УДК 681.518.3

МЕТОД СИНТЕЗА LD-ПРОГРАММ ДЛЯ ПЛК НА ОСНОВЕ КОНЦЕПЦИИ СУПЕРВИЗОРНОГО УПРАВЛЕНИЯ ДИСКРЕТНО-СОБЫТИЙНЫМИ СИСТЕМАМИ, ПРЕДСТАВЛЕННЫМИ В СЕТЯХ ПЕТРИ

А.С. Хадеев[1], С.А. Браништов[2]
(*[1]ЗАО "АтлантикТрансгазСистема", [2]ИПУ РАН*)

I. Теория управления в дискретно-событийных системах

Динамика в моделях дискртено-событийных (ДСС) определяется асинхронным возникновением дискретных событий, переводящих систему из одного состояния в другое. События могут быть контролируемыми – например нажатие клавиши на клавиатуре или включение некоего оборудования, а могут быть неконтролируемыми – например возникновение аварийной ситуации. В фундаментальной работе В. Вонхама и П. Раманджа [1] представлена теория управления в ДСС, в которой методология логического управления в ДСС основана на концепции супервизорного управления. Супервизор ограничивает поведение системы в соответствии с набором предварительно заданных ограничений. Фактически он предотвращает переход системы в одно из запретных состояний, предусмотренных ограничениями. На рис. 1 показано взаимодействие объекта G с супервизором S, где e_n – все события, возникающие в объекте G, e_u – события, разрешенные супервизором S.

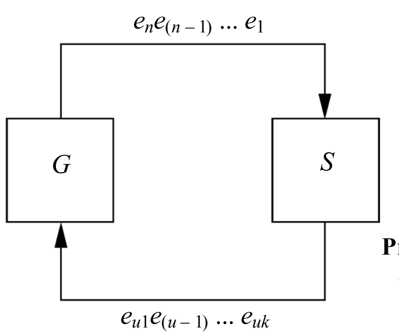

Рис. 1. *Схема взаимодействия объекта и супервизора*

В серии работ А. Амбарцумяна [8–11] исследованы ситуации, в которых из одного состояния системы могут последовать разные события, в зависимости от предыстории.

Теория супервизорного управления сформулирована в механизмах конечных автоматов и формальных языков, которые обладают известной проблемой "взрыва состояний" – при моделировании систем количество состояний модели экспоненциально растет при увеличении системы, что существенно усложняет расчет подобных моделей. Сети Петри лишены этого недостатка, к тому же обладают удобным и наглядным графическим представлением.

Международный стандарт МЭК 61131-3 предопределяет несколько языков для программирования ПЛК:

– IL (Instruction List) – аппаратно-независимый низкоуровневый язык, представляющий алгоритм выполнения в виде списка инструкций;

– LD (Ladder Diagram) – язык лестничной логики, иначе называемый релейно-контактными схемами (РКС), позволяет в графическом виде представить программную реализацию электрических схем на базе электромагнитных реле;

– FBD (Function Block Diagram) – графический язык, представляющий алгоритм в виде связи функциональных блоков, имеющих входы и выходы;

– SFC (Sequential Function Chart) – графический высокоуровневый язык, описывающий программу в виде последовательности состояний и условий переходов;

– ST (Structured Text) – язык структурированного текста, похожий на Паскаль.

Несмотря на то, что для совершенствования техник программирования ПЛК приложено множество усилий, развитию синтеза логики управления уделено гораздо меньше внимания. В данной работе рассмотрен целостный подход к формированию логики поведения ПЛК, начиная с формирования модели для объекта и кончая синтезом программы для ПЛК на языке LD.

