

Синтез цифровых автоматов

План лекции

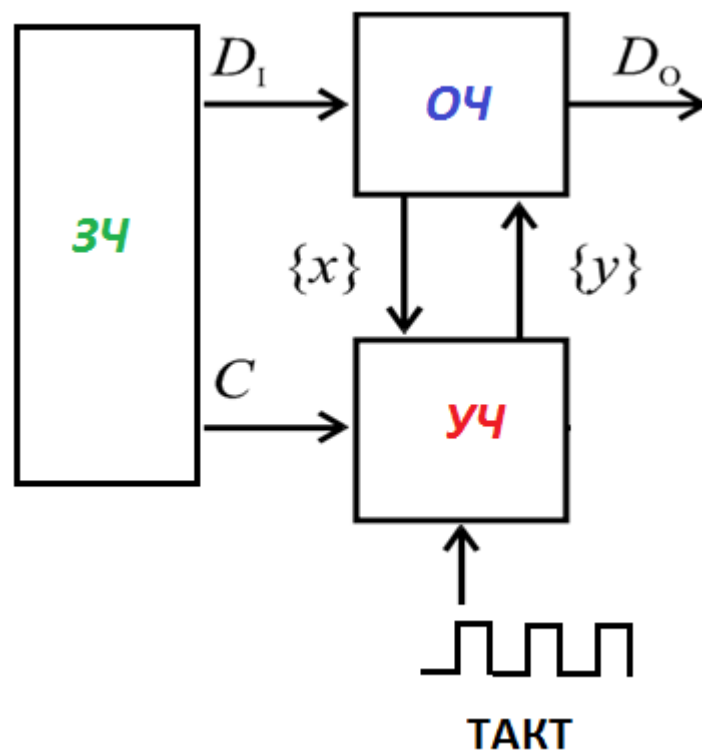
- Обобщенная структура цифровых устройств
- Классификация управляющих автоматов
- Управляющий автомат Мили.
- Управляющий автомат Мура.
- Микропрограммный автомат

Структура цифрового устройства

ЗЧ-Запоминающая часть

ОЧ-Операционная часть

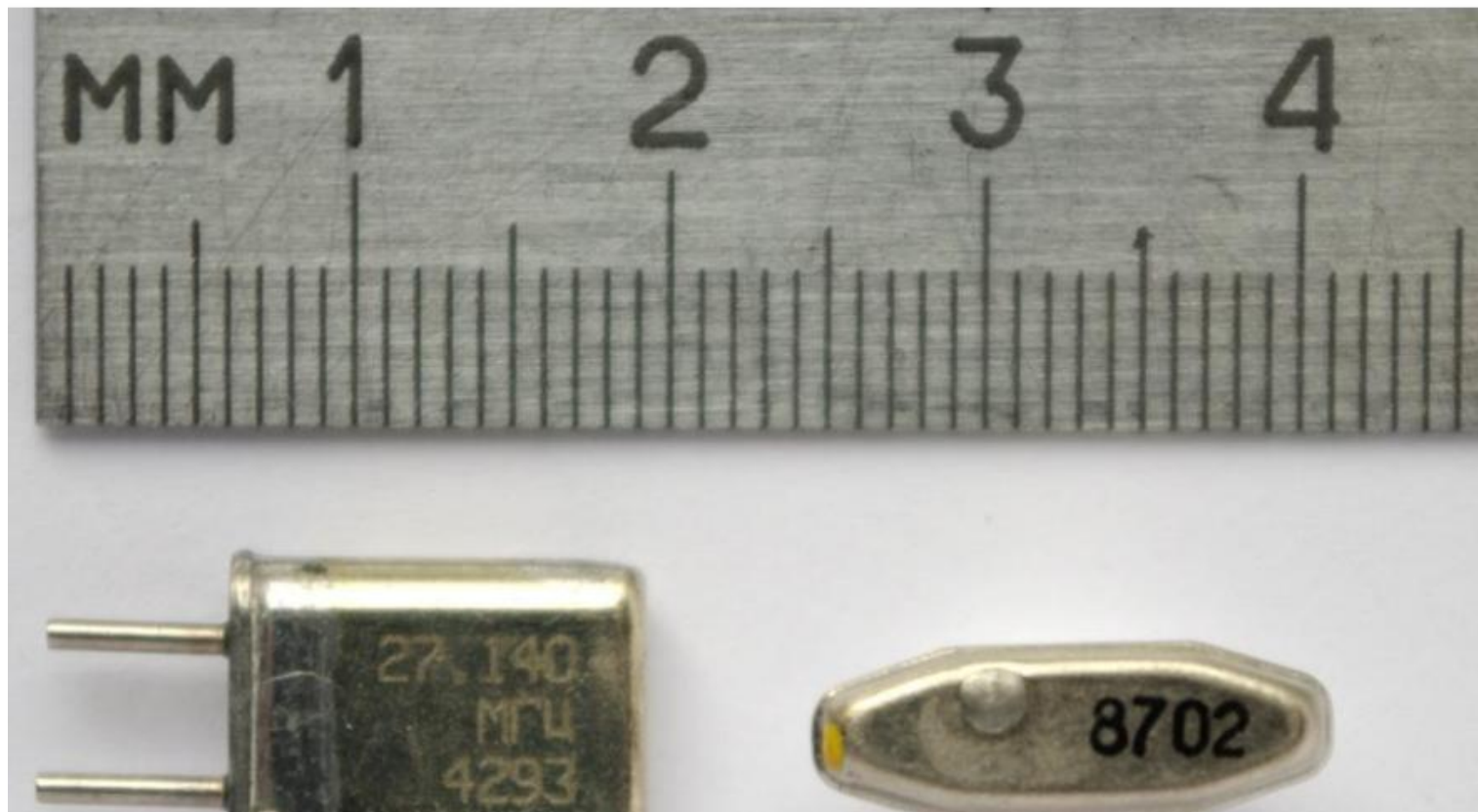
**УЧ – управляющая часть
(управляющий автомат)**



