau^*_{t_{10}}.$$

$$N^{(Q)}_{\{l_1,r_3\}} = O_2\tau^*_{t_0};\ l_1\tau^*_{t_1};\ r_3\tau^*_{t_1};\ l_1\tau^*_{t_2};\ r_3\tau^*_{t_2}.$$

$$\mathfrak{R}_1 = \{O_2,\ O_1,\ O_3\};\quad \mathfrak{R}_2 = \{O_2,\ s_{12},\ p_3^{-1},\ l_1,\ r_3,\ O_1,\ O_3\} \text{ и т. д.}$$

Для микропрограммы вида (2) получим

$$N^{(Q)}_{\{O_1,O_3\}} = N^{(Q)(\mathfrak{R}_1)}_{\min} = O_2\tau^*_{t_0};\ O_1\tau^*_{t_1};\ O_3\tau^*_{t_1}.$$

$$N^{(Q)(\mathfrak{R}_2)}_{\max} = O_2\tau^*_{t_0};\ s_{12}\tau^*_{t_1};\ l_1\tau^*_{t_2};\ r_3\tau^*_{t_2};\ s_{12}\tau^*_{t_3};\ l_1\tau^*_{t_4};\ r_3\tau^*_{t_4};\ s_{12}\tau^*_{t_5};$$

$$l_1\tau^*_{t_6};\ r_3\tau^*_{t_6};\ O_1\tau^*_{t_7};\ O_3\tau^*_{t_7}.$$

$$N^{(Q)}_{\{l_1,r_3\}} = O_2\tau^*_{t_0};\ l_1\tau^*_{t_0};\ r_3\tau^*_{t_1};\ l_1\tau^*_{t_1};\ r_3\tau^*_{t_2};$$

$$\mathfrak{R}_1 = \{O_2,\ O_1,\ O_3\};\quad \mathfrak{R}_2 = \{O_2,\ s_{12},\ l_1,\ r_3,\ O_1,\ O_3\} \text{ и т. д.}$$

Нетрудно видеть, что максимальным отрезком времени работы автомата $A_Q$ по микропрограмме вида (1) будет отрезок времени $t_0 - t_{10}$, а для микропрограммы вида (2) — отрезок времени $t_0 - t_7$.

Назовем начальное и конечное значения отрезка времени его временными координатами.

Обозначим мощность множества $B_Q$ через $h(B_Q)$, где $B_Q$ — множество всех условий $\beta_i$ в микропрограмме $Q$.

Теорема 1. *Минимальный или укороченный путь работы автомата $A_Q$ по микропрограмме в регулярной записи содержит временные координаты минимального отрезка времени работы автомата $A_Q$ при выполнении данной микрооперации $m$.*

Доказательство получим при условии, что для данной микрооперации минимальный или укороченный путь выбирается таким, чтобы максимально опустить другие микрооперации $m_4$, предшествующие данной микрооперации $m$.

Для микрооперации $r_3$ минимальным отрезком времени его выполнения является, например, отрезок $t_0 - t_1$ или $t_1 - t_2$ (в зависимости от длины тактов).

Теорема 2. *Количество минимальных путей выполнения данной микропрограммы автоматом $A_Q$ имеет значение $\overset{(Q)}{\sigma}_{\min} \leqslant 2^{h(B_Q)}$.*

Доказательство получим при условии, что микропрограмма имеет $h(B_Q)$ разветвлений, обусловленных существованием условий $\beta_i$ в микропрограмме, а каждое условие $\beta_i$ дает два разветвления. Если все эти разветвления имеют одинаковой длины отрезки времени минимального выполнения заданной микропрограммы $Q$, то все они являются минимальными отрезками времени выполнения заданной микропрограммы $Q$.

Теорема 3. *Количество максимальных путей выполнения данной микропрограммы автоматом $A_Q$ имеет значение $\overset{(Q)}{\sigma}_{\max} \leqslant 2^{h(B_Q)}$.*

Доказательство данной теоремы аналогично доказательству теоремы 2, если слово «минимальный» заменить соответственно словом «максимальный».