II. Синтез супервизора в сетях Петри

В основе моделирования большого количества динамических систем лежат три основополагающих понятия: событие, условие и состояние. Состояние системы в момент времени t описывает ее поведение в этот момент измеряемым способом. События – это действия, имеющие место как в самой системе, так и извне, по отношению к системе. Возникновение события обусловливается некоторым количеством условий. В сетях Петри события моделируются переходами, а условия – позициями. Некоторые позиции являются входными к переходам, в них задаются условия, необходимые для возникновения события. Другие условия выходные, в них определяются последствия возникновения события. Позиции, переходы и связи между ними определяют основные компоненты графа сетей Петри. В данной работе используется расширенная концепция сетей Петри, подробно и системно изложенная Д. Питерсоном [2] и В. Котовым [3].

Граф сети Петри – это взвешенный двудольный ориентированный граф (P, T, A, ω) где P – конечный набор позиций (один из типов вершин в графе); T – конечный набор переходов (другой тип вершин); $A \subseteq (P \times T) \cup (T \times P)$ – набор дуг от позиций к переходам и от переходов к позициям; $\omega : A \to \boldsymbol{N} / \{0\}$ – функция, взвешивающая дуги. Предполагается, что в графе нет изолированных позиций или переходов.

В графическом представлении позиция обозначается в виде кружка, а переход – в виде черты. Условия-позиции и события-переходы связаны отношением причинно-следственной связи, которая выражается в виде направленных дуг, ведущих из позиций в переходы и из переходов в позиции. Позиции, из которых ведут дуги в данный переход, называются входными по отношению к нему, а на которые идут дуги из перехода – выходными.

Взвешивающая функция ω сопоставляет каждому соединению двух вершин вес, или, иначе говоря, кратность, которая, определяется как число $n > 0$. Сеть Петри, в которой кратность всех дуг равна 1, называется *ординарной*. Если $n > 1$, то в графическом представлении сети число n выписывается рядом с дугой, а иногда такая дуга заменяется пучком из n дуг, соединяющих вершины. Иначе говоря, если кратность дуги $\omega(p_i, t_i) = k$, тогда от p_i к t_i ведут k дуг; то же касается и дуг от переходов к позициям, для которых $\omega(t_i, p_i) = k$.

Состояние системы в сети Петри обозначается *маркировкой* сети m, а именно присвоением маркеров (фишек) позициям сети. Маркировка нужна для моделирования последовательности событий в системе. О текущей маркировке сети Петри говорят как о состоянии сети, а изменение маркировки соответствует изменению состояния объекта. На графе сети Петри фишки изображаются точкой в кружке позиции. Допустимо вместо точек указывать цифру, соответствующую числу фишек в этой позиции. Формально, маркированная сеть Петри записывается как пятерка (P, T, A, ω, μ_0), где μ_0 – начальная маркировка сети в виде вектора, содержащего число фишек в каждой позиции. Такой вектор носит название *вектора маркировки* $\mu = [\mu(p_1), \mu(p_2), \ldots, \mu(p_n)]^T \in \boldsymbol{N}^n$, где p_i – позиции сети, пронумерованные в выбранном порядке, $\mu(p_i)$ – их текущая маркировка.

Препозицией для позици p называется набор входных по отношению к ней переходов: $I(p) = \{t \in T{:}(t, p) \in A\}$;

постпозицией для позиции p называется набор выходных по отношению к ней переходов $O(p) = \{t \in T:(p, t) \in A\}$; p будет *исходной позицией*, если $I(p) = 0$, и конечной позицией, если $O(p) = 0$. Точно так же определяются препозиция и постпозиция для переходов.

Динамика моделируемой системы находит свое отражение в *выполнении* сети Петри, которое можно представить как совокупность локальных срабатываний переходов, что соответствует реализации соответствующих им событий. Срабатывать могут только *разрешенные переходы*, т. е. такие, для которых выполнены все условия реализации соответствующего события и для которых формально число фишек во входной позиции больше либо равно весу входной дуги $\mu(p_i \geq \omega(p_i, t_i)$, для всех $p_i \in I(t_j)$. Срабатывание перехода меняет маркировку связанных с ним позиций, извлекая фишки из входных позиций и добавляя в выходные. Соответственно меняется и состояние сети.

