

,

\_\_\_\_\_ ( )

\_\_\_\_\_ ( )

\_\_\_\_\_ ( )

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_ ( )

:

\_\_\_\_\_ , \_\_\_\_\_ -18-3

\_\_\_\_\_ ( , )

123 - \_\_\_\_\_ ' \_\_\_\_\_ ( )

\_\_\_\_\_ - \_\_\_\_\_ ( - - )

\_\_\_\_\_ ( )

: \_\_\_\_\_ ( , , )

\_\_\_\_\_ ( ) \_\_\_\_\_ ( , )

,

( )

123 - '

( )

-

( - - )

( )

:

“ ” 20 .

( , ' )

1.

“ 30 ” 2020 . 478

2.

18 2020 .

3.

TypeScript

Electron Angular

: Windows (Windows 7, 8, 10), macOS (El Capitan, Sierra, High Sierra),

Ubuntu.

4.

,

5. \_\_\_\_\_ , \_\_\_\_\_ , \_\_\_\_\_ , \_\_\_\_\_  
 ( ) 15 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_  
 \_\_\_\_\_

6. \_\_\_\_\_ .1) \_\_\_\_\_ ( \_\_\_\_\_ , \_\_\_\_\_ , \_\_\_\_\_ , \_\_\_\_\_ ) \_\_\_\_\_

	( _____ , _____ , _____ , _____ )		

1		31.03.20 – 05.04.20	
2		06.04.20 – 11.04.20	
3		12.04.20 – 19.04.20	
4		20.04.20 – 27.04.20	
5		28.04.20 – 05.05.20	
6		06.05.20 – 11.05.20	
7		12.05.20 – 13.05.20	
8		14.05.20 – 15.05.20	

30 2020 .

\_\_\_\_\_ ( ) \_\_\_\_\_

\_\_\_\_\_ ( ) \_\_\_\_\_ ( \_\_\_\_\_ , \_\_\_\_\_ , \_\_\_\_\_ ) \_\_\_\_\_

: 70 ., 32 ., 13 ., 1

., 17 .

, , ,  
, , ELECTRON,

ANGULAR.

, : .  
:

: .  
:  
( ).

: ,  
,  
.  
:  
,  
,

, ,

.

## ABSTRACT

Master's thesis: 70 pages, 32 figures, 13 tables, 1 appendix, 17 sources.

FINITE STATE MACHINE, STATE DIAGRAM, TRANSITION CONDITIONS, ORTHOGONAL FUNCTIONS, CROSSPLATFORM, ELECTRON, ANGULAR.

Object of research: graph models of finite state machines (FSM).

Subject of research: analysis of the correct formation of the transition conditions function in state diagram of FSM for automated synthesis of microprogram systems and logic control systems.

The purpose of research: to develop a formal method for analyzing the correctness of the transition conditions in graph models of FSM for automated synthesis of microprogram systems and logic control systems in computer-aided design.

Scientific novelty: a formal method was offered for the transition conditions correctness analysis in FSM's state diagram, it uses the terms of the orthogonal function of transition conditions from the considered state to build a table for analysis of transition conditions completeness and consistency. This allows to determine the FSM's states with incorrectly described transitions by formal signs.

Practical significance: the developed method makes it possible to reduce the search time for sources of incorrect synthesis, when there are errors in describing the conditions of state transitions at the stages of the FSM language model and state diagram formation.

	,	,	,	7
	.....			8
1			.....	10
1.1		.....		10
1.2			.....	16
1.3	.....			18
2			.....	21
2.1			.....	21
2.2		.....		25
3			.....	35
3.1			.....	35
3.2	.....			37
4			....	48
4.1	.....			48
4.2		.....		51
4.3	.....			54
	.....			59
		.....		61
			.....	63

, , ,

- -  
- ,  
-  
-  
-  
-  
-  
-  
-  
-  
-  
-

FSM - ( ., Finite State Machine)

,  
 .  
 -  
 , ,  
 , -  
 -  
 .  
 ,  
 :  
 ( ), - ( ).  
 ,  
 ,  
 ,  
 .  
 ,  
 .  
 ,  
 ,  
 .  
 ,  
 ,  
 .  
 ,  
 ,  
 ( ) ,







[2].

$$Y = \{y_1, y_2, \dots, y_m\}, m = \overline{1, M}.$$

:

-

,

;

-

Y

X;

-

( )

,

-

;

-

,

.

:

-

$$D = \{d_1, \dots, d_H\},$$

;

-

$$R = \{r_1, \dots, r_Q\},$$

;

-

$$S = \{s_1, \dots, s_N\},$$

.

,

,

,

$$D \subseteq S \quad R \subseteq S;$$

-

$$Y = \{y_m\} \quad (m = \overline{1, M}),$$

$$S = \sum_{m=1}^M s_m(s)$$

,

m -

;

-

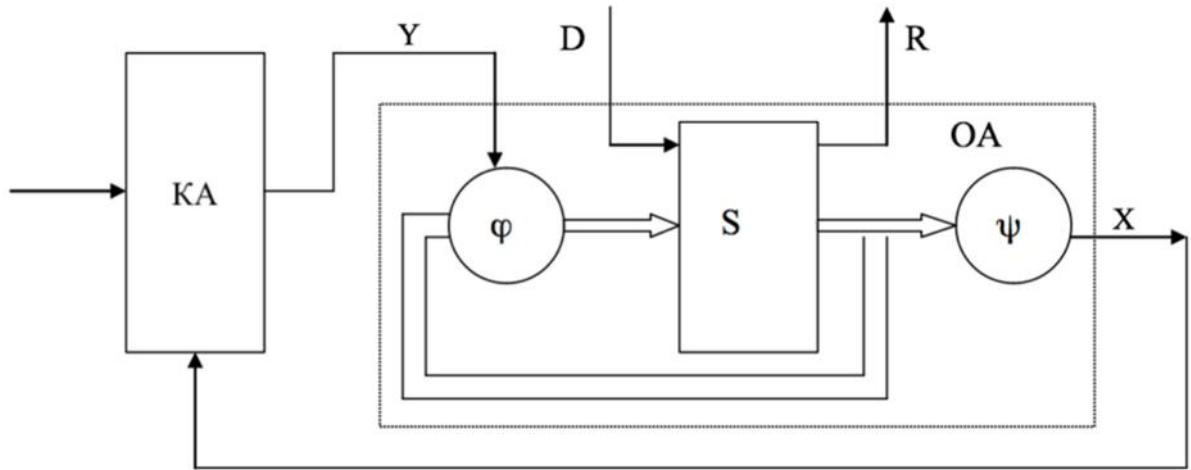
$$X = \{x_l\} \quad (l = \overline{1, L}), \quad x_l = x_l(s) \quad l =$$

.

D, R, S, Y,

X.

(1.2).



1.2 –

$S_1, \dots, S_N,$

R.

$Y = \{y_m\}$

Y,

[3, 4].