- D_1 – входные данные
- D_0 – выходные данные
- X – сигналы условий (признаков)
- Y – управляющие сигналы
- C – команды

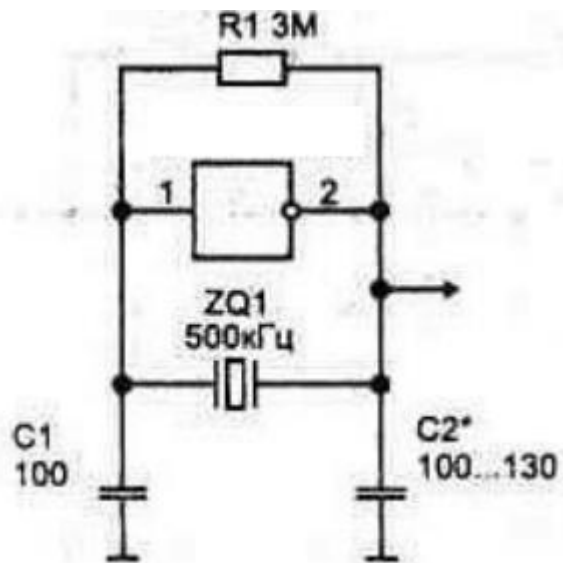
Операционная часть выполняет обработку цифровых данных под управлением управляющего автомата.

Такт и Кварцевый резонатор

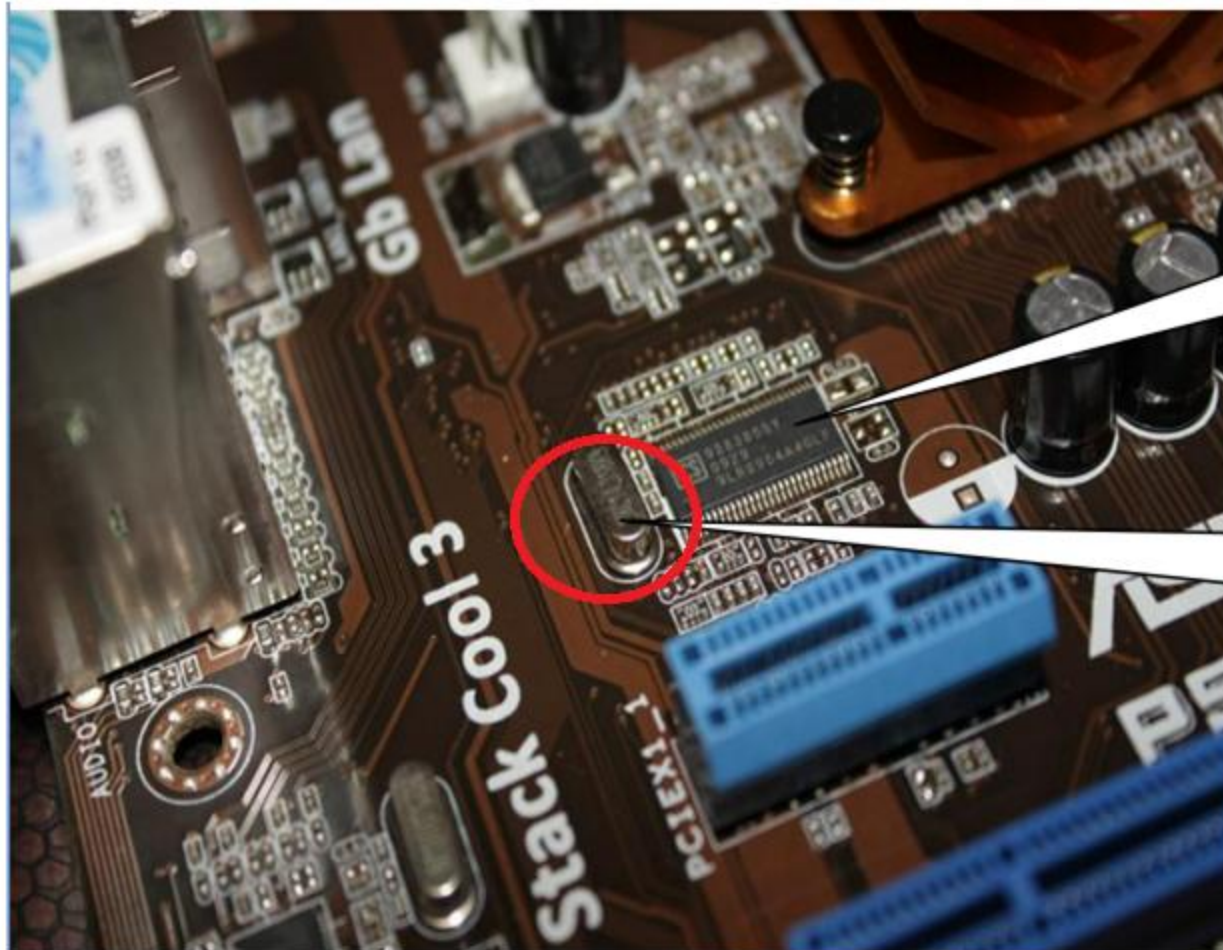


- ✓ Работа любого цифрового вычислительного устройства синхронизируется специальными тактовыми импульсами.
- ✓ Кварцевый резонатор позволяет сделать частоту этих импульсов стабильной, не зависящей от внешних факторов

