Formalized Scheme of Technical Documentation Based on the Accounting Process and Control of Automatic and Telemechanics Devices

Baratov Dilshod Khamidullaevich, Aripov Nazirjon Mukaramovich

Abstract—The article discusses the problem of the synthesis of algorithmic mapping systems of railway automation and remote control based on the process of accounting and control of automation devices and telemechanics. This process is described using a formalized description language. Considering the verification of some properties of control algorithms and their transformation in order to optimize the structure of the process of accounting and control of automation and remote control devices, the language of logic circuits of algorithms was chosen. As a formalized language, logical schemes of algorithms were chosen. A three-dimensional cube was used as a graphic method for representing the process of recording and controlling railway automation and telemechanics devices.

A method has been developed for minimizing the states of logic circuits of algorithms, based on finding the minimum set of terms with the smallest number of logical conditions that allow one to obtain the entire output value. Application of the proposed methods allows to reduce the volume of simulation models and, accordingly, the number of errors in the simulation program.

Accounting and control of railway automation and remote control devices, logic circuits of algorithms, state minimization method, technical documentation

INTRODUCTION

In connection with the need for broad modernization, reconstruction and replacement of railway automation devices, an important task is to improve the quality of the process of control and accounting of railway automation and remote control devices (CARCD). The existing process technology CARCD does not ensure the adoption of quick and effective decisions.

To solve this problem in this work it is proposed to create a model of CARCD as an electronic document flow of technical documentation for alarm, centralization and blocking devices. In this connection, a survey was made of the actual processes of creating, verifying and using technical documentation in automation and remote control systems. This made it possible to identify document flow scenarios and protocols for technical document properties. [1-5].

CARCD will be described using a formalized description language. These languages include the language of Petri nets, because it is close to the Turing machines in its descriptive capabilities and, therefore, can be used to represent any algorithms. However, check some properties of the control algorithms and transform them to optimize the structure CARCD advantageously carried out with the use of other languages, including the language of the logical schemes of algorithms (LSA) and some its modifications [6-9].

LSA is used as a language for setting algorithms for the operation of software control devices [10-12].

The LSA searches for an algorithm for processing some initial information, that is, in the selection of individual operations, or acts of the algorithm, and the search for the order of their execution. Each such act (operation) in the LSA is associated with an operator, denoted by capital Latin letters A, B, C ... Different operators may be denoted by different letters or by the same letter, but with different indices: A1, A2, ..., B1, B2 ... If the operator depends on parameters, then these parameters can be set as indices Ai, Aij, Aijk ... or in brackets: A(i), A(ij), A(ijk) ... Operators with different parameters, they perform actions on different parts of the original or intermediate data, i.e., on different parts of the processed information.

The process of accounting and control of railway automation and remote control devices is described as follows:

\[ p_g \in P, \ g = \overline{1}, \overline{G} \]

where \( p_g \) – is the device (device), the set forms a set of devices. Also determined by the parameters of the device:

\[ h_{g,m} \in H_g, \ m = \overline{1}, M \]

\[ p_{g,m} \] – the instrument parameter, the instrument parameter set forms the sets of all the considered parameters of the instrument (each m parameter is entered in its instrument position).

Definition 1. The set of operations and checks of logical conditions performed in a specific sequence in the CARCD process is an algorithm. \( A_g \).

Definition 2. The operation \( O_p \) is an elementary action to account for and control devices from the set of \( S \). All operations performed in the process of accounting and control \( s_g \in S \), form a set of \( O = \{ O_p \}, \overline{1}, \overline{P} \). The index of the operation specifies the number of the participant and the algorithm, as well as its individual number in the sequence of entries.

Revised Version Manuscript Received on 14 February, 2019.

Baratov Dilshod Khamidullaevich, Department of automation and telemechanics in railway transport” Tashkent Institute of railway engineers
Aripov Nazirjon Mukaramovich, Department of automation and telemechanics in railway transport” Tashkent Institute of railway engineers

Retrieval Number: C11000283S19/190BEIESP

Published By:
Blue Eyes Intelligence Engineering & Sciences Publication

International Journal of Engineering and Advanced Technology (IJEEAT)
ISSN: 2249 – 8958, Volume-8, Issue-3S, February 2019

479
Let us introduce the symbolism of the record of private algorithms of CARCD in the language of logic schemes of algorithms (LSA) with regard to the generalized formalized scheme [13]. The main elements are operators corresponding to operations \( O_p \), logical conditions \( \alpha_k, k = 1, K \), marked with arrows \( \alpha_i \uparrow p, p = 1, P \), where \( p \) is the index of the arrow. The transition with a false value \( \alpha_k \) is carried out to the element of the LSA, marked with an arrow with the same index \( \downarrow p \).