$y_{m_1} \quad y_{m_2},$

$S_{\alpha_1} := \phi_{m_1}(S_{\beta_1}, \dots, S_{w_1}); \quad S_{\alpha_2} := \phi_{m_2}(S_{\beta_2}, \dots, S_{w_2}),$

$\phi_{m_1}(S_{\beta_1}, \dots, S_{w_1}); \phi_{m_2}(S_{\beta_2}, \dots, S_{w_2})$

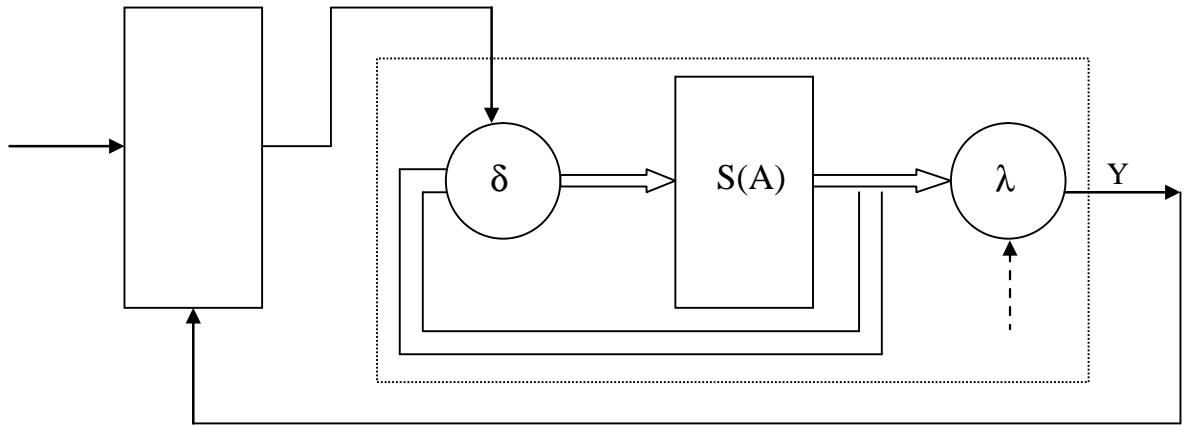
$S_{\alpha_1},$

$S_{\alpha_2} \in S.$

$$X_i := (S_{y_1}, \dots, S_{y_k}),$$

$$X = \{x_i\}.$$

( )  
 $y_1, \dots, y_L$  ( . 1.3).



1.3 –

[5, 6, 7].

δ λ.

S(A) –

( )

$x_1, \dots, x_L$

$y_1, \dots, y_M$

,

.

,

—

$s_1, \dots, s_N$

,

S

X,

.

,

[8].

,

,

,

.

.

(

)

(

).

:

,

.

(switch, case)

(if, elsif-else) [9].

,

[10].

[11]

[12].

,

—

,

.

{t, t+1, t+2},

$$Y(t) = g(X(t), Z(t)), Z(t+1) = f(X(t), Z(t)), \quad (1.1)$$

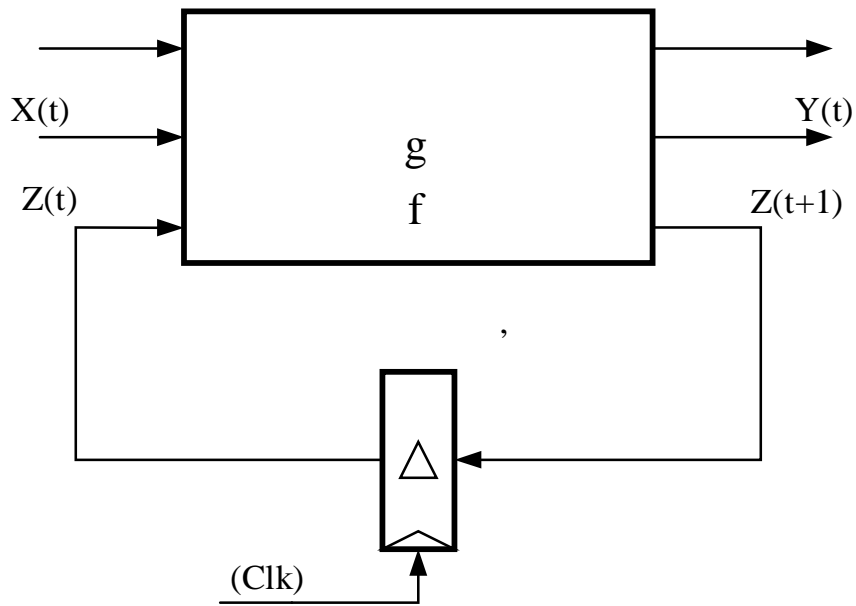
g -

, f -

$$Z(t+1) \equiv Z(t)$$

( )

1.4).



1.4 -

( )

( )

) [2].

(State Diagrams).

[12].

$$f_j(x_1, x_2, \dots, x_n) \quad ( \quad )$$

[13]

1.2

2000 .



Windows).

AJAX

Google Facebook,

iOS Android.

Windows –

Mac OS Linux.

Mono Qt,

C, C++ C#,

, - ,  
 ,  
 , NW.js Electron  
 Windows, Mac OS Linux ,  
 - [14].

Node.js

Node.js

Node.js

1.3

:



- , .
- (Y)
- as;
- ,
- ;
- ;
- ,
- ;
- ;
- 32, 64
- Windows, Mac OS Ubuntu;
- .

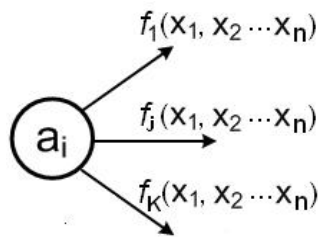
2

2.1

[12].

. 2.1  
K .

$a_i$



2.1 –

$f(x_1, x_2, \dots, x_n)$  , ( ):

$$f(x_1, x_2, \dots, x_n) = f_1(x_1, x_2, \dots, x_n) \vee \dots \vee f_j(x_1, x_2, \dots, x_n) \vee \dots \vee f_k(x_1, x_2, \dots, x_n). \tag{2.1}$$

, , :

$$\bigvee_{j=1}^K f_j(x_1, x_2, \dots, x_n) = 1 \tag{2.2}$$

$2^n$   
 $n -$  ( , ) ,  
 $f(x_1, x_2, \dots, x_n) = 1.$

( )  
 $\forall (f_g \cdot f_h = 0), g \neq h$  [2].

( , . ) ,  
 [15].

$f(x_1, x_2, \dots, x_n) = 1 -$   
 $f^* -$  (n-1) ,  $f^{**} -$  (n-2) ,  
 $f^{***} -$  (n-3) .  
 $f = f^* = f^{**} = f^{***} \equiv 1.$



, (n+1) , n  
 f(x<sub>1</sub>, x<sub>2</sub>, ..., x<sub>n</sub>) , (n+1) .

2.2 ( ) (2.5).