Традиционная модель сетей Петри представляет структуру управления, но не отражает ее связи с объектом управления. Для ее развития появились различные расширения сети Петри, в которых по-разному моделируется взаимодействие сети Петри с управляемым процессом. Расширения предлагают новые свойства дуг, переходов и позиций, новые правила выполнения. Одна из модификаций, носящая называния *управляемых сетей Петри*, предполагает ввод в сеть Петри специального множества *управляющих позиций*, каждая из которых представляет отдельный входной или выходной сигнал, за счет чего достигается взаимодействие с объектом. Динамика в этой модели управляется помещением метки в позиции, соответствующие входным сигналам. Метки, появившиеся в позициях, соответствующих выходным сигналам, удаляются, имитируя передачу сигналов из модели.

Матрица инцидентности сети Петри содержит информацию о структуре сети Петри, представленной весами дуг, соединяющих позиции с переходами. Формально матрица D определяется размерностью $n \times l$, где n – число позиций, l – число переходов. Элементы матрицы $d_{ij} = \omega(t_j, p_i) - \omega(p_i, t_j)$. Проще говоря, элементы матрицы содержат числа, отражающие, как изменится наполненность позиций фишками после срабатывания перехода.

Матрицу инцидентности можно представить в виде композиции двух матриц: выходной D^-, представляющей дуги от позиций к переходам, и входной D^+, представляющей дуги от переходов к позициям:

$$D = D^+ - D^- \quad (D^+, D^- \geq 0).$$

Соответственно, элементы d_{ij}^+ и d_{ij}^- матриц D^+ и D^- определяются следующим образом:

$$d_{ij}^+ = \begin{cases} \omega\left(t_j,\, p_i\right), \text{ если } \left(t_j,\, p_i\right) \in A \\ 0 \end{cases};$$

$$d_{ij}^- = \begin{cases} \omega\left(p_i,\, t_j\right), \text{ если } \left(p_i,\, t_j\right) \in A \\ 0 \end{cases}.$$

Сеть Петри имеет следующий вид:

$$S = (P, T, D^-, D^+).$$

Инвариантой позиций называют целочисленный вектор $X = \{x_i\}$, $i = 1, 2,\ldots, n$, такой, что $x_i \in \{0, 1\}$, и этот вектор является решением линейной системы

$$X^T D = 0. \tag{1}$$

Вектор X единичными значениями характеризует подмножество позиций, в которых при любой достижимой маркировке число фишек постоянно.

Пусть задана модель исследуемого объекта в виде сети Петри с n позициями и l переходами $D_p \in \boldsymbol{Z}^{l\times n}$, $\boldsymbol{Z}$ – множество целых чисел, равно как и набор ограничений, определяющих ее поведение. Наша цель – сконструировать супервизор $D_c = \boldsymbol{Z}^{l\times n_c}$, который, будучи присоединенным к сети модели объекта $D \in \boldsymbol{Z}^{l\times(n+n_c)}$, обеспечит выполнение заданных ограничений. Супервизор будет состоять из новых позиций, но использовать переходы сети Петри, моделирующей объект.

Эффективный метод задания ограничений предполагает, что маркировки m_i могут быть выражены в виде неравенств

$$\mu_i + \mu_j \leq 1, \tag{2}$$

где μ_i, μ_j – маркировки позиций p_i, p_j. Данная формула говорит о том, что обе позиции не могут содержать фишки одновременно. Это неравенство может быть преобразовано в равенство путем добавления подстановочной переменной $\mu_s \geq 0$:

$$\mu_i + \mu_j + \mu_s = 1. \tag{3}$$

Подстановочная переменная представляет новую позицию p_s, обеспечивающую стабильность суммы меток в позициях p_i и p_j, которая всегда будет равна 1. Новая позиция входит в *сеть супервизора* и носит название *позиции супервизора*. Таких позиций будет столько же, сколько ограничений наложено на модель.