The set \( A = \{\alpha_k\}, k = 2 \) includes probabilistic logical conditions of the form

\[
\alpha_k = \begin{cases} 
1 \text{— positive result;} \\
0 \text{— otherwise;}
\end{cases}
\]

The sequence of execution of statements in the LSA is determined by the order in which they are written. For example, A11A12A13 means that the operator A11 is first executed, then A12, and then A13. The order of execution of operators in the LSA can be strictly fixed — a linear algorithm — or depending on certain conditions — a branched algorithm. In the latter case, the LSA uses logical conditions denoted by small Latin letters, \( p, q, r \) ... Like operators, different logical conditions (LC) are denoted by different letters or by the same letter, but with different indices.

Logical conditions may depend on several variables. Logical conditions depending on the values of the function \( n \) variables are denoted by

\[
A_n = A_1 \downarrow^1 A_2 k 1 A 21 A 22 A 2 \uparrow^1 A 23 A 2 \uparrow^2 k 1 A 13 \omega \uparrow^1 k 2 \downarrow^2 A 24 A 3 \uparrow^4 k 3 A 1 k 4 k 1 \times A 41 A 42 k 6 A 43 k 7 A 51 A 52 A 53 k 7 A 61 A 62 A 63 k 1 A 32 \omega \uparrow^5 \downarrow^4 A 31 \downarrow^5 A 3 A 3 A 4 k 4 k 5 A 35 \downarrow^k
\]

where \( \alpha_1, \alpha_2, \alpha_3 \) — logical conditions, the probability of fulfillment of which depends on the current value of the quality indicator CARCD.

Logical scheme determines the order of execution of operators depending on the value of the LC included in it.

The algorithm begins with the execution of the leftmost operator of the scheme. After the operators of the scheme \( A_n \) are executed, it is determined which operator of the scheme should follow after it. After the operator A11, the operator of the scheme that is immediately to his right (A12) must be executed. After the logical condition \( \alpha_1 \), two cases are possible: if the condition being checked is satisfied, then operator A23 on the right must be executed; if it is violated, the operator A24 is executed, to which an arrow leads, starting after this condition.

The operation of the algorithm ends when the last executed statement A63 contains an instruction to terminate the operation of the algorithm.

Each elementary operation of the AU algorithm, in turn, is represented by a lower level algorithm in its alphabet of operators. This is achieved by the construction of a hierarchical structure of the description of the CARCD.

\[
\alpha = \{\alpha_1, \alpha_2, \ldots, \alpha_6\}
\]

(3)

It is considered that logical conditions can take only two values: the condition being checked is satisfied \( (\alpha = 1) \) or not \( (\alpha = 0) \). Depending on the value of the currently checked LU, the further order of execution of the operators and LC is determined.

Often among logical conditions it is advisable to select those that always take a zero (false) value, that is, identically false logical conditions. Identical logical conditions do not require verification. We denote them by \( \omega \). Operators and LUs are basic, and identically false logical conditions are auxiliary members of the logical scheme of the algorithm.

Each LU has an arrow. The beginning of the i-th arrow (denoted by \( \uparrow^i \)) is to the right of the logical condition, and its end (denoted by \( \downarrow^i \)) is to the left of the LSA member that must be met if the LC takes a zero value.

LSA are called expressions made up of operators and a LC following each other, as well as numbered arrows arranged in a certain way. The logic scheme of the algorithm is some way of describing the algorithm for solving the problem [14].

Description of the algorithm using logic circuits is the first step in the formalization of the algorithm. This stage is preceded by a meaningful description of the algorithm. The logic scheme of the algorithm allows both formal and substantive equivalent transformations.

As a result of the analysis of the processes included in CARCD at all levels, an LSA of the form:

\[
A_n = \sum_{m} f_m (\alpha_1, \alpha_2, \alpha_3) Z_m
\]

(5)

where, \( Z_1, Z_2, \ldots, Z_9 \) value after LC function.

In summary:

\[
A_n = \sum_{m} f_m (\alpha_1, \alpha_2, \alpha_3) Z_m
\]

(6)

where, \( m = 1, M \).

Thus, the distribution of LC values in a logical scheme determines the order in which the operators included in this scheme are executed.

Since each of the LC can take only two values - 0 and 1, the maximum number of unique sets of LC, and, therefore, the number of rows in the truth table, can be determined by the formula:

\[
N = 2^n
\]

(7)
where \( b \) – base of the number system (all LCs can take only one of two possible values); \( \alpha \) – LC number.