. 2.2 ( )

$$f(x_1, x_2, x_3) = \overline{x_1} \vee x_1 \overline{x_2} \vee x_1 x_2 \overline{x_3} ,$$

$$x_1 x_2 x_3 .$$

$$, f^* \neq 1 .$$

x <sub>1</sub> \x <sub>2</sub> x <sub>3</sub>	00	01	11	10
0	1	1	1	1
1	1	1	1	1

)

x <sub>1</sub> \x <sub>2</sub> x <sub>3</sub>	00	01	11	10
0	1	1	1	1
1	1	1	1	1

)

x <sub>1</sub> \x <sub>2</sub> x <sub>3</sub>	00	01	11	10
0	1	1	1	1
1	1	1	1	1

)



. 2.2 ( )

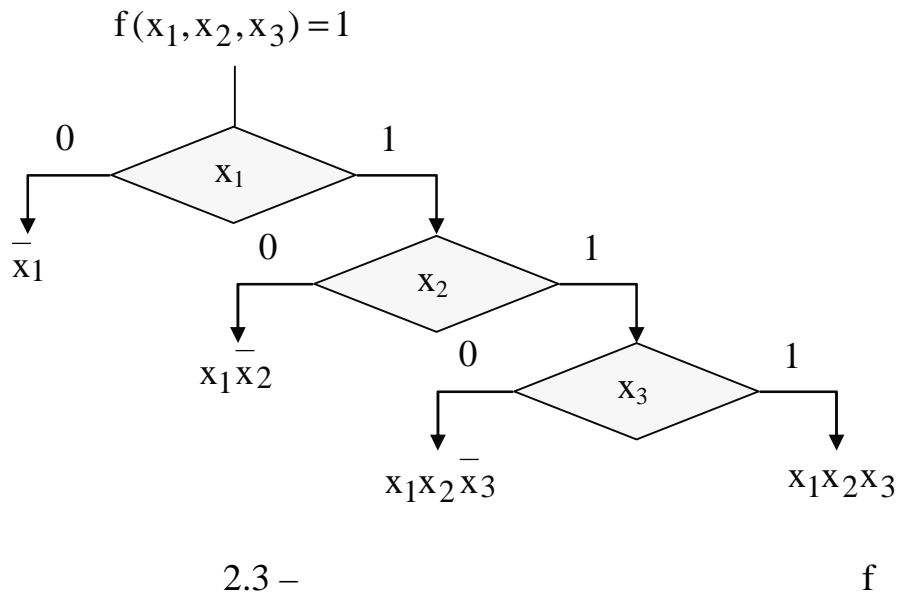
$$f(x_1, x_2, x_3) = \overline{x_1} \vee x_1 \overline{x_2} \vee x_1 x_2,$$

, 3. ,

(n+1) , .

2.2

( . 2.3).



( )

, . 2.2 ( ). 3 4

( ),  
(2.5).

$x_1, x_2, \dots, x_n$

$(m + 1), m -$  ( , ) .

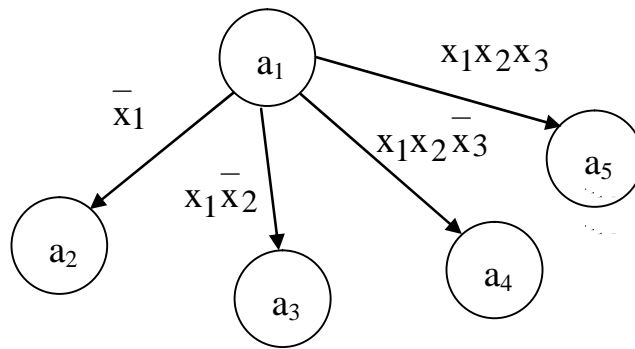
$a_i$

$(n+1), n -$  .

. 2.4

f.

1, 2, ...



2.4 -

f

f

:

$$f(x_1, x_2, x_3) = \bar{x}_1 \vee \bar{x}_1 \bar{x}_2 \vee x_1 x_2 \bar{x}_3 \vee x_1 x_2 x_3 \tag{2.6}$$

( . 2.1).

2.1 -

f

( )	$x_1 x_2 x_3$	f				
		$\bar{x}_1$	$\bar{x}_1 \bar{x}_2$	$x_1 x_2 \bar{x}_3$	$x_1 x_2 x_3$	
0	000	1	0	0	0	1
1	001	1	0	0	0	1
2	010	1	0	0	0	1

2.1

( )	$x_1x_2x_3$	f				
		$\bar{x}_1$	$x_1\bar{x}_2$	$x_1x_2\bar{x}_3$	$x_1x_2x_3$	
3	0 1 1	1	0	0	0	1
4	1 0 0	0	1	0	0	1
5	1 0 1	0	1	0	0	1
6	1 1 0	0	0	1	0	1
7	1 1 1	0	0	0	1	1

. 2.1

f

$f_n$ ,

(

),

f.

$$f_n(x_1, x_2, x_3) = \bar{x}_1 \vee x_1\bar{x}_2 \vee x_1x_2 \vee x_1x_2x_3 \tag{2.7}$$

. 2.5

$f_n$  (2.7).

$x_1 \backslash x_2x_3$	00	01	11	10
0	1	1	1	1
1	1	1	1	1

2.5 –

$f_n$



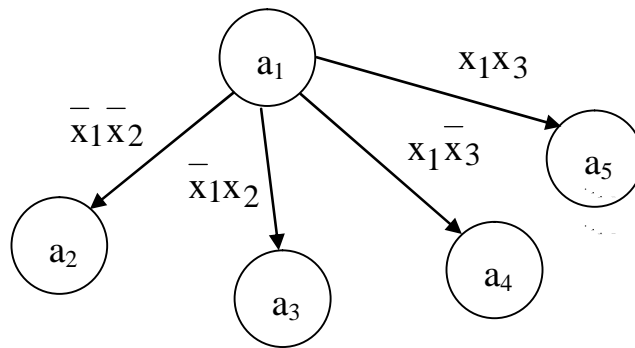
( . 2.6).

3 4 ( ),  
:

$$f_1(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1 \bar{x}_3 \vee x_1 x_3 \quad (2.8)$$

2.7

$f_1$ .



2.7 -

$f_1$

. 2.8.

$x_1 \backslash x_2 x_3$	00	01	11	10
0	1	1	1	1
1	1	1	1	1

2.8 -

$f_1$

( . 2.3).

2.3 –

$f_1$

( )	$x_1x_2x_3$	$f_1$				
		$\bar{x}_1\bar{x}_2$	$\bar{x}_1x_2$	$x_1\bar{x}_3$	$x_1x_3$	
0	000	1	0	0	0	1
1	001	1	0	0	0	1
2	010	0	1	0	0	1
3	011	0	1	0	0	1
4	100	0	0	1	0	1
5	101	0	0	0	1	1
6	110	0	0	1	0	1
7	111	0	0	0	1	1

. 2.3

$f_1$

$f_{1n}$ ,

(

1- 5,

$f_1$ ),

$f_1$ :

$$f_{1n}(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1 \bar{x}_3 \vee x_1 x_3 \quad (2.9)$$

. 2.9

$f_{1n}$ .

