

## СИНТЕЗ АВТОМАТНЫХ МОДЕЛЕЙ ПРИ ПРОЕКТИРОВАНИИ ЛОКАЛЬНОГО УПРАВЛЕНИЯ АВТОМАТИЗИРОВАННОЙ СИСТЕМОЙ ПРОТИВОПОЖАРНОЙ ЗАЩИТЫ

*К.М. Чудотворова*  
Академия ГПС МЧС России

*В современном ритме технологического развития повышение уровня систем противопожарной защиты нефтеперерабатывающих предприятий занимает одну из важнейших составных частей обеспечения защиты населения от угроз техногенного характера. Для большей эффективности необходимо применять искусственный интеллект при создании автоматизированных систем пожарной безопасности и их проектировании. Данное исследование направлено на математическое моделирование функционирования системы идентификации инцидентов на нефтеперерабатывающем предприятии.*

**Ключевые слова:** Мили автомат, цифровой автомат, вершина графа, минимизация логической функции.

## SYNTHESIS OF AUTOMATIC MODELS IN DESIGNING OF LOCAL CONTROL OF AUTOMATED FIRE PROTECTION SYSTEM

*K.M. Chudotvorova*  
Academy of State Fire Service of the Ministry of Emergency Situations of Russia

*In the modern rhythm of technological development, increasing the level of fire protection systems of oil refineries is one of the most important parts of ensuring the protection of the population from man-made threats. For greater efficiency, it is necessary to apply artificial intelligence in the creation of automated fire protection systems and their design. This study is aimed at mathematical modeling of the functioning of the incident identification system at an oil refinery.*

**Keywords:** Miles automaton, digital automaton, graph vertex, minimization of a logical function.

Разработка новых принципов начинается с определения принципа опережения. Разработка системы управления является составной частью более широкого процесса создания автоматизированной системы противопожарной службы.

Принцип опережения заключается в том, что разработка системы управления должна начинаться с формализации и анализа результатов выполнения предшествующей фазы разработки.

При разработке проектов автоматизирования ППЗ этапу создания АСУ ППЗ предшествует этап разработки технологии и формирования технологического задания, следовательно, проект АСУ ППЗ должен начинаться с формализации технологического описания объекта и его требуемого поведения.

При этом достигаются следующие преимущества: во-первых, формализуется реальный результат проекта - технологические схемы, описание конструкций, описание требуемого поведения и т.д.; во-вторых, в описании требуемого поведения объекта нет уточнений, а, следовательно, доопределений, зависящих от выбранного способа управления.

Технологический алгоритм в виде последовательности операций

Имя агрегата	Операции	Воздействие	Реакции
A <sub>1</sub>	V <sub>1</sub>	Z <sub>1</sub> =1	$\overline{X_1}, X_2$
A <sub>1</sub>	V <sub>1</sub>	Z <sub>1</sub> =0	$\overline{X_2}, X_4$
A <sub>2</sub>	V <sub>2</sub>	Z <sub>2</sub> =1	$\overline{X_3}, X_4$
A <sub>2</sub>	V <sub>2</sub>	Z <sub>2</sub> =0	$\overline{X_4}, X_3$
A <sub>3</sub>	V <sub>3</sub>	Z <sub>3</sub> =1	$\overline{X_5}, X_6$
A <sub>3</sub>	V <sub>3</sub>	Z <sub>3</sub> =0	$\overline{X_6}, X_5$