Добавление новой позиции изменяет композитную матрицу инцидентности D объединенной сети, добавляя к ней строку, представляющую подстановочную позицию. Фактически эта строка принадлежит матрице инцидентности супервизора D_c, ненулевые элементы в которой обозначают дуги, соединяющие позицию P_s с переходами сети модели процесса.

Все ограничения могут быть сгруппированы и приведены к матричной форме

$$L \cdot \mu_p \leq b, \tag{4}$$

где μ_p – вектор маркировки сети Петри объекта $\mu_p \in \boldsymbol{Z}^n$, $\mu_p \geq 0$; $L \in \boldsymbol{Z}^{n_c\times n}$; $b \in \boldsymbol{Z}^{n_c}$; n_c – количество ограничений типа (2). Другими словами, мы хотим исключить из достижимых маркировок сети Петри те, которые не соответствуют (4).

Аналогично преобразуем неравенство в равенство

$$L \cdot \mu_p + \mu_c = b, \tag{5}$$

где $\mu_c \in \boldsymbol{Z}^n$, $\mu_c \geq 0$ – вектор, представляющий маркировку сети супервизора. Далее вычислим всю структуру сети супервизора, включая дуги, соединяющие позиции этой сети с переходами сети, описывающей процесс.

Замкнутая система (оригинальная сеть Петри плюс супервизор) имеет следующую структуру:

$$D = \left[\frac{D_p}{D_c}\right]; \; m_0 = \left[\frac{\mu_{p0}}{\mu_{c0}}\right]. \qquad (6)$$

Расширим формулу инварианты позиций (1)

$$X^T D = [LI]\left[\frac{D_p}{D_c}\right] = 0 \Leftrightarrow LD_p + D_c = 0 \Updownarrow$$

$$D_c = -LD_p, \qquad (7)$$

где L – единичная матрица $n_c \times n_c$, так как все коэффициенты подстановочных переменных равны 1. Матрица супервизора D_c, размерности $n_c \times l$ содержит дуги, соединяющие позиции сети супервизора с переходами сети процесса.

Формула (7) вычисляет супервизор сети Петри D_c, который удовлетворяет заданным ограничениям при известной модели процесса. Начальная маркировка позиций супервизора μ_{c0} при известной маркировке сети процесса μ_{p0} вычисляется следующим образом:

$$L\mu_{p0} + \mu_{c0} = b \Leftrightarrow \mu_{c0} = b - L\mu_{p0}, \qquad \mu_{c0} \geq 0. \qquad (8)$$

III. Пример построения супервизора

Интересным является пример, имеющий практическое применение, такой, как железнодорожный стрелочный перевод (стрелка). Стрелка, как показано на рис. 2, имеет два конечных положения: на первом и на втором путях, а также два промежуточных, в которые попадает путем подачи команды на перестановку. Помимо этого, стрелка оснащена датчиком, распознающим нахождение на стрелке вагона. Задача управления в данном случае заключается в том, чтобы предотвратить переключение стрелки в момент нахождения на ней железнодорожного вагона.

Дискретно-событийная модель стрелки в виде сети Петри представлена на рис. 3.