$x_1 \backslash x_2x_3$	00	01	11	10
0	1	1	1	1
1	1	1	1	1

2.9 –

$f_{1n}$

$f_{1n}$

2.4

0 4,

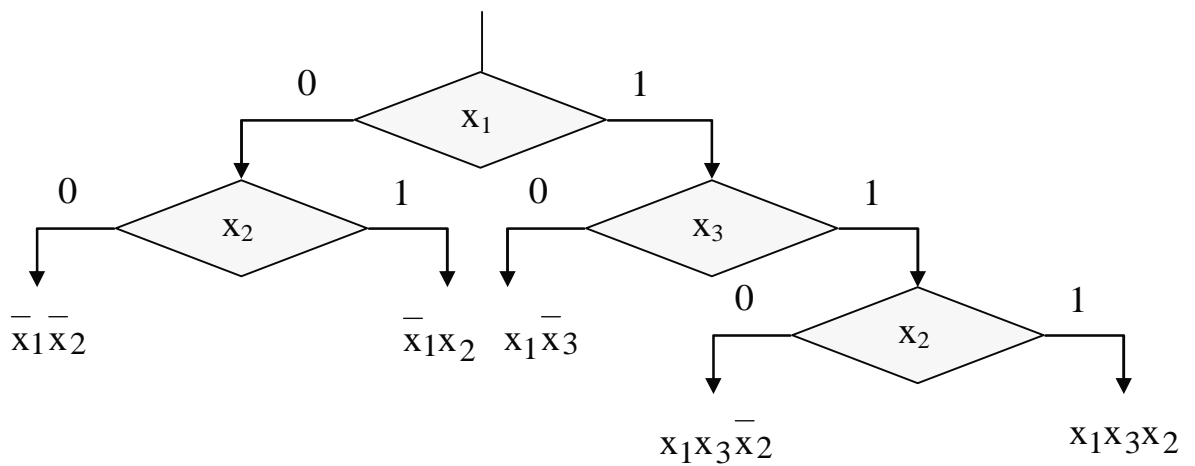
5 7

2.4 –

$f_{1n}$

( )	$x_1x_2x_3$	$f_{1n}$				
		$\bar{x}_1\bar{x}_2$	$\bar{x}_1x_2$	$x_1\bar{x}_3$	$\bar{x}_1\bar{x}_3$	
0	000	1	0	0	1	2
1	001	1	0	0	0	1
2	010	0	1	0	0	1
3	011	0	1	0	0	1
4	100	0	0	1	1	2
5	101	0	0	0	0	0
6	110	0	0	1	0	1
7	111	0	0	0	0	0

$f_2(x_1, x_2, x_3) = 1$



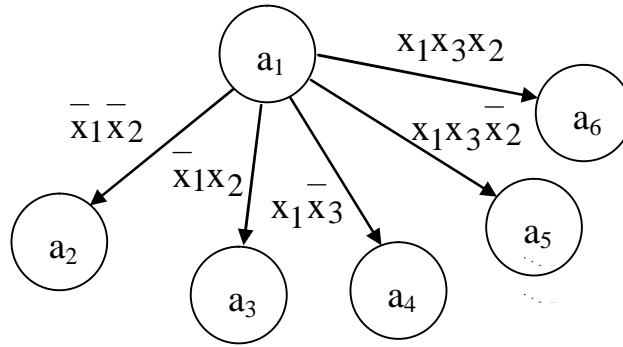
2.10 –

$f_2$

( . 2.10).

. 2.11

$f_2$ .



2.11 -

$f_2$

4

5

(

),

:

$$f_2(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1 \bar{x}_3 \vee x_1 x_3 \bar{x}_2 \vee x_1 x_2 x_3 \quad (2.10)$$

. 2.12.

$x_1 \backslash x_2 x_3$	00	01	11	10
0	1	1	1	1
1	1	1	1	1

2.12 -

$f_2$

( . 2.5).



2.5 –

$f_2$

( )	$x_1x_2x_3$	$f_2$					
		$\bar{x}_1\bar{x}_2$	$\bar{x}_1x_2$	$x_1\bar{x}_3$	$x_1x_3\bar{x}_2$	$x_1x_2x_3$	
0	000	1	0	0	0	0	1
1	001	1	0	0	0	0	1
2	010	0	1	0	0	0	1
3	011	0	1	0	0	0	1
4	100	0	0	1	0	0	1
5	101	0	0	0	1	0	1
6	110	0	0	1	0	0	1
7	111	0	0	0	0	1	1

. 2.5

$f_2$

$f_{2n}$ ,

( 1-

5

$f_2$ ),

$f_2$ :

$$f_{2n}(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1 \bar{x}_3 \vee x_1 x_3 \bar{x}_2 \quad (2.11)$$

. 2.13

$f_{2n}$ .

$x_1 \backslash x_2 x_3$	00	01	11	10
0	1	1	1	1
1	1	1	1	1

2.13 –

$f_{2n}$

$f_{2n}$

2.6

7

2.6 –

 $f_{2n}$ 

( )	$x_1x_2x_3$	$f_{2n}$				
		$\bar{x}_1\bar{x}_2$	$\bar{x}_1x_2$	$x_1\bar{x}_3$	$x_1x_3\bar{x}_2$	
0	000	1	0	0	0	1
1	001	1	0	0	0	1
2	010	0	1	0	0	1
3	011	0	1	0	0	1
4	100	0	0	1	0	1
5	101	0	0	0	1	1
6	110	0	0	1	0	1
7	111	0	0	0	0	0

3

3.1

( ) ( ).

,

:

- , ( (t));

- , ( (t+1));

- (t) - (t+1) ,

.

,

:

1.  $\epsilon$  , - , (t) - (t+1).

2. (t) (t)

,

(t)

- (t+1) ( . 2.5). ,

( ) ,

,

3. (t)

.

( .

. 2.1):

- ;
- ;
- .

· ,

, ( .

. 2.1).

: (t) (t) - (t+1),

· ,

1,

( ) (t).

4.

( . 2.1),

( ) · ,

(t) · ,

(t) ,

, (t) .

0, ,

> 1,

( . . 2.4).

,

(t),

(t) - (t+1).

5.

·

·

(t)

(t).

6.

t

t+1.

3.2

Y

X,

( )

$$X = \{x_1, x_2, x_L\}$$

( )

$$Y = \{y_1, y_2, y_M\}$$

,

$y_1, y_2, y_M$

,

$X(0), X(1), \dots, X(t)$

$Y(0), Y(1), \dots, Y(t)$

-

Y.

-

,

$x_l y_m$

( )

( )

,

,

,

.

:

-  $a_1$  ,

, ;

- , ,

$a_2, a_3, \dots$ ;

- , ,

;

- .

. 3.1,

,

,

,

,

.