Схема включения кварцевого генератора



На материнской плате



Микросхема генерации

Кварцевый резонатор

Типы управляющих автоматов

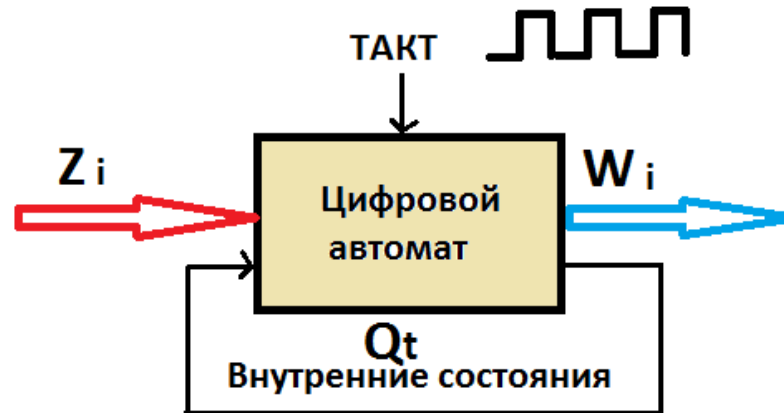
- Управляющий цифровой автомат
 - Преобразователь информации, способный принимать различные состояния, переходить под воздействием входных сигналов из одного состояния в другое по шагам заданного алгоритма и выдавать в каждом состоянии выходные управляющие сигналы.
- Дискретный управляющий автомат с жёсткой (фиксированной) логикой
 - Реализуется аппаратно.
- Микропрограммный автомат
 - Реализуется с помощью хранимой в памяти автомата микропрограммы

Дискретный управляющий автомат с
жёсткой (фиксированной) логикой

Дискретный автомат на жесткой логике

- **Абстрактный автомат**
 - математическая модель не учитывающая его внутреннюю структуру и описывает только его поведение
- **Структурный автомат**
 - реализация цифрового автомата с помощью заданного набора логических элементов.

Абстрактный автомат



- *Абстрактный автомат* – задается вектором

$$S = (Q, Z, W, \delta, \lambda, q_0)$$

- $Q = \{q_1, \dots, q_m\}$ – множество внутренних состояний абстрактного автомата,
- $Z = \{z_1, \dots, z_k\}$ – множества входных слов
- $W = \{w_1, \dots, w_l\}$ – множества выходных слов
- δ – функция переходов,
- λ – функция выходов,
- q_0 – начальное состояние автомата.

Автомат называется конечным, если множества Q, Z, W конечны.

Абстрактные Мили и Мура

- По способу формирования функции выходов все множество автоматов с жесткой логикой можно подразделить на два основных класса:

- *Автомат Мили* описывается системой уравнений:

$$Q(t+1) = \delta(Q(t), Z(t));$$

$$W(t) = \lambda(Q(t), Z(t)).$$

- *Автомат Мура* описывается системой уравнений.

$$Q(t+1) = \delta(Q(t), Z(t));$$

$$W(t) = \lambda(Q(t)).$$

Способы задания абстрактного автомата

- С помощью графа
- С помощью таблиц переходов и выходов

$Q = \{q_1, q_2, q_3, q_4\}$, $Z = \{z_1, z_2, z_3, z_4\}$, $W = \{w_1, w_2, w_3, w_4, w_5, w_6\}$

Таблица переходов

δ	q_1	q_2	q_3	q_4
z_1	q_2	q_2	-	q_4
z_2	q_4	-	q_2	q_2
z_3	q_3	q_3	q_4	q_3
z_4	-	q_4	-	q_1

Таблица выходов

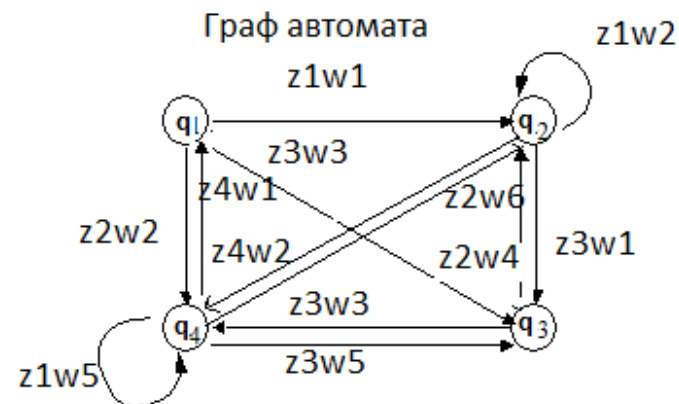
λ	q_1	q_2	q_3	q_4
z_1	w_1	w_2	-	w_5
z_2	w_2	-	w_4	w_6
z_3	w_3	w_1	w_3	w_5
z_4	-	w_2	-	w_1