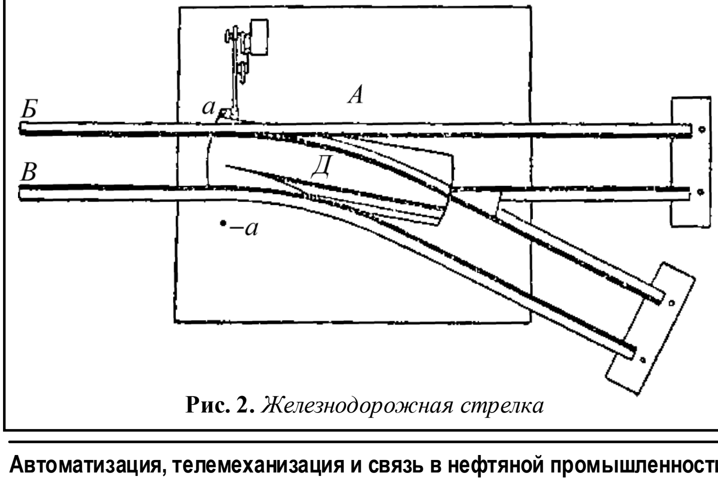

Рис. 2. *Железнодорожная стрелка*

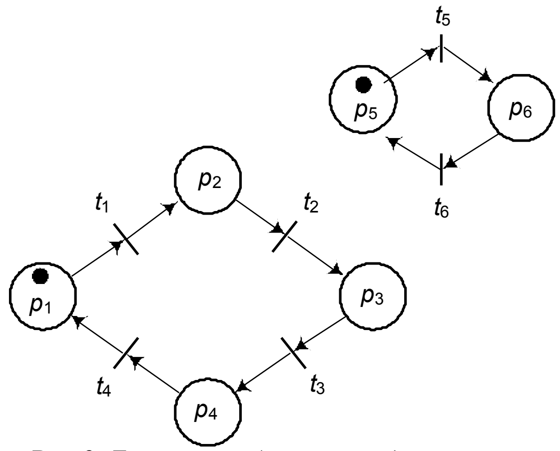

Рис. 3. *Дискретно-событийная модель стрелки:* p_1 – стрелка указывает на путь 1; p_2 – стрелка переводится на путь 2; p_3 – стрелка указывает на путь 2; p_4 – стрелка переводится на путь 1; p_5 – стрелка свободна; p_6 – стрелка занята вагоном; t_1 – разрешение на перевод стрелки на путь 1; t_2 – стрелка переведена на путь 1; t_3 – разрешение на перевод стрелки на путь 2; t_4 – стрелка переведена на путь 2; t_5 – вагон находится на стрелке; t_6 – вагон покинул стрелку

Матрица инцидентности D_p для данной сети Петри приведена ниже:

$$D_p = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}.$$

Ограничения для данной сети Петри, представляющей модель стрелки, состоят в том, что позиции p_2 и p_4 достижимы только при отсутствии маркировки в позиции p_6. Зададим ограничения в соответствии с формулой (5):

$$\mu_2 + \mu_4 + \mu_6 \leq 1.$$

Соответственно вектор ограничений L будет выглядеть следующим образом:

$$L = [0\ 1\ 0\ 1\ 0\ 1].$$

Проведя преобразование в соответствии с формулой (8) получим вектор супервизора D_c:

$$D_c = [-1\ 1\ -1\ 1\ -1\ 1]. \qquad (9)$$

Полная матрица инцидентности D сети Петри с супервизором приведена ниже:

$$D = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix}.$$

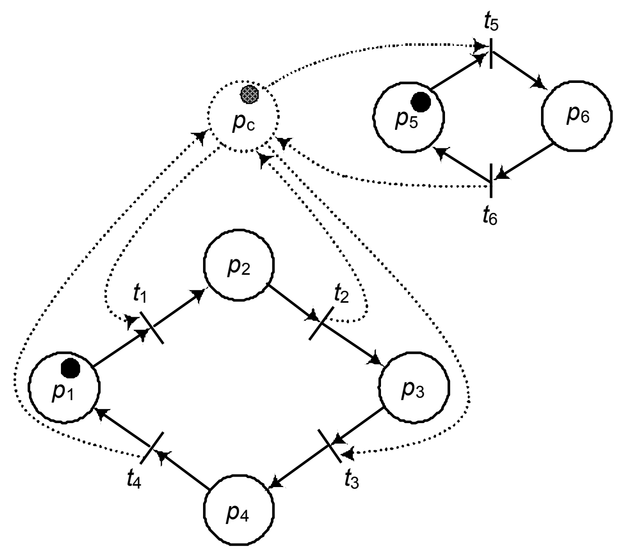

Рис. 4. *Сеть Петри с супервизором*

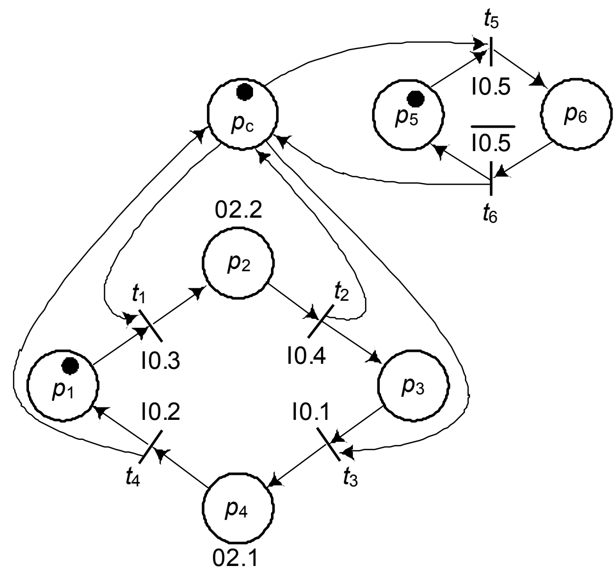

Рис. 5. *Сопоставление входов и выходов ПЛК позициям и переходам сети Петри*

Начальная маркировка μ_c в соответствии с (9) будет следующей:

$$\mu_c = 1.$$

Сеть Петри с синтезированным супервизором приведена на рис. 4.

IV. Метод преобразования в язык LD

Составление модели поведения объекта и сопоставление ей входов и выходов супервизора является творческим занятием, требующим участия человека. Но методы преобразования избавляют от необходимости ручного учета последовательностей команд, а представление модели в виде сети Петри делает поведение объекта наглядным, что снижает вероятность ошибок. Входы ПЛК сопоставляются условиям для срабатывания переходов, а выходы – как позиции в сети Петри.

Для рассматриваемого примера сопоставление входов и выходов ПЛК модели объекта в сети Петри показано на рис. 5.

Входы ПЛК PLC_o	Условие для перехода $x(t_i)$	Описание
I0.1	t_1	Разрешение на перевод стрелки на путь 1
I0.2	t_2	Стрелка переведена на путь 1
I0.3	t_3	Разрешение на перевод стрелки на путь 2
I0.4	t_4	Стрелка переведена на путь 2
I0.5	t_5	Вагон находится на стрелке

Выходы ПЛК PLC_I	Позиции P	Описание
O2.1	p_4	Стрелка переводится на путь 1 (команда)
O2.2	p_2	Стрелка переводится на путь 2 (команда)

Преобразование модели на сетях Петри в LD представляет из себя следующий алгоритм:

1. Переход t_i учитывается в программе LD в виде ступеньки, если хоть одна из выходных позиций для него p_k отмечена как выход ПЛК. Для каждой такой позиции составляется отдельная ступенька LD:

$$\forall t_i,\ t_i \in I(p_k),\ p_k \in PLC_o.$$

2. Формулируются условия возникновения перехода $\chi(t_i)$ по принципу конкатенации условий всех входных переходов для каждой из позиций p_i, от которой идет дуга к переходу t_i, включая и условия для самого t_i.

$$\chi(t_i) = \bigwedge_{j=1}^{l} x(t_j),\ t_j \in I(p_l),\ t_i \in O(p_l).$$

Для перехода t_1 условия будут следующими: $\chi(t_1) = x(t_1) \wedge x(t_2) \wedge x(t_4) \wedge x(t_6)$. Пример определения входных условий для перехода t_1 приведен на рис. 6. $\chi(t_2)$ определяются аналогичным образом.

3. Из $\chi(t_i)$ исключаются условия, принадлежащие переходам, к которым идут дуги из p_k:

$$x(t_j),\quad t_j \in O(p_к).$$

Для t_1 это будут условия $x(t_2)$, как показано на рис. 7.