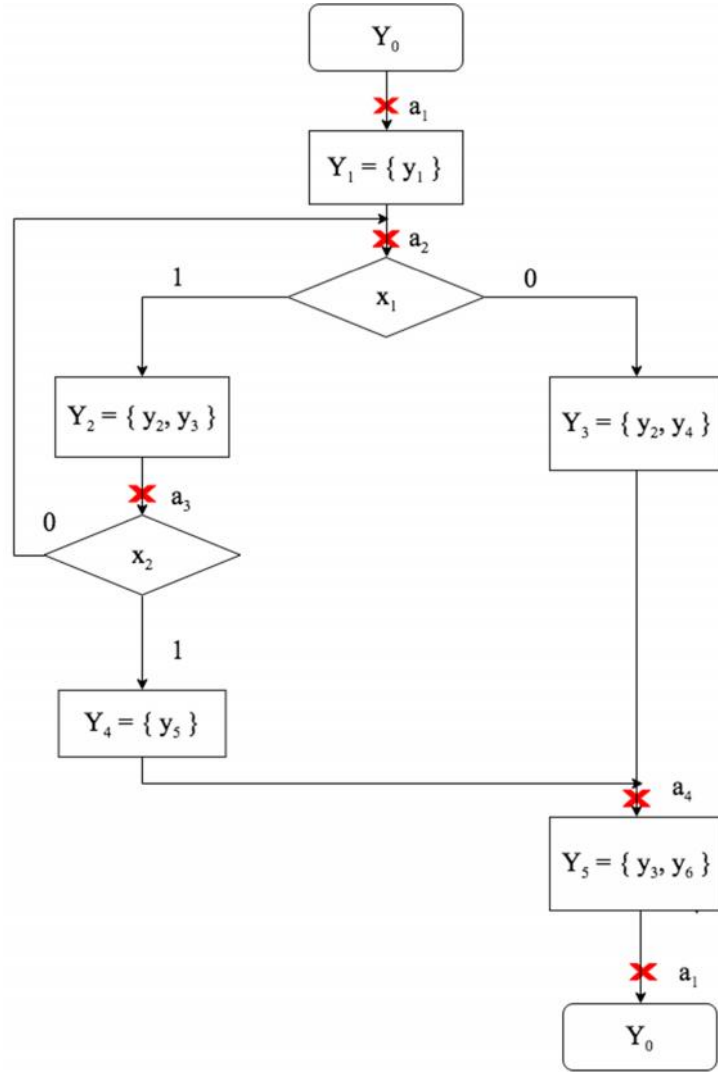
. 3.1

3.1.

3.1 –

h	$a_m$	$k(a_m)$	$a_s$	$k(a_s)$	$X_h$	$Y_h$	$F_h$
1	$a_1$		$a_2$		$1*Z$	$Y_1(y_1)$	
2	$a_2$		$a_3$		$x_1$	$Y_2(y_2, y_3)$	
3			$a_4$		$\bar{x}_1$	$Y_3(y_2, y_4)$	
4	$a_3$		$a_2$		$\bar{x}_2$	$Y_0$	
5			$a_4$		$x_1x_2$	$Y_4(y_5)$	
6	$a_4$		$a_1$		1	$Y_5(y_3, y_6)$	

3.1:  $h -$  ;  $a_m -$   
 $t; k(a_m) -$  ;  $a_m; a_s -$   
 $t+1; k(a_s) -$  ;  $X_h -$  ;  $h; Y_h$   
 $-$  ;  $a_m a_s$  ;  $h; F_h -$



3.1 -

$k(a_m), k(a_s) F_h.$  ;  $h$

$k(a_m), k(a_s) \quad F_h$

( )

( 3.2),

$a_1, a_2$

$a_m,$

$- a_s.$

3.2 –

h	$a_m$	$k(a_m)$	$a_s$	$k(a_s)$	$X_h$	$Y_h$	$F_h$
1	$a_4$	10	$a_1$	00	1	$Y_5(y_3, y_6)$	$T_2$
2	$a_1$	00	$a_2$	01	$1*Z$	$Y_1(y_1)$	$T_1$
3	$a_3$	11	$a_2$	01	$\bar{x}_2$	$Y_0$	$T_2$
4	$a_2$	01	$a_3$	11	$x_1$	$Y_2(y_2, y_3)$	$T_2$
5	$a_2$	01	$a_4$	10	$\bar{x}_1$	$Y_3(y_2, y_4)$	$T_2, T_1$
6	$a_3$	11	$a_4$	10	$x_1x_2$	$Y_4(y_5)$	$T_1$

(

(Z=0)

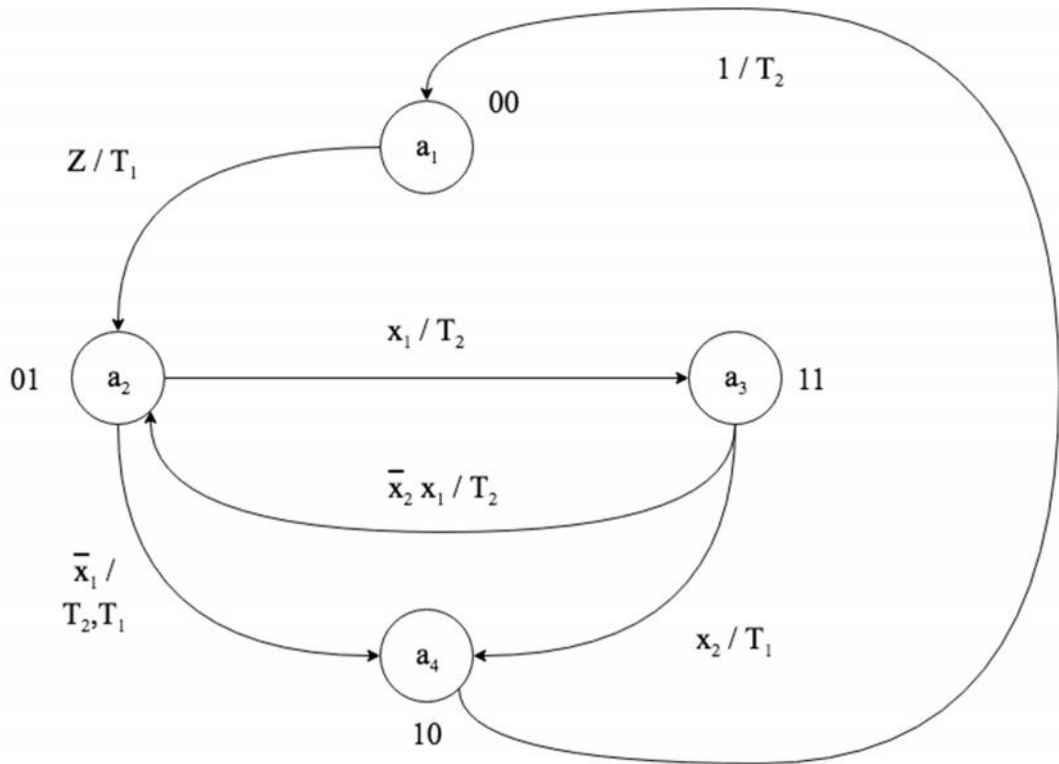
, Z=1),

Z.

3.2.

. 3.2.





3.2 -

-

. 3.2,

$k(a_1) = 00; k(a_2) = 01; k(a_3) = 11; k(a_4)$

= 10.

$k(a_i)$

$k(a_i)$

. 3.2.

$$: K^T = \frac{7}{6} = 1,17.$$

,

**h**

,

$$y_i (i = \overline{1,6}) \quad T_j (j = \overline{1,2}),$$

.