$$Q(t+1) = \delta(Q(t), Z(t))$$

$$W(t) = \lambda(Q(t), Z(t))$$

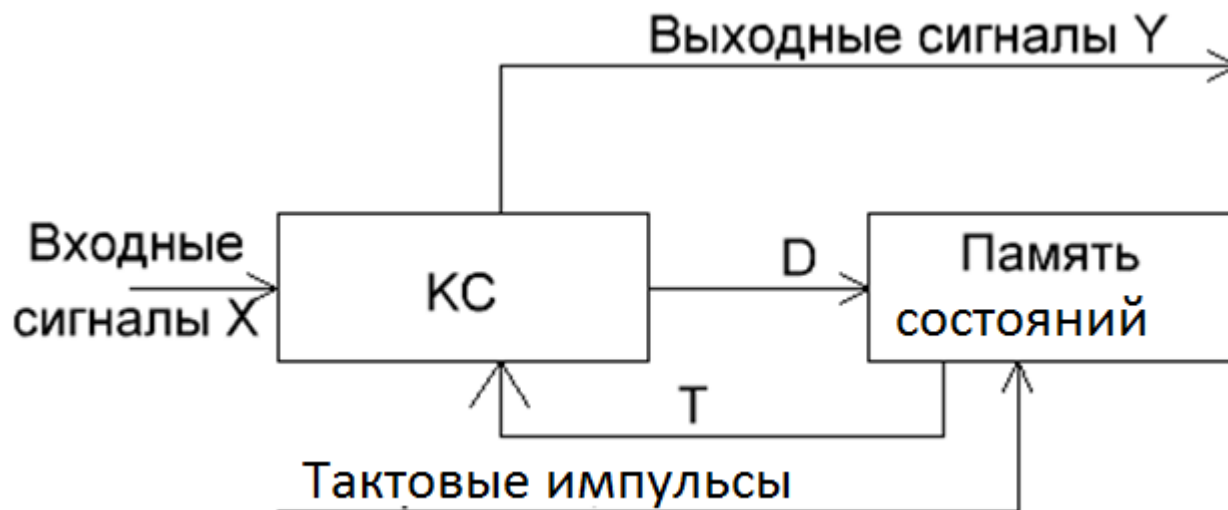
Совмещенная таблица

δ	q_1	q_2	q_3	q_4
z_1	$q_2 w_1$	$q_2 w_2$	-	$q_4 w_5$
z_2	$q_4 w_2$	-	$q_2 w_4$	$q_2 w_6$
z_3	$q_3 w_3$	$q_3 w_1$	$q_4 w_3$	$q_3 w_5$
z_4	-	$q_4 w_2$	-	$q_1 w_1$



Так как следующее состояние получается из текущего, то текущее состояние должно храниться в памяти состояний.

Структурный автомат МИЛИ



КС - комбинационная схема формирует выходные управляющие сигналы **Y** и следующие состояния **D**

X - множество входных сигналов

Y - множество выходных сигналов

D – номер следующего состояния (сигналы «возбуждения» памяти)

T- номер текущего состояния

$D = \delta(X, T)$ – функция переходов $Y = \lambda(X, T)$ функция выходов

Переход от абстрактного к структурному автомату

- Кодирование входных (Z), выходных сигналов (W) текущих и следующих состояний (Q)

Кодирование множеств Z , W , Q

$z \backslash x$	x_1	x_2
z_1	0	0
z_2	0	1
z_3	1	0
z_4	1	1

$w \backslash y$	y_1	y_2	y_3
w_1	0	0	0
w_2	0	0	1
w_3	0	1	0
w_4	0	1	1
w_5	1	0	0
w_6	1	0	1

$q \backslash T$	T_1	T_2
q_1	0	0
q_2	0	1
q_3	1	0
q_4	1	1

$q \backslash D$	D_1	D_2
q_1	0	0
q_2	0	1
q_3	1	0
q_4	1	1

Количество входных, выходных линий и разрядов памяти автоматов

$$L = \lceil \log_2 Z \rceil = \lceil \log_2 4 \rceil = 2$$

$$N = \lceil \log_2 Q \rceil = \lceil \log_2 4 \rceil = 2$$

$$M = \lceil \log_2 W \rceil = \lceil \log_2 6 \rceil = 3$$

Переход от абстрактного к структурному автомату

Совмещенная таблица

δ	q_1	q_2	q_3	q_4
z_1	$q_2 w_1$	$q_2 w_2$	-	$q_4 w_5$
z_2	$q_4 w_2$	-	$q_2 w_4$	$q_2 w_6$
z_3	$q_3 w_3$	$q_3 w_1$	$q_4 w_3$	$q_3 w_5$
z_4	-	$q_4 w_2$	-	$q_4 w_1$

Абстрактный автомат

Таблица переходов

$T_1 T_2$ $x_1 x_2$	00 q_1	01 q_2	10 q_3	11 q_4
00 z_1	01	01	-	11
01 z_2	11	-	01	01
10 z_3	10	10	11	10
11 z_4	-	11	-	00