4. Составляются ступеньки, условие которых формируется из $\chi(t_i)$, а выход определяется позицией p_k.

Для рассматриваемого примера (ж/д стрелка) будут получены следующие условия:

$$\begin{aligned} \chi(t_1) &= x(t_1) \wedge x(t_4) \wedge x(t_6) = \mathrm{I}0.3 \wedge \mathrm{I}0.2 \wedge \overline{\mathrm{I}0.5}; \\ \chi(t_2) &= x(t_2) \wedge x(t_3) \wedge x(t_6) = \mathrm{I}0.1 \wedge \mathrm{I}0.4 \wedge \overline{\mathrm{I}0.5}. \end{aligned} \qquad (10)$$

Язык лестничной логики (Ladder Diagram) предлагает инженеру в наглядной и интуитивно понятной графической форме представить логические операции как электрическую цепь с замкнутыми и разомкнутыми контактами. При этом программа представляет собой электрическую шину с отходящими от нее ступень-

$$D_p = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix}$$

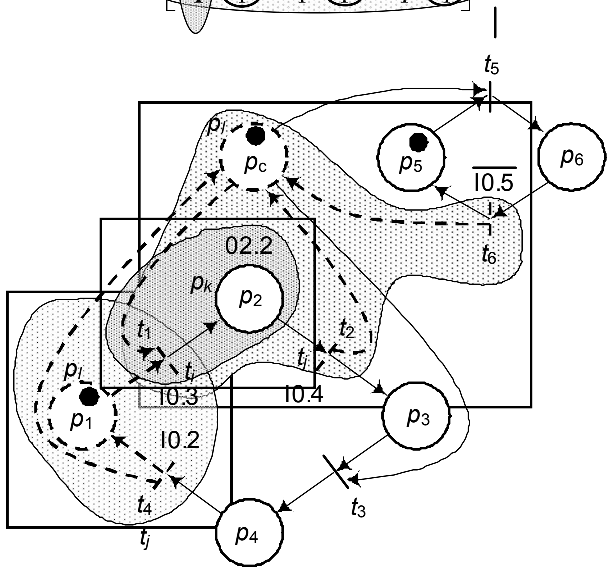

Рис. 6. *Определение условий для перехода t_1*

$$D_p = \begin{bmatrix} -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 \\ -1 & 1 & -1 & 1 & -1 & 1 \end{bmatrix}$$