$$y_1 = 2; y_2 = 4 \vee 5; y_3 = 1 \vee 4; y_4 = 5; y_5 = 6; y_6 = 1;$$

$$T_1 = 2 \vee 5 \vee 6; T_2 = 1 \vee 3 \vee 4 \vee 5;$$

$$y_1 = a_1 * Z;$$

$$y_2 = a_1 x_1 \vee a_2 \bar{x}_1 = a_2;$$

$$y_3 = a_4 \vee a_2 x_1;$$

$$y_4 = a_2 \bar{x}_1;$$

$$y_5 = a_3 x_2;$$

$$y_6 = a_4;$$

$$T_1 = a_1 Z \vee a_2 \bar{x}_1 \vee a_3 x_2;$$

$$T_2 = a_4 \vee a_3 \bar{x}_2 x_1 \vee (a_2 x_1 \vee a_2 \bar{x}_1) = a_4 \vee a_3 \bar{x}_2 x_1 \vee a_2.$$

$$y_1 = \overline{(a_1 Z)} = \overline{(a_1 | Z)} = (a_1 | Z) | 1;$$

$$y_2 = a_2;$$

$$y_3 = \overline{a_4 \vee a_2 x_1} = \bar{a}_4 * \overline{a_2 x_1} = \bar{a}_4 | (a_2 | x_1);$$

$$y_4 = a_2 \bar{x}_1 = (a_2 | \bar{x}_1) | 1;$$

$$y_5 = a_3 x_2 = (a_3 | x_2) | 1;$$

$$y_6 = a_4;$$

$$T_1 = (a_1 | Z) | (a_2 | \bar{x}_1) | (a_3 | x_2);$$

$$T_2 = \bar{a}_4 | (a_3 | \bar{x}_2 | x_1) | a_2.$$

k(a<sub>1</sub>)

$$a(t + 1) = [a(t), x(t)], \quad (3.1)$$

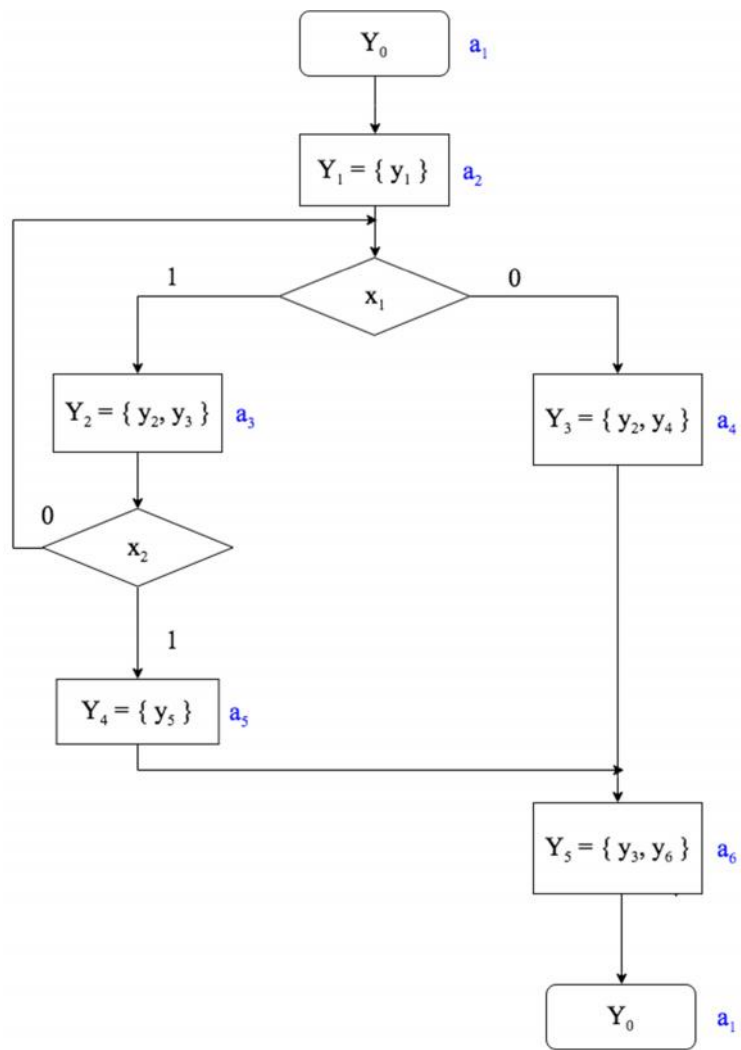
$$y(t) = [a(t)]. \quad (3.2)$$

a(t),

y(t) ,

:  
-  $a_1$   
;  
-  $a_2, a_3, \dots$   
-

. 3.3,



( 3.3).

$y_i$        $a_s(Y_h)$   
 $Y_h$        $a_s$   
 $3.3$        $k(a_m)$   
 $k(a_s), F_h$

3.3 –

h	$a_m$	$k(a_m)$	$a_s(Y_h)$	$k(a_s)$	$X_h$	$F_h$
1	$a_6$	101	$a_1(Y_0) \_$	000	1	$T_2^*$
2	$a_1$	000	$a_2(Y_1) y_1$	111	$1*Z$	–
3	$a_2$	111	$a_3(Y_2) y_2, y_3$	001	$x_1$	$T_1^*$
4	$a_3$	001			$\bar{x}_2 x_1$	$T_3^*, T_2^*, T_1^*$
5	$a_2$	111	$a_4(Y_3) y_2, y_4$	010	$\bar{x}_1$	$T_2^*$
6	$a_3$	001			$\bar{x}_2 \bar{x}_1$	$T_3^*$
7	$a_3$	001	$a_5(Y_4) y_5$	110	$x_2$	–
8	$a_4$	010	$a_6(Y_5) y_3, y_6$	101	1	–
9	$a_5$	110			1	$T_3^*$

D-

3.4

D -

$$d_{pk}^n = (0, 1, 0, 1); Q_{t+1} = D_t.$$

$Q_t - Q_{t+1}$	D	$d_{pk}^n$
0 - 0	0	0
0 - 1	1	1
1 - 0	0	0
1 - 1	1	1

3.4 -

D -

( ) ( 3.4).

3.4 -

T	$a_1 \ a_2 \ a_3$
$x_1$	$a_1 \ a_2 \ a_3$
$x_2$	$a_2 \ a_3 \ a_2$

D-

, :  
 -  $N_i$  ;  
 -  $N_1, N_2, \dots, N_i, \dots, N$  ;  
 -  $a_t$   $N_t$   
 $k_t=00\dots00$ ;  
 -  $n$   
 $: 00\dots01, 00\dots010, \dots$  ;  
 - , ,  
 , . . . ,  
 $N_1=1; N_2=3; N_3=2,$   $K_2=00$ ;  
 $K_3=01; K_1=10.$

3.5.

3.5 –

	10	00	01
	a <sub>1</sub>	a <sub>2</sub>	a <sub>3</sub>
1	10	00	01
2	00	01	00

3,

$$= \frac{3}{6} = 0.5.$$

( . 3.4).

: K<sub>1</sub>= 001; K<sub>2</sub>=010; K<sub>3</sub>=100.

3.6.

3.6 –

	001	010	100
	a <sub>1</sub>	a <sub>2</sub>	a <sub>3</sub>
1	001	010	100
2	010	100	010

3,

$$: = \frac{3}{9} = 0.33.$$

( ) , , , , , ( , ) .

3.4.

$$N_1 = 0, N_2 = 1, N_3 = 2,$$

$$K_1 = 00;$$

$$K_2 = 01; K_3 = 10.$$

3.7.

3.7 –

	00	01	10
	a <sub>1</sub>	a <sub>2</sub>	a <sub>3</sub>
1	00	01	10
2	01	10	01

2,

$$= \frac{2}{6} = 0.33.$$

4

4.1

Angular,

[16].

validate,

```

    validate(tableConfig: ITableConfig, tableData: ITableRow[]):
    ValidationError[]

```

4.1 –

```

private getMapBySrcStateId(tableData: ITableRow[]): Map<number,
ITableRow[]> {
    const map = new Map<number, ITableRow[]>();
    tableData.forEach((tableRow) => {
        const srcStateId = tableRow.srcStateId as number;
        if (!map.has(srcStateId)) map.set(srcStateId, []);
        const srcStateTableRows = map.get(srcStateId) as ITableRow[];
        srcStateTableRows.push(tableRow);
    });
    return map;
}

```

4.2 –



```

    ,
    .
    :
-   ,
-   ,
    ,
    a_i a_j;
-   .
    .
    ,
    :
-   ,
-   ;
-   ;
-   ,
.