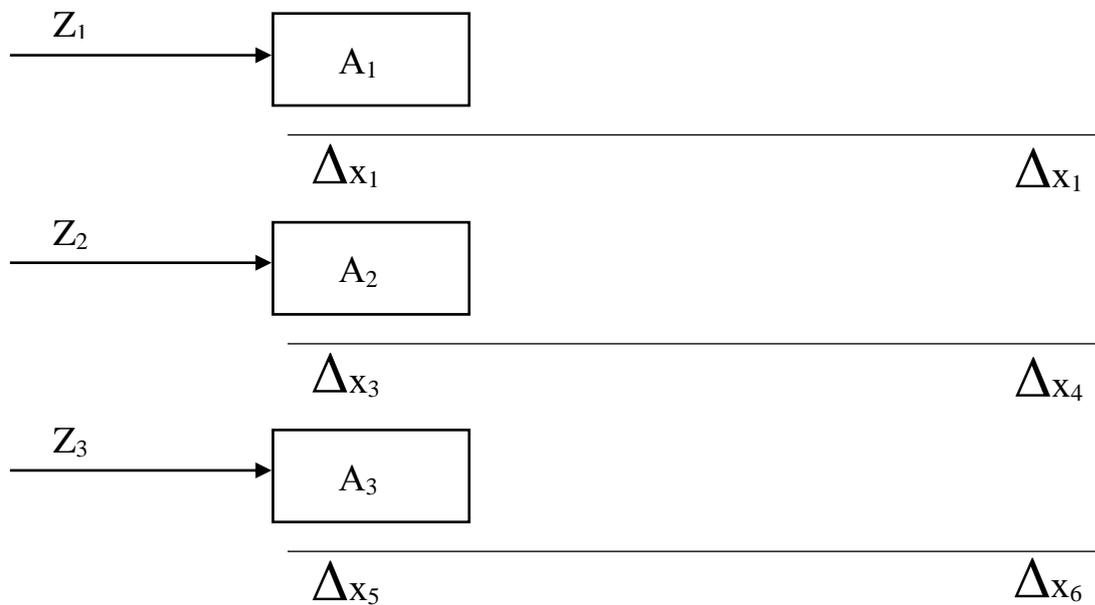


Рис. 1. Фрагмент объекта управления автоматизированной системой противопожарной защиты

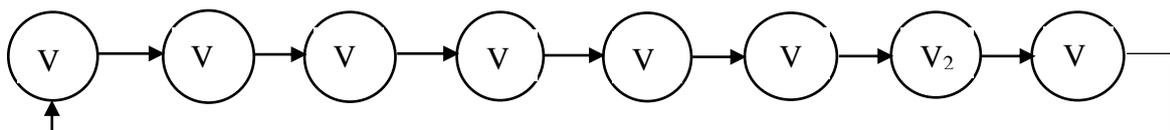


Рис. 2. Состояние в языках описания логических алгоритмов

Принцип опережения заключается в том, что разработка системы управления должна начинаться с формализации и анализа результатов выполнения предшествующей фазы разработки.

При разработке проектов автоматизирования ППЗ этапу создания АСУ ППЗ предшествует этап разработки технологии и формирования технологического задания, следовательно, проект АСУ ППЗ должен начинаться с формализации технологического описания объекта и его требуемого поведения.

При этом достигаются следующие преимущества: во-первых, формализуется реальный результат проекта - технологические схемы, описание конструкций, описание требуемого поведения и т.д.; во-вторых, в описании требуемого поведения объекта нет уточнений, а, следовательно, доопределений, зависящих от выбранного способа управления.

Поясним вышеизложенное простым примером. Пусть фрагмент объекта управления содержит три одинаково устроенных агрегата: А1, А2, А3 (рис. 2.). Каждый агрегат может выполнять две операции.

Агрегат А1: Операция  $V_1$  - перемещение от  $x_1$  до  $x_2$ ; операция  $V_2$  - перемещение от  $x_2$  до  $x_1$ .

Агрегат А2: Операция  $V_3$  - перемещение от  $x_3$  к  $x_4$ ; операция  $V_4$  - перемещение от  $x_4$  к  $x_3$ .

Агрегат А3: Операция  $V_5$  - перемещение от  $x_5$  к  $x_6$ ; операция  $V_6$  - перемещение от  $x_6$  к  $x_5$ .

Для простоты пусть каждый агрегат управляется одним элементом управления А1 -  $Z_1$ , А2 -  $Z_2$ , А3 -  $Z_3$ , представляемым значением булевой переменной:  $Z_1=1$  - выполняется 1-я операция,  $Z_2=0$  - выполняется 2-я операция. Кроме того, если агрегат достиг крайнего положения, то, пока не изменится команда, он остается в этом крайнем положении.