In the logical scheme (4), the order of execution of operators depending on the values of the LC is as follows:

1) if \( \alpha_1 = 0, \alpha_2 = 0 \) and \( \alpha_3 = 0 \), then \( k2A24A31A33A34k4k5A35 \);

2) if \( \alpha_1 = 0, \alpha_2 = 0, \alpha_3 = 1 \), then \( k2A24k3A14k1A41A42k6A43k7A51A52A53k7A61A62A63k1A32A33A34k4k5A35 \);

3) if \( \alpha_1 = 0, \alpha_2 = 1, \alpha_3 = 0 \), then \( k2A24A31A33A34k4k5A35 \);

4) if \( \alpha_1 = 0, \alpha_2 = 1, \alpha_3 = 1 \), then \( k2A24k3A14k1A41A42k6A43k7A51A52A53k7A61A62A63k1A32A33A34k4k5A35 \);

5) if \( \alpha_1 = 1, \alpha_2 = 0, \alpha_3 = 0 \), then \( A23A24A31A33A34k4k5A35 \);

6) if \( \alpha_1 = 1, \alpha_2 = 0, \alpha_3 = 1 \), then \( A23A24k3A14k1A41A42k6A43k7A51A52A53k7A61A62A63k1A32A33A34k4k5A35 \);

7) if \( \alpha_1 = 1, \alpha_2 = 1, \alpha_3 = 0 \), then \( A23k1A13A12k1A22 \);

if \( \alpha_1 = 1, \alpha_2 = 1, \alpha_3 = 1 \), then \( A23k1A13A12k1A22 \).

The truth table of \( A_U \) is presented in table 1.

<table>
<thead>
<tr>
<th>Set number</th>
<th>( \alpha_3 )</th>
<th>( \alpha_2 )</th>
<th>( \alpha_1 )</th>
<th>( A_U ) algorithm value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>( k2A24A31A33A34k4k5A35 )</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>( k2A24k3A14k1A41A42k6A43k7A51A52A53k7A61A62 \times A63k1A32A33A34k4k5A35 )</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>( k2A24A31A33A34k4k5A35 )</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>( k2A24k3A14k1A41A42k6A43k7A51A52A53k7A61A62 \times A63k1A32A33A34k4k5A35 )</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>( A23A24A31A33A34k4k5A35 )</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>( A23A24k3A14k1A41A42k6A43k7A51A52A53k7A61A62 \times A63k1A32A33A34k4k5A35 )</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>( A23k1A13A12k1A22 )</td>
</tr>
<tr>
<td>7</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>( A23k1A13A12k1A22 )</td>
</tr>
</tbody>
</table>

From the possible values of \( A_U \), we choose similar LSA. It is easy to see that the values of 0, 2 and 4 sets (with the difference of the operator \( A23 \) and \( k2 \)), 1, 3 and 5 sets (with the difference of the operator \( A23 \) and \( k2 \)), 6 and 7 sets are similar. The same parts of the LSA are denoted as follows.
Table 2 The same parts of the LSA $A_1U$

<table>
<thead>
<tr>
<th>Set number</th>
<th>LSA</th>
<th>Selection element</th>
<th>Name of the common part of the LSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$A_24A_31A_33A_34k_4k_5A_35$</td>
<td>$k_2$</td>
<td>$L_1$</td>
</tr>
<tr>
<td>1</td>
<td>$k_2A_24k_3A_14k_1A_41A_42k_6A_43k_7A_51A_52A_53k_7A_61A_62 \times A_63A_1A_32A_33A_34k_4k_5A_35$</td>
<td>$k_2$</td>
<td>$L_2$</td>
</tr>
<tr>
<td>2</td>
<td>$k_2A_24A_31A_33A_34k_4k_5A_35$</td>
<td>$k_2$</td>
<td>$L_1$</td>
</tr>
<tr>
<td>3</td>
<td>$k_2A_24k_3A_14k_1A_41A_42k_6A_43k_7A_51A_52A_53k_7A_61A_62 \times A_63A_1A_32A_33A_34k_4k_5A_35$</td>
<td>$k_2$</td>
<td>$L_2$</td>
</tr>
<tr>
<td>4</td>
<td>$A_23A_24A_31A_33A_34k_4k_5A_35$</td>
<td>$A_23$</td>
<td>$L_1$</td>
</tr>
<tr>
<td>5</td>
<td>$A_23A_24k_3A_14k_1A_41A_42k_6A_43k_7A_51A_52A_53k_7A_61A_62 \times A_63A_1A_32A_33A_34k_4k_5A_35$</td>
<td>$A_23$</td>
<td>$L_2$</td>
</tr>
<tr>
<td>6</td>
<td>$A_23k_1A_13A_12k_1A_22$</td>
<td>-</td>
<td>$L_3$</td>
</tr>
<tr>
<td>7</td>
<td>$A_23k_1A_13A_12k_1A_22$</td>
<td>-</td>
<td>$L_3$</td>
</tr>
</tbody>
</table>

We construct a transition table for this LSA.