Теорема 4. *Если $\overset{(Q)}{\sigma}_{\max} = \overset{(Q)}{\sigma}_{\min} = 2^{h(B_Q)}$, то максимальные пути работы автомата $A_Q$ по данной микропрограмме в регулярной записи совпадают с минимальными путями работы автомата $A_Q$ по данной микропрограмме в регулярной записи.*

Доказательство получим при условии, что все разветвления в микропрограмме — минимальные пути работы автомата $A_Q$, а максимальные пути выбираются не иначе, как минимальные пути выполнения заданной микропрограммы автоматом $A_Q$.

Длина указанных выше минимальных отрезков времени выполнения микрооперации $m$ является величиной допустимого времени $t^{(\nu)}_{\text{доп}}$ появления сигнала на выходе схемы, реализующей микрооперацию $m$ и описываемой ВП-функцией $f^{\Delta(m)}_{\Omega_\nu}$. Если для данного входа триггера регистра получают несколько ВП-функций $f^{\Delta(m)}_{\Omega_\nu}$, то поступают так, как указано в [1], а получаемое значение $t^{(\nu)}_{\text{доп}}$ будет допустимым временем появления сигнала на $\upsilon$-ом выходе входной части схемы данного триггера регистра, которая описывается ВП-функцией $f^{\Delta(m)}_{\Omega_\nu}$.

Если функция возбуждения входа триггера регистра является многотактной (например, с применением счетчиков в потенциально-импульсной элементной структуре [11–13]), то нужно, по мере надобности, между двумя соседними рассмотренными выше тактами ввести новые вспомогательные такты, так как длины тактов пока не определены, а сигналы для них получают из вспомогательного источника тактных сигналов.

Остальные вопросы образования ВП-функций от микропрограмм будут рассмотрены в следующих работах.

## ЛИТЕРАТУРА


1. Сиймон А., Изв. АН ЭССР, Физ. Матем., 22, 21 (1973).
2. Рабинович З. Л., Тр. Междунар. симп. по теории релейн. устройств и конечн. автоматов (ИФАК). Теория конечных и вероятностных автоматов, М., 1965, с. 215.
3. Рабинович З. Л., Кибернетика, № 3, 36 (1968).
4. Рабинович З. Л., Кибернетика, № 4, 25 (1968).
5. Сиймон А., Изв. АН ЭССР, Физ. Матем., 17, 270 (1968).
6. Сиймон А., Изв. АН ЭССР, Физ. Матем., 17, 391 (1968).
7. Сиймон А., Изв. АН ЭССР, Физ. Матем., 18, 347 (1969).
8. Сиймон А., Изв. АН ЭССР, Физ. Матем., 20, 468 (1971).
9. Глушков В. М., Кибернетика, № 5, 1 (1965).
10. Глушков В. М., Синтез цифровых автоматов, М., 1962.
11. Сиймон А., Изв. АН ЭССР, Физ. Матем., 19, 172 (1970).
12. Сиймон А., Изв. АН ЭССР, Физ. Матем., 19, 397 (1970).
13. Сиймон А., Изв. АН ЭССР, Физ. Матем., 19, 401 (1970).


*Институт кибернетики*
*Академии наук Эстонской ССР*

Поступила в редакцию
21/III 1973

*A. SIIMON*

**MEETOD VEKTOR-AJA ÜMBERLÜLIMISFUNKTSIOONIDE MOODUSTAMISEKS MIKROPROGRAMMIDEST**

Artiklis esitatakse märkus lähtemikroprogrammide minimeerimise kohta ja meetod juhtsignaalide ajakoordinaatide määramiseks, arvestades sünteesitava diskreetse automaadi maksimaalset töökiirust.

*A. SIIMON*

**ABOUT THE METHOD OF FORMING VECTOR-TIME SWITCHING FUNCTIONS FROM MICROPROGRAMS**

The paper contains a remark about the minimization of initial microprograms. A method of determining time coordinates for control signals in automata with a minimum working time is presented.