Пусть по технологии агрегаты должны циклически выполнять следующую последовательность операций:

$$V_1, V_3, V_2, V_5, V_1, V_4, V_2, V_6, V_1, V_3, \text{ и т.д.}$$

С точки зрения технолога задание на управление уже сформулировано - описана конструкция (рис. 2 и табл. 1), задано требуемое поведение - технологический алгоритм в виде последовательности операций. Задача управленца: по заданной технологии сформулировать условия управления (включение / отключение) приводов. Как правило, это творческий процесс, отражающий опыт, вкус, индивидуальность проектанта. Рассмотрим два возможных варианта:

Вариант 1 При анализе условий включения приводов просматривается счетный характер управления приводов  $Z_2$  и  $Z_3$ . Условия следуя этому свойству, можно сформулировать следующим образом:

- привод агрегата А2 -  $Z_2$  переключается в 1 на нечетных переходах в 1 датчика  $x_2$ , на четных -  $Z_2$  переходит в 0;

- привод агрегата А3 -  $Z_3$  переключается в зависимости от  $x_1$ , причем на четных переключениях  $Z_3$  становится 1, а на нечетных  $Z_3=0$ ;

- привод агрегата А1 -  $z_1$  переключается по более сложным условиям и зависит от четырех переменных  $x_3, x_4, x_5, x_6$ .

Сформулированные условия легко представимы автоматами Мура М1, М2, М3, графы переходов которых представлены на рис.5, 5.1.

Вариант 2 Проектант имеет склонность к сложным логическим построениям, в этом случае условия включения-отключения могут быть сформулированы следующим образом:

- привод агрегата А1- $Z_2$  включается переключателем  $x_5$ , пока не включен  $x_4$ , или переключателем  $x_6$ , пока не включен  $x_3$ ;

- привод А2- $Z_2$  включается переключателем  $x_2$  при невключенном  $x_6$  и остается включенным при отключенном  $x_2$ , если не включен  $x_4$ ;

- привод  $A_3-Z_3$  включается переключателем  $x_2$  если  $x_3$  выключен, и сохраняет значение 1 при выключенном  $x_1$  если  $x_5$  не включен, и выключается, если  $x_1$  становится равным 1. Сформулированные условия описывают комбинационный автомат М4:

$$M4 \begin{cases} Z_1 = \bar{x}_3 x_6 \vee \bar{x}_4 x_5 \\ Z_2 = \bar{x}_2 x_4 \vee \bar{x}_2 x_6 \\ Z_3 = \bar{x}_1 x_5 \vee \bar{x}_1 x_3 \end{cases}$$

По приведенным вариантам записи исходных данных для проектирования управления следует заметить:

Замечание 1. Автоматы  $M_1, M_2, M_3$  никакими формальными преобразованиями без дополнительной информации не сводятся к автомату М4;

Замечание 2. Автоматы  $M_1, M_2, M_3$  сложнее автомата М4 и их реализация требует памяти и, следовательно, сложнее, и менее предпочтительна для систем управления, важных для безопасности.

В этом примере специально логика упрощена. В практических ситуациях логика намного сложнее, поскольку связность между агрегатами (узлами) функционально глубже, чем по перемещению.

Из замечаний 1, 2 следует, что если технология проектирования на входе имеет только данные по управлению, то решения по управлению будут более субъективными, чем, если бы на входе информационной технологии проектирования, основываясь на принципе опережения, использовалось описание требуемого поведения объекта, то есть описание технологических процессов.

Принцип поэтапной верификации (ППВ). Одной из особенностей современных информационных технологий проектирования является непрерывность, то есть представление объекта цепочкой моделей  $A_1, A_2, \dots, A_n$ , а само проектирование - как последовательность  $P_1, \dots, P_{(n-1)}$  процедур преобразования или генерации этих моделей (рис. 4). В зависимости от результата проектирования - модели  $A_{(j+1)}$  процедура  $P_j$  может быть повторена или повторится часть процесса проектирования, начиная с некоторого предшествующего шага (на рис. 5 ребра графа представляют переходы между процедурами). На каждом шаге проектирования процедура  $P_j$  получает на входе модель  $A_j$ , дополнительную информацию  $B_j$  и осуществляет на основе этого синтез модели  $A_{(j+1)}$ . Преобразование очень близко отображению структуры в структуру:

$$\{A_i \# B_i\} \xrightarrow{P_i} A_{i+1}$$

Генерация предполагает использование  $A_i$  как ограничения на процедуру, а структура  $A_{i+1}$  синтезируется по некоторым своим правилам. Процедуры на каждом шаге многовариантны (см. пример рис. 1, 2), как правило, основаны на интуиции и искусстве проектировщика, поддерживаемых моделированием, перебором и анализом.