```

**getIndexCodeValueMap:**

```

private getIndexCodeValueMap(uniqueIndexes: number[]): Map<number,
number> {
    const map = new Map();
    uniqueIndexes.forEach((signalIndex, position) => {
        map.set(signalIndex, 2 ** (uniqueIndexes.length - 1 - position));
    });
    return map;
}

```

4.3 –

```

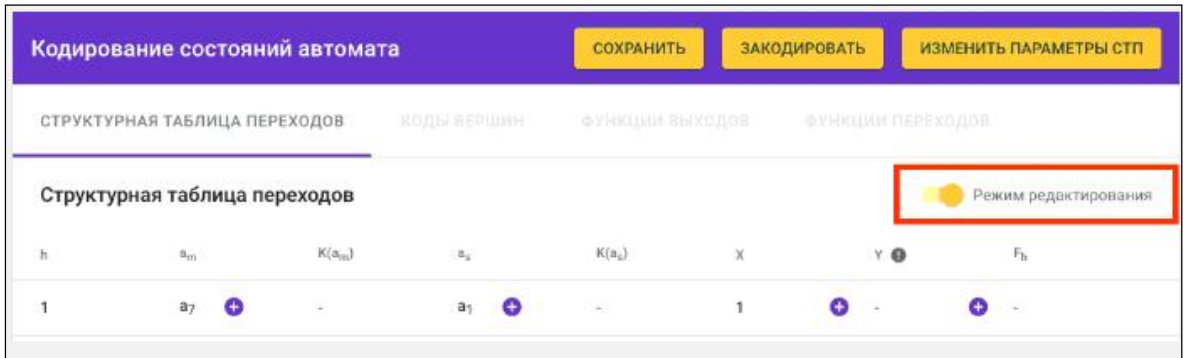
    ,
    .
    ,
    ,
    .
    ,
    ,
    .

```



4.2

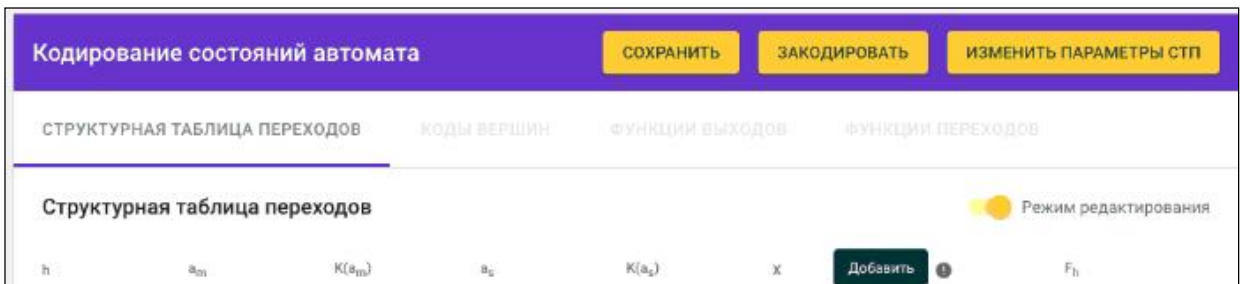
$\{a_1, \dots, a_n\}$ ,  $\{x_1, \dots, x_m\}$ ,  $\{y_1, \dots, y_l\}$ ,  $\{f_1, \dots, f_k\}$ ,  $(D, T, T^*)$ ,  $(\{a_1, \dots, a_n\}, \{x_1, \dots, x_m\}, \{y_1, \dots, y_l\}, \{f_1, \dots, f_k\})$ ,  $Y(\{a_1, \dots, a_n\}, \{x_1, \dots, x_m\})$ .  
 « $\dots$ »,  $\dots$ .  
 $\dots$ ,  $\dots$ ,  $\dots$ .  
 (  $\dots$  ).  
 ( 4.1).



4.1 –

«+»

$\dots$ ,  $\dots$  (  $\dots$ , ' check boxes) ( 4.2).



4.2 –

« » ( . 4.3).

Кодирование состояний автомата

СОХРАНИТЬ ЗАКОДИРОВАТЬ ИЗМЕНИТЬ ПАРАМЕТРЫ СТП

СТРУКТУРНАЯ ТАБЛИЦА ПЕРЕХОДОВ КОДЫ ВЕРШИН ФУНКЦИИ ВЫХОДОВ ФУНКЦИИ ПЕРЕХОДОВ

Структурная таблица переходов Режим редактирования

h	$a_m$	$K(a_m)$	$a_s$	$K(a_s)$	x	Y	$F_h$
1	$a_7$ +	-	$a_1$ +	-	1	+ -	+ -

4.3 –

а<sub>s</sub>, Y а<sub>s</sub>.

(D<sub>1</sub>)

« » ( . 4.4).

Кодирование состояний автомата

СОХРАНИТЬ ЗАКОДИРОВАТЬ ИЗМЕНИТЬ ПАРАМЕТРЫ СТП

СТРУКТУРНАЯ ТАБЛИЦА ПЕРЕХОДОВ КОДЫ ВЕРШИН ФУНКЦИИ ВЫХОДОВ ФУНКЦИИ ПЕРЕХОДОВ

Структурная таблица переходов Режим редактирования

h	$a_m$	$K(a_m)$	$a_s$	$K(a_s)$	x	Y	$F_h$
1	$a_7$ +	-	$a_1$ +	-	1	+ -	+ -

4.4 –

D- ( , ,  
 ). T- \* -  
 , ,  
 .  
 « »,  
 « » « » ( . 4.5).

Кодирование состояний автомата

СОХРАНИТЬ ЗАКОДИРОВАТЬ ИЗМЕНИТЬ ПАРАМЕТРЫ СТП

СТРУКТУРНАЯ ТАБЛИЦА ПЕРЕХОДОВ КОДЫ ВЕРШИН ФУНКЦИИ ВЫХОДОВ ФУНКЦИИ ПЕРЕХОДОВ

Структурная таблица переходов Режим редактирования

h	$a_{h1}$	$K(a_{h1})$	$a_{h2}$	$K(a_{h2})$	X	Y	$F_h$
1	$a_7$ +	1000000	$a_1$ +	0000001	1	+ -	+ 0000001
2	$a_6$ +	0100000	$a_1$ +	0000001	$\bar{x}_3$	+ -	+ 0000001

4.5 –

« »  
 .  
 « » « »

- ( . 4.6).