$D_1 D_2$

$D_1 D_2$

Таблица выходов

$T_1 T_2$ $x_1 x_2$	00 q_1	01 q_2	10 q_3	11 q_4
00 z_1	000 w_1	001 w_2	-	100 w_5
01 z_2	001 w_2	-	011 w_4	101 w_6
10 z_3	010 w_3	000 w_1	010 w_3	100 w_5
11 z_4	-	001 w_2	-	000 w_1

$y_1 y_2 y_3$

$y_1 y_2 y_3$

Структурный автомат

$$D1 = \overline{T1} \cdot \overline{T2} \cdot \overline{x1} \cdot x2 + \overline{T1} \cdot \overline{T2} \cdot x1 \cdot \overline{x2} + \overline{T1} \cdot T2 \cdot \overline{x1} \cdot x2 + \overline{T1} \cdot T2 \cdot x1 \cdot \overline{x2} + T1 \cdot \overline{T2} \cdot \overline{x1} \cdot \overline{x2} + T1 \cdot \overline{T2} \cdot x1 \cdot x2 + T1 \cdot T2 \cdot \overline{x1} \cdot \overline{x2} + T1 \cdot T2 \cdot x1 \cdot x2$$

$$D2 = \overline{T1} \cdot \overline{T2} \cdot \overline{x1} \cdot \overline{x2} + \overline{T1} \cdot \overline{T2} \cdot \overline{x1} \cdot x2 + \dots$$

$$Y1 = T1 \cdot T2 \cdot \overline{x1} \cdot \overline{x2} + T1 \cdot T2 \cdot \overline{x1} \cdot x2 + T1 \cdot T2 \cdot x1 \cdot \overline{x2}$$

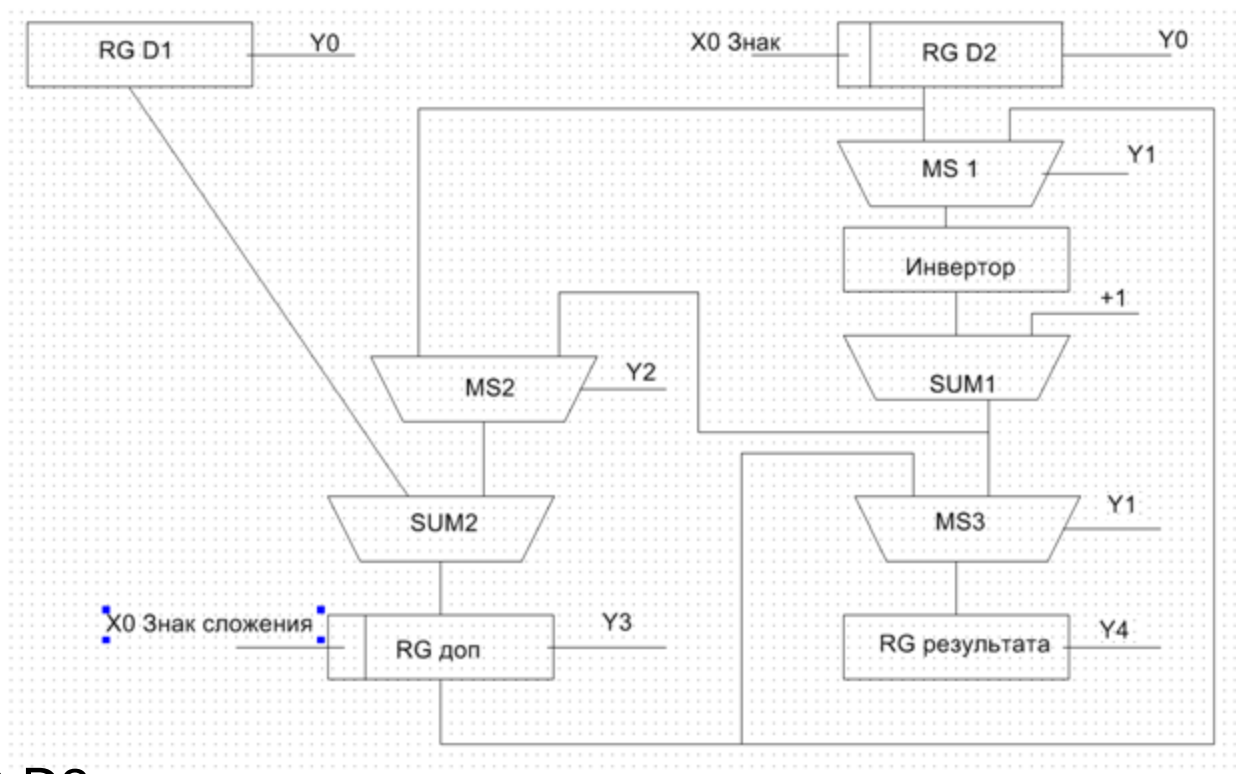
$$Y2 = \overline{T1} \cdot \overline{T2} \cdot x1 \cdot \overline{x2} + T1 \cdot \overline{T2} \cdot x1 \cdot \overline{x2} + \dots$$

Далее реализуем автомат по заданным функциям

Пример синтеза УА

- Разработать устройство суммирования двух чисел .
- Одно из слагаемых может быть положительным или отрицательным.
- Используем дополнительный код