Поясним этот принцип на анализе эволюции важнейшего понятия - состояние в языках описаний логических алгоритмов.

В первых разработках САПР устройств и систем управления [7] была попытка построить систему проектирования устройств и систем управления, используя в качестве языков спецификации некоторые расширения языков теории автоматов или языка теории управления [10]. Так, для САПР дискретных устройств основные понятия языка - это: входы, выходы, внутренние состояния, булевы функции, автомат, блок и структура из блоков. При иерархическом построении в конечном итоге каждый блок представляется некоторым преобразователем, описывающим зависимость выходов от входов с привлечением по-

нения внутреннее состояние или элемент памяти. САПР, построенные на этих принципах, носили исследовательский характер, и практически использовались авторами.

Вторая волна работ, начатая с появлением машин третьего поколения в [7] характеризуется значительно большим вниманием к языкам спецификаций, вводятся новые понятия: процесс, приоритеты процессов, описания по умолчанию.

Расширяется и система базовых понятий: так наряду с понятием состояние, вводится понятие ситуация [14] и развивается теория конечно-автоматных языков. В языках, описывающих управление технологическими процессами, вводится понятие операции как воздействия на объект, соответствующего некоторой технологической функции. Следует заметить, что введенные понятия, расширяющего: 1 понятие состояние, значительно усилили привлекательность языков теории в практических приложениях. Так, например, был разработан ряд промышленных продуктов на основе языка вариантов и только в некоторых случаях, подкрепляемых формальными преобразованиями моделей. Схематически работа проектанта на каждом этапе представлена диаграммой рис. 3 и 4. Важнейшей задачей анализа проектных решений является верификация, то есть проверка отсутствия ошибок [8, 9].

Поскольку процедуры интерактивны, то при их выполнении или при введении дополнительных данных велика вероятность внесения ошибок проектантом, поэтому важно "не накапливать неисправности", а стремиться на каждом шаге проекта, обнаруживать вероятные ошибки, исходя из свойств моделей и данных, то есть осуществлять верификацию синтезируемых проектных решений.

Принцип поэтапной верификации заключается в разработке для каждого этапа (модели) правил корректности моделей и данных, и проверки их выполнения.

Для приведенного выше примера с тремя агрегатами для выполнения ППВ применительно к структуре переходов необходимо, например, для фазы анализа технологии проверять правильность переходов между операциями (рис. 1), а для фазы генерации алгоритмов управления проверять правильность переходов между управляющими командами (рис. 5, 5.1).

При этом первая проверка верифицирует корректность следования, а вторая - условия завершения операции.

Принцип объектно-ориентированного агрегирования. Важнейшим этапом в разработке языка описания является структурирование данных (понятий, атрибутов) в базовые конструкции языка. Принцип объектно-ориентированного агрегирования заключается в группировании данных в базовые конструкции языка описания по степени их "близости" в объекте управления или его поведении.

Поясним этот принцип на анализе эволюции важнейшего понятия - состояние в языках описаний логических алгоритмов.