Кодирование состояний автомата

СОХРАНИТЬ ЗАКОДИРОВАТЬ ИЗМЕНИТЬ ПАРАМЕТРЫ СТП

СТРУКТУРНАЯ ТАБЛИЦА ПЕРЕХОДОВ КОДЫ ВЕРШИН ФУНКЦИИ ВЫХОДОВ ФУНКЦИИ ПЕРЕХОДОВ

Функции выходов Выбрать представление

Y:

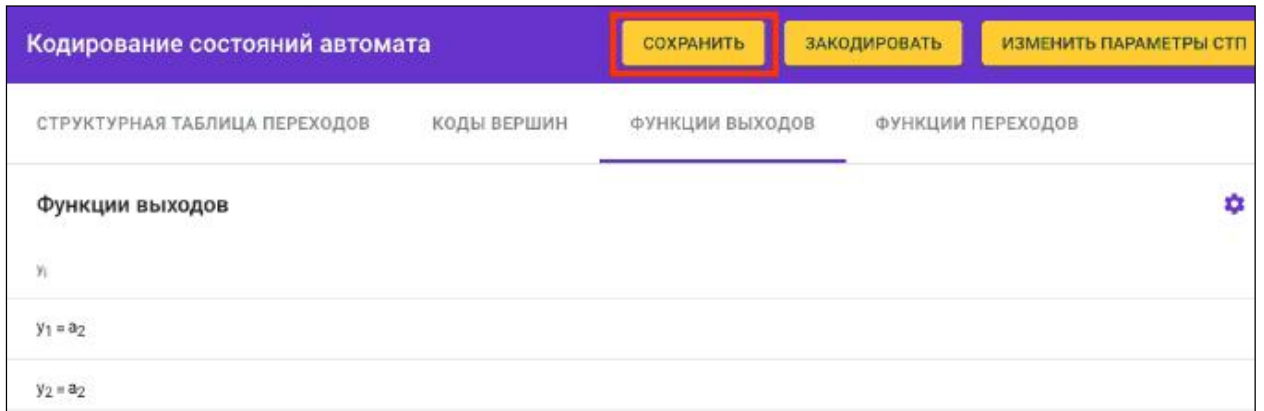
Y<sub>1</sub> = 0<sub>2</sub>

4.6 –

« ».

« »,

« » ( .4.7).



4.7 –

docx,

:

4.3

2 3:

2

2

5

3,

( .4.8).

Кодирование состояний автомата

coding-states Редактировать Файл Вид Окно Помощь

**Кодирование состояний автомата** Автомат Мили

СОХРАНИТЬ ЗАКОДИРОВАТЬ ИЗМЕНИТЬ ПАРАМЕТРЫ СТП

СТРУКТУРНАЯ ТАБЛИЦА ПЕРЕХОДОВ КОДЫ ВЕРШИН ФУНКЦИИ ВЫХОДОВ ФУНКЦИИ ПЕРЕХОДА

Структурная таблица переходов Режим редактирования

h	$a_{11}$	$K(a_{11})$	$a_3$	$K(a_3)$	X	Y	$F_{11}$
1	$a_1$	-	$a_2$	-	1	-	-
2	$a_2$	-	$a_1$	-	$\bar{x}_1x_2$	-	-
3	$a_2$	-	$a_3$	-	$x_1$	-	-
4	$a_2$	-	$a_3$	-	$\bar{x}_1\bar{x}_2$	-	-
5	$a_3$	-	$a_1$	-	$x_2$	-	-

1 - 5 из 5

4.8 –

( . 4.9).

( . 4.9).

( . 4.10).





Кодирование состояний автомата  
coding-states Редактировать Файл Вид Окно Помощь

Кодирование состояний автомата  
Автомат Милли / Частотный алгоритм кодирования

СОХРАНИТЬ ЗАКОДИРОВАТЬ ИЗМЕНИТЬ ПАРАМЕТРЫ СТП

СТРУКТУРНАЯ ТАБЛИЦА ПЕРЕХОДОВ КОДЫ ВЕРШИН ФУНКЦИИ ВЫХОДОВ ФУНКЦИИ ПЕР

Структурная таблица переходов Режим редактирования

h	$a_m$	$K(a_m)$	$a_s$	$K(a_s)$	x	y	$F_h$
1	$a_1$	00	$a_2$	01	1	-	01
2	$a_2$	01	$a_1$	00	$x_1x_2$	-	00
3	$a_2$	01	$a_3$	10	$\bar{x}_1$	-	10
4	$a_3$	10	$a_1$	00	1	-	00

1 - 4 из 4

Проверьте условия переходов из состояния  $a_2$

4.10 –

( )

Кодирование состояний автомата  
coding-states Редактировать Файл Вид Окно Помощь

Кодирование состояний автомата  
Автомат Мура / Частотный алгоритм кодирования

СОХРАНИТЬ ЗАКОДИРОВАТЬ ИЗМЕНИТЬ ПАРАМЕТРЫ СТП

СТРУКТУРНАЯ ТАБЛИЦА ПЕРЕХОДОВ КОДЫ ВЕРШИН ФУНКЦИИ ВЫХОДОВ ФУНКЦИИ ПЕР

Структурная таблица переходов Режим редактирования

h	$a_m$	$K(a_m)$	$a_s$	$K(a_s)$	x	y	$F_h$
1	$a_1$	00	$a_2$	01	1	$Y_2Y_3$	01
2	$a_2$	01	$a_1$	00	$x_1x_2$	-	00
3	$a_2$	01	$a_1$	00	$\bar{x}_1$	$Y_2Y_3$	00
4	$a_2$	01	$a_3$	10	$\bar{x}_2$	$Y_1Y_4$	10
5	$a_3$	10	$a_1$	00	1	-	00

1 - 5 из 5

В графе автомата Мура не должно быть кратных дуг, т.е. должен быть единственный переход  $a_i \rightarrow a_j$ . Проверьте переходы из состояния  $a_2$

4.11 –

( )

(t) (t+1)

(t) (t+1).

- ;  
 - , ;  
 - ;  
 - , ;  
 - ,  $a_i$   
 -  $(n+1), n -$  ;  
 - ,  $a_i$   
 -  $(m+1), m -$  ;  
 -  
 :  
 ,  
 ,  
 ,  
 .  
 :  
 ,  
 ,  
 .  
 ,

Electron Angular 8  
 TypeScript. Electron  
 3 – Windows  
 (Windows 7, 8, 10), macOS (El Capitan, Sierra, High Sierra) Ubuntu.

(t) (t+1) (t) (t+1).

[13,17],

2020

1. [ ] :  
/ . . . . – .: . . . ., 1987. – 255 .
2. ( [ ] / . . . . – 2- . . . . – .: . . . ., 1979. – 232 .
3. [ ] : . . . . / . . . . – : . . . ., 2013. – 84 .
4. [ ] / . . . . , . . . . – Morgan Kaufman, 2015. – 1628 . – ISBN 978-0-9954839-1-0.
5. [ ] : / . . . . , . . . . , . . . . – .: . . . ., 2007 ( 2009). – 744 .
6. [ ] : / . . . . – .: . . . ., 2016. – 200 . – ISBN 978-966-657-642-5.
7. [ ] : . . . . / . . . . – 3- . . . . – .: - . . . ., 2010. – 800 .
8. Volnei, A. P. Finite state machines in hardware : theory and design (with VHDL and SystemVerilog) [ ] / A. P. Volnei. – The MIT Press Cambridge, 2013. – 349 . – ISBN 978-0-262-01966-8.
9. [ ] / . . . . , . . . . , . . . . , . . . . . // . . . . . – 2018. – 3. – . 75-81.