Структурная схема операционной части



RG D1, RG D2 – регистры для хранения входных чисел

X0,X1 – знак

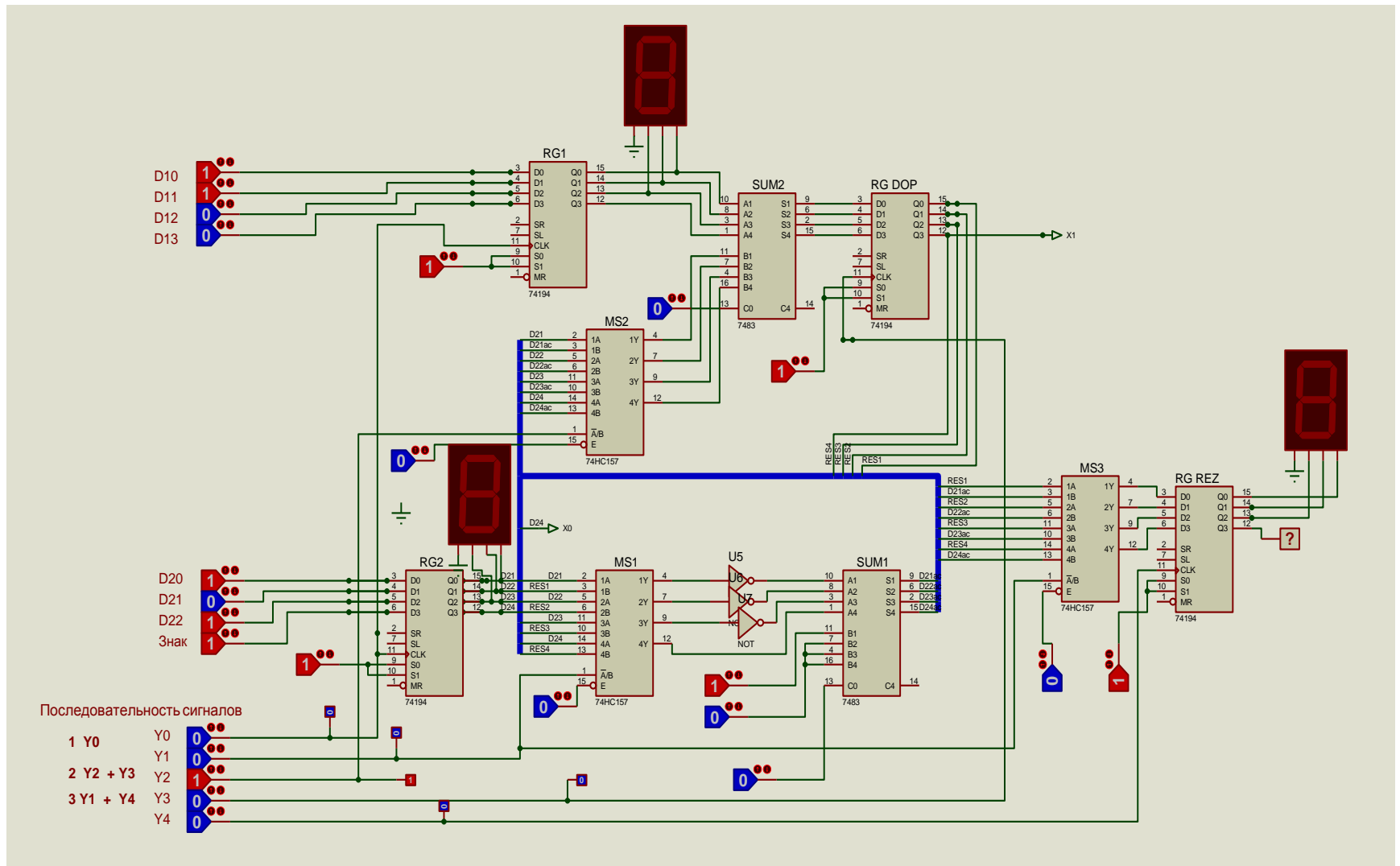
Y0 – сигнал записи в RG1, RG2

Y1 – сигнал управления мультиплексором MS1

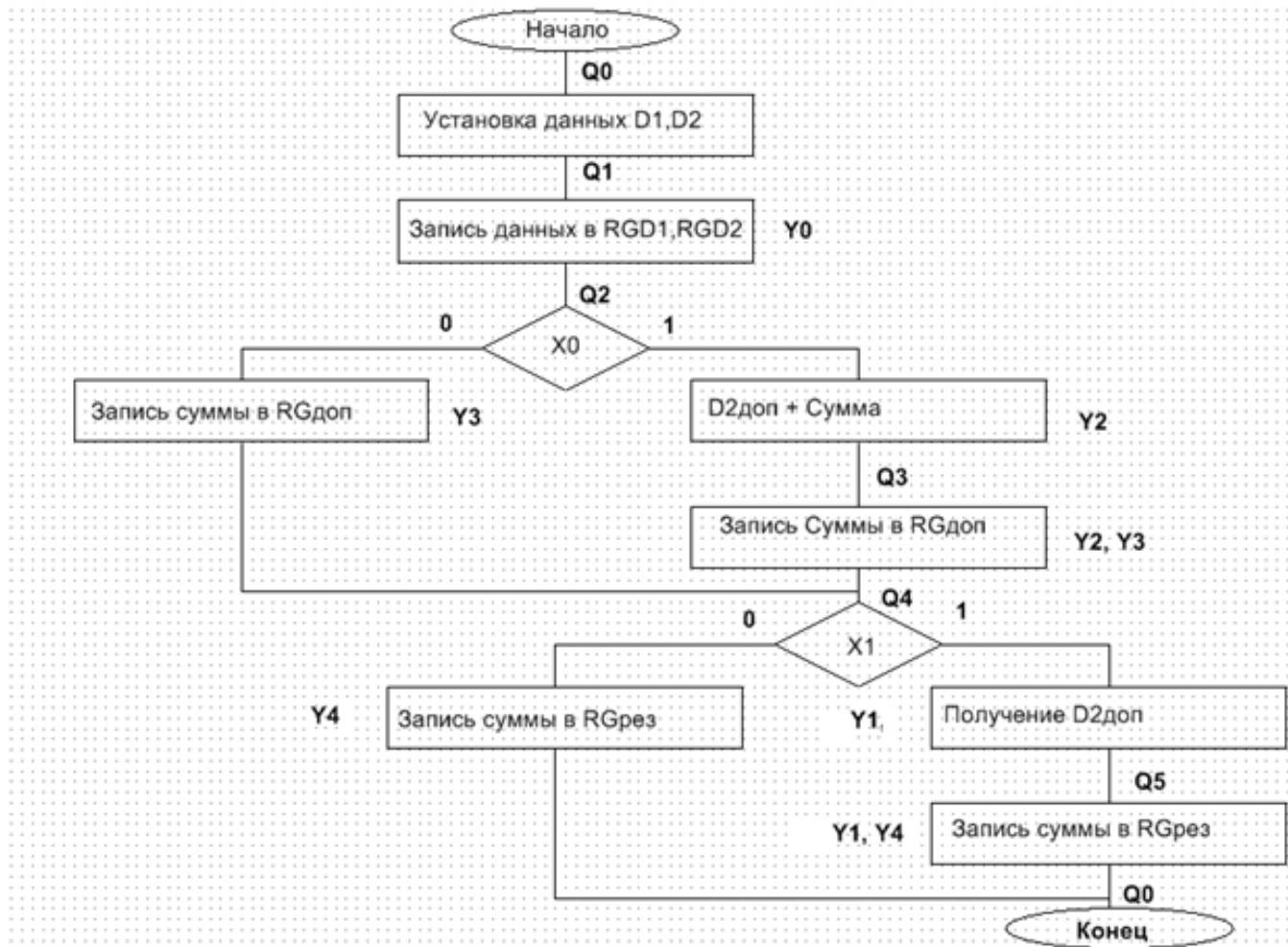
Y2 – сигнал управления мультиплексором MS2

Y3, Y4 – сигналы записи в регистры RGдоп и RGрез

Операционная часть



Разметка графа



1. Символом Q0 помечаем вход вершины, следующий за начальной и вход конечной вершины.
2. Входы всех вершин, **следующих за операторными**, помечаем символами состояний Q1-Q5
3. Вход вершины отмечаются одним символом .

Кодирование состояний

- Всего шесть состояний для их кодирования надо три элемента памяти (триггера)

Номер состояния	Код состояния		
	T1	T2	T3
Q0	0	0	0
Q1	1	0	0
Q2	0	1	0
Q3	1	1	0
Q4	0	0	1
Q5	1	0	1

Объединенная таблица переходов и выходов



Исх. сост.	Код <u>исходного</u> состояния			Условия перехода		След. Сост.	Код <u>следующего</u> состояния			<u>Выходные сигналы</u>					
	T1	T2	T3	X0	X1		D1	D2	D3	Y0	Y1	Y2	Y3	Y4	