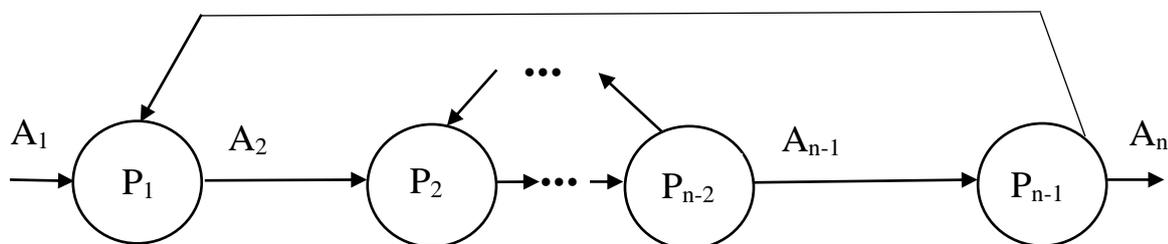


Рис. 3. Схематическая работа проектирования

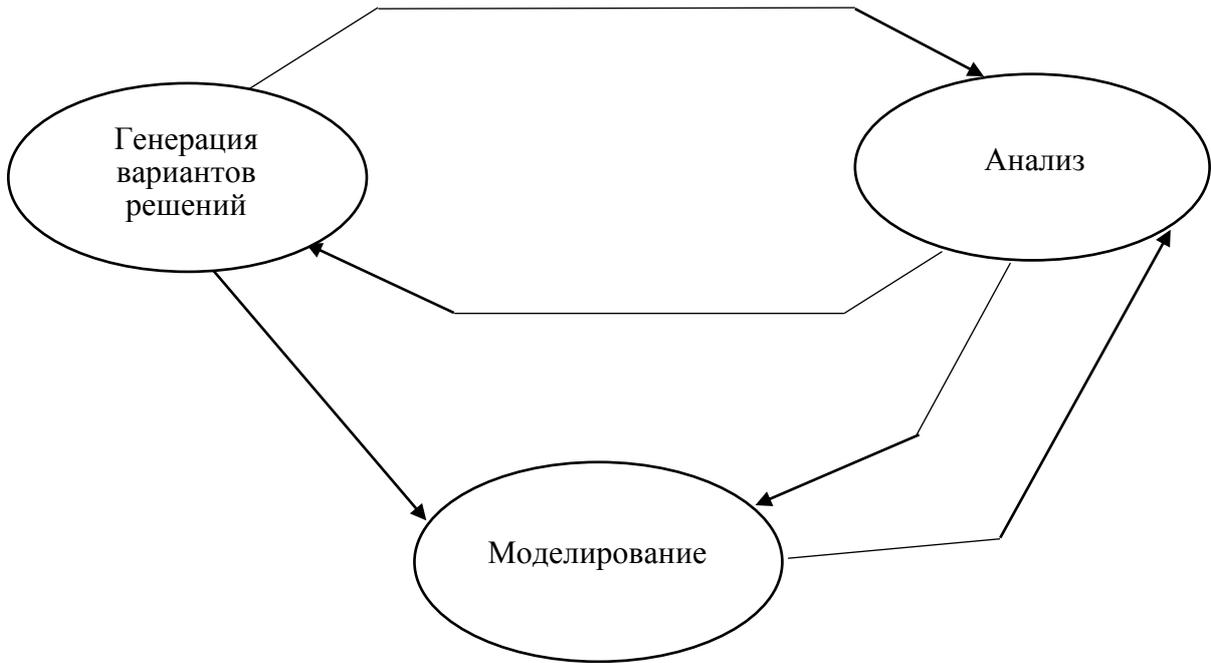


Рис. 4. Схематический процесс формализации

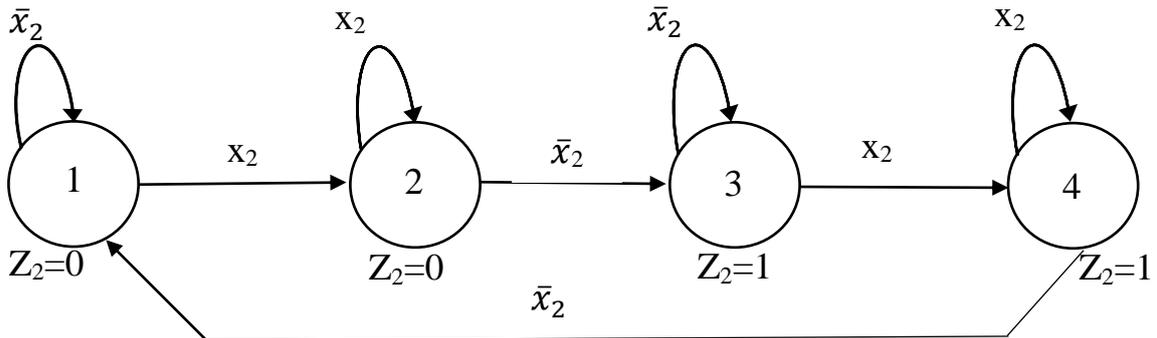


Рис. 5. Переходы между управляющими командами первого уровня

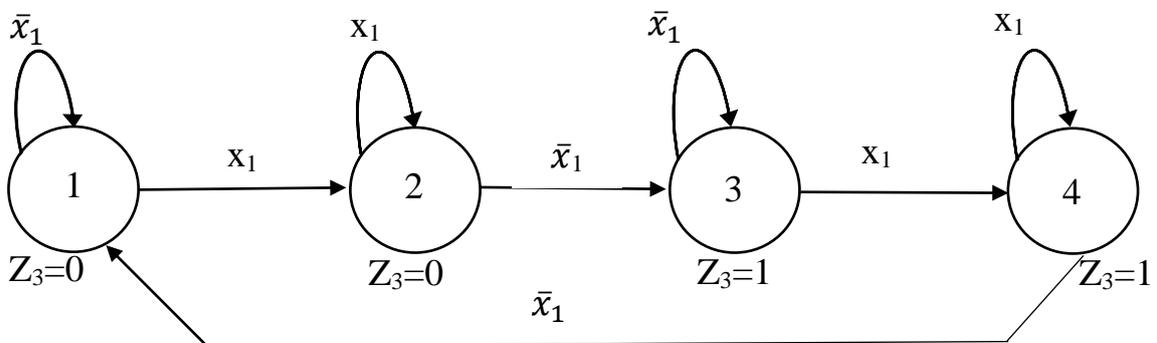


Рис. 5.1. Переходы между управляющими командами второго уровня

В первых разработках САПР устройств и систем управления [11] была попытка построить систему проектирования устройств и систем управления, используя в качестве языков спецификации некоторые расширения языков теории автоматов или языка теории управления [10]. Так, для САПР дискретных устройств основные понятия языка - это: входы, выходы, внутренние состояния, булевы функции, автомат, блок и структура из блоков. При иерархическом построении в конечном итоге каждый блок представляется некоторым преобразователем, описывающим зависимость выходов от входов, с привлечением понятия внутреннее состояние или элемент памяти. САПР, построенные на этих принципах, носили исследовательский характер, и практически использовались авторами.

Вторая волна работ, начатая с появлением машин третьего поколения [9] характеризуется значительно большим вниманием к языкам спецификаций, вводятся новые понятия: процесс, приоритеты процессов, описания по умолчанию.

Расширяется и система базовых понятий: так наряду с понятием состояние, вводится понятие ситуация [11] и развивается теория конечно-автоматных языков [11]. В языках, описывающих управление технологическими процессами, вводится понятие операции как воздействия на объект, соответствующего некоторой технологической функции [10]. Следует заметить, что введенные понятия, расширяющегося понятие состояние, значительно усилили привлекательность языков теории в практических приложениях.