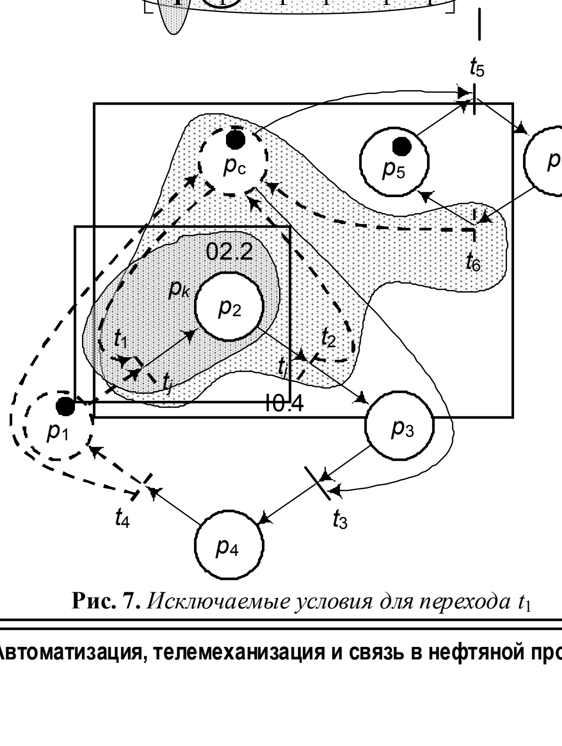

Рис. 7. *Исключаемые условия для перехода t_1*

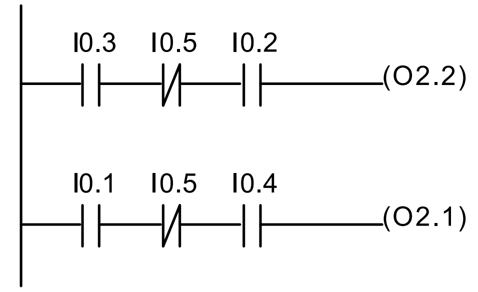

Рис. 8. *Алгоритм управления на языке LD*

ками электрических цепей, протекание или отсутствие тока в которых соответствует совокупности результатов логических операций, установленных на этой цепи контактов реле. Контакты бывают нормально замкнутые и нормально разомкнутые, что можно сопоставить с нормально замкнутыми и нормально разомкнутыми кнопками в электрических цепях (таблица).

Элемент LD	Описание
—| |—	Нормально разомкнутый контакт: разомкнут при значении false, назначенной ему переменной, и замыкается при значении true
—|/|—	Нормально замкнутый контакт: напротив, замкнут, если переменная имеет значение false, и разомкнут, если переменная имеет значение true
—( )	Итог логической цепочки копируется в целевую переменную

Условия (10) на языке LD примут вид, как показано на рис. 8.

Заключение

Предложенный в данной работе метод позволяет для простой сети Петри, представляющей поведение промышленной системы, синтезировать алгоритм на языке лестничной логики. Рассмотрен пример железнодорожной стрелки, для которой построена дискретно-событийная модель и в соответствии с теорией супервизорного управления синтезирован супервизор для заданного набора ограничений. Показано применение предложенного авторами метода к полученной модели. Полученную программу на языке LD можно записать в ПЛК в качестве алгоритма управления исследуемой системы.

В данной работе предполагалось, что все события в системе являются контролируемыми и наблюдаемыми, тогда как вопросы возникновения неконтролируемых событий, а также существования тупиков и циклов рассмотрены не были, и являются направлениями для дальнейших исследований.

ЛИТЕРАТУРА

1. *Wonham W., Ramadge P. The Control of Discrete Event Systems. Proceedings of the IEEE. – 1989.*
2. *Питерсон Дж. Теория сетей Петри и моделирование систем / пер. с англ. – 1984.*

3. Котов В.Е. Сети Петри. – М.: Наука, 1984.
4. Таль А.А., Юдицкий С.А. Иерархия и параллелизм в сетях Петри // АиТ. – 1982. – С. 111–139.
5. Holloway L.E., Krogh B.H. Synthesis of Feedback Logic for a Class of Controlled Petri Nets. IEEE Trans. Automat. Contr., 35. – 1990. – P. 514–523.
6. Holloway L., Krogh B., Giua A. A Survey of Petri Net Methods for Controlled Discrete Event Systems. Discrete Event Dynamic Systems: Theory and Application. – 1997. – P. 151–190.
7. Feedback Control of Petri Nets Based on Place Invariants / K. Yamalidou, J. Moody, M. Lemmon, P. Antsaklis // Automatica. – 1996. – № 32(1). – P. 15–28.
8. Амбарцумян А.А. Супервизорное управление структурированными динамическими дискретно-событийными системами // АиТ. – 2009. – С. 156–176.
9. Амбарцумян А.А., Томилин Е.Е. Метод прямого синтеза супервизора для структурированных дискретно-событийных систем // АиТ. – 2010. – С. 168–188.
10. Амбарцумян А.А. Моделирование и синтез супервизорного управления на сетях Петри для рассредоточенных объектов. I. Механизм взаимодействия и базовый метод // АиТ. – 2011.
11. Амбарцумян А.А. Моделирование и синтез супервизорного управления на сетях Петри для рассредоточенных объектов. II. Метод синтеза супервизора по множеству последовательностей общего вида // АиТ. – 2011.