Q0	0	0	0	*	*	Q1	1	0	0	0	0	0	0	0	
Q1	1	0	0	*	*	Q2	0	1	0	1	0	0	0	0	
Q2	0	1	0	0	*	Q4	0	0	1	0	0	0	1	0	
Q2	0	1	0	1	*	Q3	1	1	0	0	0	1	0	0	
Q3	1	1	0	1	*	Q4	0	0	1	0	0	1	1	0	
Q4	0	0	1	*	0	Q0	0	0	0	0	0	0	0	1	
Q4	0	0	1	*	1	Q5	1	0	1	0	1	0	0	0	
Q5	1	0	1	*	1	Q0	0	0	0	0	1	0	0	1	

Состояние Q0 необходимо для начальной установки данных D1, D2.

В состоянии Q1 данные записываются во входные регистры.

Логические выражения

$$Y_0 = T_1 \cdot \overline{T_2} \cdot \overline{T_3}$$

$$Y_1 = \overline{T_1} \cdot \overline{T_2} \cdot T_3 \cdot x_1 + T_1 \cdot \overline{T_2} \cdot T_3 \cdot x_1$$

$$Y_2 = \overline{T_1} \cdot T_2 \cdot \overline{T_3} \cdot x_0 + T_1 \cdot T_2 \cdot \overline{T_3} \cdot x_0 = T_2 \cdot \overline{T_3} \cdot x_0 (\overline{T_1} + T_1) = T_2 \cdot \overline{T_3} \cdot x_0$$

$$Y_3 = \overline{T_1} \cdot T_2 \cdot \overline{T_3} \cdot \overline{x_0} + T_1 \cdot T_2 \cdot \overline{T_3} \cdot X_0$$

$$Y_4 = \overline{T_1} \cdot \overline{T_2} \cdot T_3 \cdot \overline{x_1} + T_1 \cdot \overline{T_2} \cdot T_3 \cdot \overline{x_1}$$