В теоретическом плане значение этого перехода исследовано Кузнецовым О.П. [11] и нашло выражение в понятии выразительность языка. Заметим, что эволюция понятия состояния (такта): вход/выход в таблицах включений [8] и таблицах переходов [8], подмножество входов/выход - граф переходов, подмножество входов/выходной интервал - граф переключений [13]; направлена на обобщение свойств и учет специфики логических алгоритмов. В то же время, если ориентироваться на учет специфики объекта и ближайшего аналога понятия состояния в поведении объекта - операции, то целесообразно в конструкции языка (соответствующей операции) отразить:

- контроль условия начала операции;
- ожидаемую реакцию объекта;
- контроль условия окончания операции;
- интерактивность;
- параллелизм.

Принцип минимальной достаточности (МД-принцип). Одной из важных особенностей первичных описаний логических алгоритмов является локальность, то есть описание не полных входо/выходных наборов, а лишь отличий от предыдущего шага.

Принцип минимальной достаточности заключается в использовании таких процедур анализа и синтеза алгоритмов управления, для которых достаточно имеющейся локальной информации, содержащейся в операционных описаниях, либо требующей минимального ее расширения. Проиллюстрируем этот принцип (МД-принцип) на примере синтеза алгоритма управления тремя агрегатами (табл. 2) по описанию последовательности операций (рис. 6).