Table 3 LSA transition table $A_1U$

<table>
<thead>
<tr>
<th>$\alpha_1\alpha_2$</th>
<th>00</th>
<th>01</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha_3$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>$k_2L_1$</td>
<td>$k_2L_1$</td>
<td>$A_23L_1$</td>
<td>$L_3$</td>
</tr>
<tr>
<td>1</td>
<td>$k_2L_2$</td>
<td>$k_2L_2$</td>
<td>$A_23L_2$</td>
<td>$L_3$</td>
</tr>
</tbody>
</table>

With the graphical method, each set of values of an LC corresponds to a certain point of n-dimensional space. The coordinates of the vertices of the n-dimensional cube correspond to the sets of values of the LC, and their designations are assigned the values of AU on these sets. Since each of the LCs can take only two values: 0 and 1, each edge connecting two adjacent vertices, the sets of which differ by one variable, has a unit length. Therefore, a n-dimensional cube is called a unit cube.

The number of vertices of a n-dimensional cube is equal to the number of rows in the truth table, and the number of coordinate axes is equal to the number of n LC.

The three-dimensional cube corresponding to the LC, the truth table given earlier (Table 1), is shown in Figure 1. The top of the cube and the cell of Table 3, the contents of which describes the same set of variables (No. 6), are highlighted by a dotted line. Similarly, the top of the cube is matched with the rest of the truth table cells.
Using the coordinate method, the LSA AU is defined as a state coordinate map, called a Carnot map. The total number of cells in the Carnot map corresponds to the number of sets of the algorithm AU.

### Fig.2. Carnot map for LSA AU

Designations in Carnot map for LSA $A_U$:

- $\alpha_i \rightarrow (\alpha_i = 1), \overline{\alpha_i} \rightarrow (\alpha_i = 0)$

In brackets inside the cells are the numbers of the corresponding sets from the truth table (Table 3).

### Fig.3. LSA Truth Table Sets AU

The task of minimizing the LSA $A_U$ is to find the minimum set of terms with the smallest number of LСs, allowing to get all the output value.

In (5) $Z_1, Z_2, ..., Z_9$ is not a variable. If we accept the condition that the symbol should always remain at the end of the addendum, and the inversion operation should not be applied to it, we can formally operate with it as with the variable.

All cells containing the same LSA value are combined into closed areas, each area must represent a rectangle with the function of the LС.

To obtain the minimum form of the function, closed areas cover cells with the same LSA values and when recording members, the values of the LC are taken with a constant value within the respective areas.

So, from Figure 3 we obtain the function of conditions $f(\alpha_1, \alpha_2, \alpha_3)$ for LSA $A_U$:

$$f = \alpha_2\overline{\alpha_3} \lor \alpha_1\overline{\alpha_3} \lor \overline{\alpha_1}\alpha_3$$

(8)

Having executed minimization, we receive:

$$f = \alpha_2\alpha_3 \lor \alpha_1\overline{\alpha_3} \lor \alpha_1\alpha_3$$

(9)

### CONCLUSION

The presented formalized scheme will make it possible to determine the sets of source data from the algorithmic and parametric mapping of the CARCD. The main feature of CARCD algorithms is the presence of parallel branches.

The formalized scheme provides sufficient flexibility for the description of the CARCD, since it is based on the algorithmic mapping of the system. In accordance with this, the methodology of formalization should be aimed primarily at identifying and describing CARCD algorithms.

Using the language of logic circuits of algorithms for identifying and describing CARCD processes on railway transport allowed us to develop a new survey methodology aimed at identifying and describing CARCD algorithms.

The proposed formalized scheme of the CARCD process using parallel logic circuits of algorithms provides formalization of the transition procedures to automated technology. The formalized scheme makes it possible to carry out high-quality and accelerated operational research of the electronic document flow CARCD in the systems of automation and telemechanics on the railway transport.

### REFERENCES

1. Баратов Д. Х., Арипов Н. М. Разработка системы учета и контроля электротехнических комплексов железнодорожной автоматики и телемеханики //Фундаментальные и прикладные научные исследования: актуальные вопросы, 2018. С. 64.


Formalized Scheme of Technical Documentation Based On the Accounting Process and Control of Automatic and Telemechanics Devices

7. Седых Д. В., Суханов С. А. Применение отраслевого формата технической документации на устройства железнодорожной автоматики и телемеханики для интеграции приложений //Известия Петербургского университета путей сообщения, 2005. №. 3. С.64-72.


10. Денисов Б. П., Рубинштейн Н. И., Растегаев С. Н., Воробей Н. Ю. Автоматизация проектирования систем железнодорожной автоматики и телемеханики на базе АРМ-ПТД версии 6 //Актуальные вопросы развития систем железнодорожной автоматики и телемеханики, 2013. №. 1. С.33-38.