Таблица 2

Входо/ выходная последовательность

	X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	X <sub>4</sub>	X <sub>5</sub>	X <sub>6</sub>	Z <sub>1</sub>	X <sub>2</sub>	Z <sub>3</sub>
a <sub>1</sub>	1	0	1	0	1	0	1	0	0
a <sub>2</sub>	0	0	1	0	1	0	1	0	0
a <sub>3</sub>	0	1	1	0	1	0	1	1	0
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•
a <sub>1σ</sub>	1	0	1	0	0	0	0	0	0

Таблица переходов

$a_1$	$a_2$	•	•	•	$a_{1\sigma}$
$\langle 1 \rangle^{100}$	$\langle 1 \rangle^{100}$		–	–	–
		$\langle 2 \rangle^{110}$			
–	–	–	–	–	–
•	•	•	•	8	
•	•	•	•		
•	•	•	•		
1	–	–		$\langle 8 \rangle^{000}$	$\langle 8 \rangle^{000}$

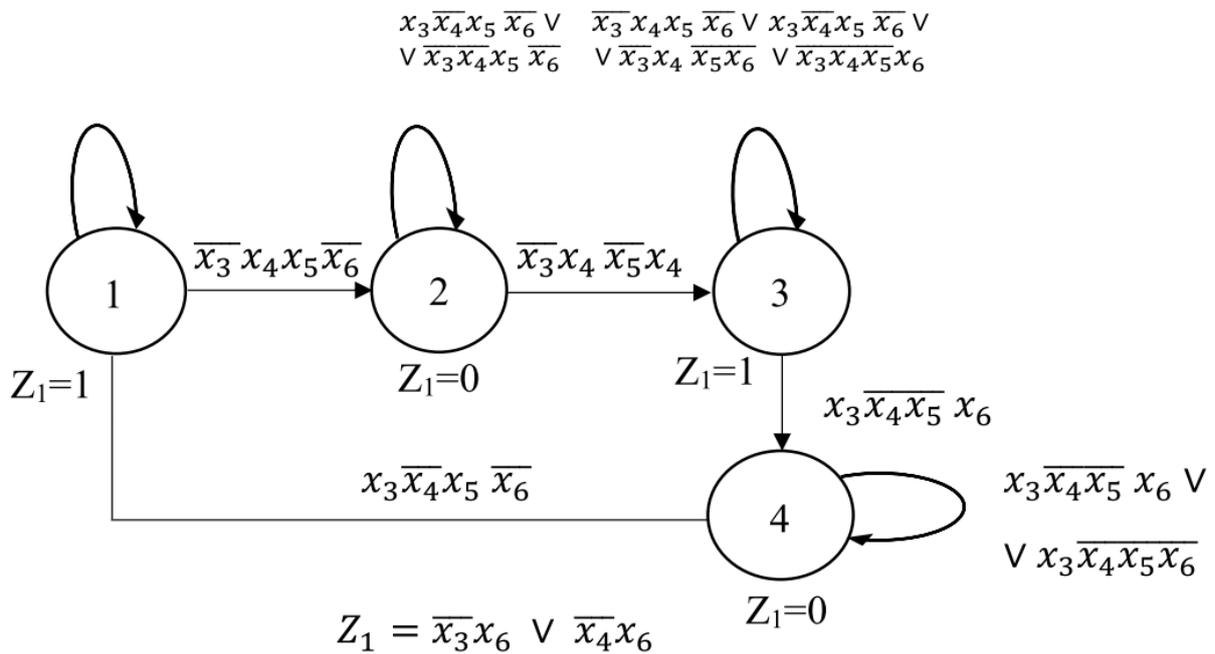


Рис. 6. Синтез алгоритма управления тремя агрегатами

По графу операций в данном достаточно простом случае легко восстановить полностью входо/выходную последовательность (табл. 2), затем построить таблицу переходов (табл. 3) и провести синтез автомата по всем классическим правилам [8].

Понятно, что для реальных примеров даже средней размерности в несколько десятков входов такой путь сопряжен со значительными трудностями и самое главное - экспоненциальным ростом объема информации. Если производить декомпозицию на ранних стадиях проекта, то это сопряжено с сужением возможностей поиска оптимальных решений. В работе автора [13] предложена модель - след автомата и методы синтеза на основе МД-принципа, результаты которого представлены на рис. 6. Как видно, удается без "разбухания" информации получить минимальное решение.

На основе сформулированных принципов автором предложена методология сквозного проектирования, реализованная в программе «Формирование номенклатуры программно-аппаратных средств по оптимизации системы сигнализации с помощью цифрового автомата» [16].

## Литература

1. Топольский Н.Г. Проблемы и принципы создания интегрированных систем безопасности и жизнеобеспечения // Материалы четвертой международной конференции «Информатизация систем безопасности» - ИСБ-95. - М.: ВИПТШ МВД РФ. - 1995. - С. 14-
2. Пупков К.А., Егупов Н.Д. Методы классической и современной теории автоматического управления. М.: Издательство МГТУ им. Н.Э. Баумана. - 2004.
3. Энциклопедия безопасности [Электронный ресурс]. - Режим доступа: <http://protivpozhara.ru/tipologija/teorija/pozhary-i-vzryvy-na-vzryvopozharnoопасных-objektax> - Против пожара.
4. Юревич Е.И. Теория автоматического управления. СПб.: ВХБ-Петербург. - 2007.
5. Little, I.D.C. Models and Managers. The Concept of a Decision Calculus // Management Science. – 1970. – v. 16. № 8.
6. Ларичев О.И., Петровский А.В. Системы поддержки принятия решений. Современное состояние и перспективы их развития. // Итоги науки и техники. Сер. Техническая кибернетика. – Т.21. М.: ВИНТИ. - 1987. С. 131-164.
7. Edwards, J.S. Expert Systems in Management and Administration – Are they different from Decision Support Systems? // European Journal of Operational Research. - 1992. – V. 61. – P. 114-121.
8. Топольский Н.Г., Тараканов Д.В. Моделирование динамики параметров мониторинга пожара в здании на основе клеточных автоматов // Системы безопасности – 2016: материалы 25-й Международной научно-технической конференции. – М.: Академия ГПС МЧС России. - 2016. – С. 585–588.
9. Чудотворова К.М. Информационная технология сквозного проектирования локального управления автоматизированной системой противопожарной защиты// Материалы XII Международной научно-практической конференции «Актуальные проблемы пожарной безопасности, предупреждения и ликвидации ЧС» – М.: Академия ГЗ им. Малика Габдуллина МЧС Республики Казахстан.
10. Martinov G.M. Implementation of Control for Peripheral Machine Equipment Based on the External Soft PLC Integrated with CNC / G.M. Martinov, N.V. Kozak, R.A. Nezhmetdinov // 2017. - International Conference on Industrial Engineering. - Applications and Manufacturing (ICIEAM). - 16-19 May. - 2017. - p.1-4.
11. Martinova L.I. The Russian multi-functional CNC system AxiOMA control: Practical aspects of application / L.I. Martinova, N.V. Kozak, R.A. Nezhmetdinov // Automation and remote control. - Vol: 76. - No: 1. - p. 179-186. - JAN 2015.
12. Martinov G.M. Approach to implementing hardware-independent automatic control systems of lathes and lathe-milling CNC machines / G.M. Martinov, R.A. Nezhmetdinov, and A.U. Kuliev // Izv.Vuz. Av. Tekhnika. - 2016. - no. 2. - pp. 128–131. [Russian Aeronautics (Engl. Transl.) vol.59, no. 2, pp. 293-296.]
13. Mori, Masahiko. 5 axis mill turn and hybrid machining for advanced application / Mori, Masahiko, Fujishima Makoto, Yohei, // Procedia CIRP 1 (2012). - p.p. 22-27.
14. Patent EP 0690426 (A2), 03.01.1996, кл. G09B 19/00 Système d'entraînement à l'emploi de l'ordinateur.
15. Patent EP 1111966 A, 27.06.2001, Signaling Device, G08B 5/00.
16. Свидетельство о государственной регистрации программы для ЭВМ 2021664194, 01.09.2021, Волкова К.М.

## Сведения об авторе

**Чудотворова Ксения Михайловна**, старший лейтенант внутренней службы, Академия ГПС МЧС России, E mail: ksenifire@bk.ru, тел. 8 966 026 17 18