$$D_1 = \overline{T_1} \cdot \overline{T_2} \cdot \overline{T_3} + \overline{T_1} \cdot T_2 \cdot \overline{T_3} \cdot x_0 + \overline{T_1} \cdot \overline{T_2} \cdot T_3 \cdot x_1$$

$$D_2 = T_1 \cdot \overline{T_2} \cdot \overline{T_3} + \overline{T_1} \cdot T_2 \cdot \overline{T_3} \cdot x_0$$

$$D_3 = \overline{T_1} \cdot T_2 \cdot \overline{T_3} \cdot \overline{x_0} + T_1 \cdot T_2 \cdot \overline{T_3} \cdot x_0 + \overline{T_1} \cdot \overline{T_2} \cdot T_3 \cdot x_1$$

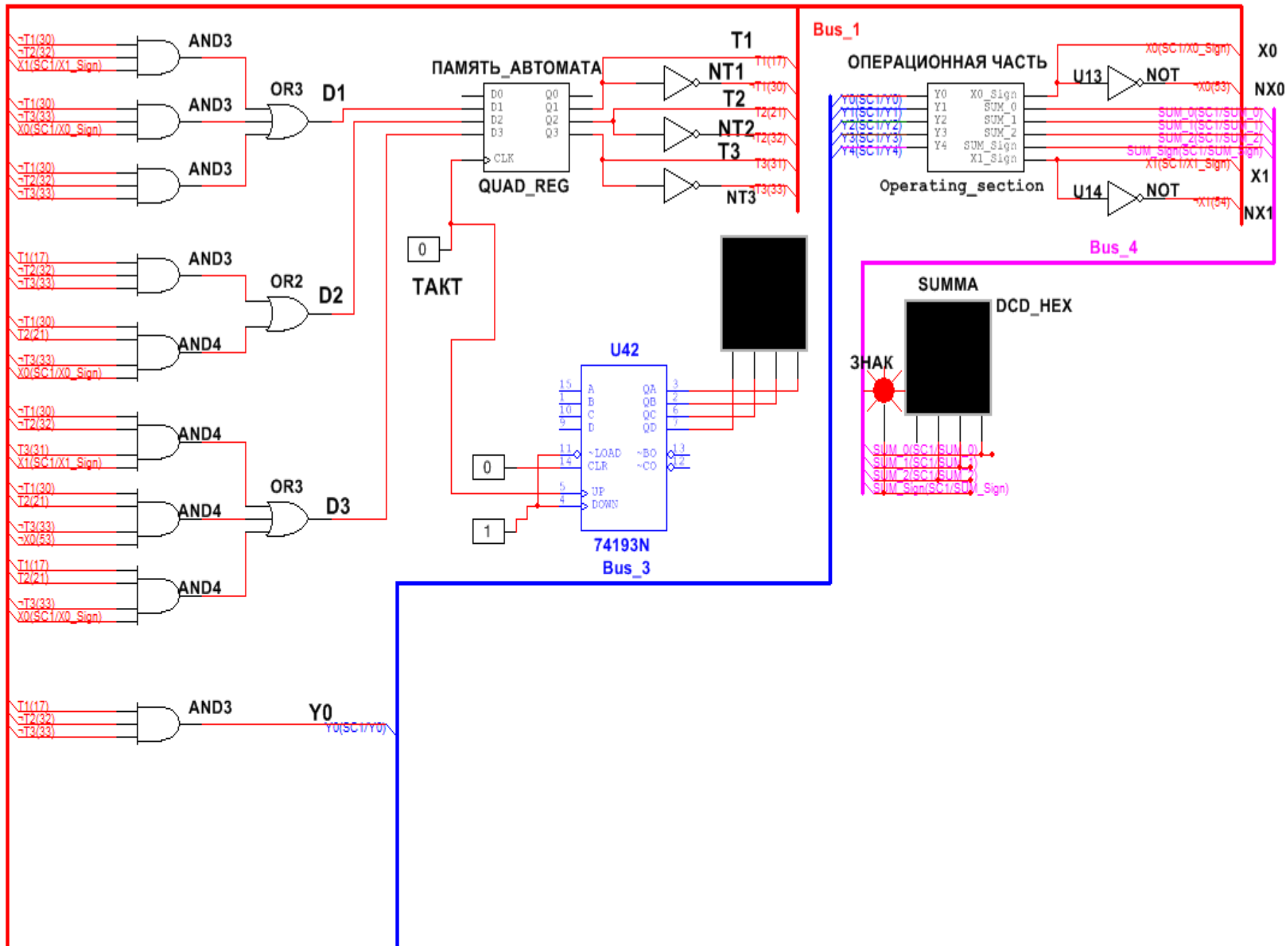
Логические выражения

$$D1 = \overline{T1} \cdot \overline{T2} \cdot \overline{T3} \cdot \overline{x0} \cdot \overline{x1} + \overline{T1} \cdot \overline{T2} \cdot \overline{T3} \cdot x0 \cdot \overline{x1} + \overline{T1} \cdot \overline{T2} \cdot \overline{T3} \cdot \overline{x0} \cdot x1 + \overline{T1} \cdot \overline{T2} \cdot \overline{T3} \cdot x0 \cdot x1 + \\ \overline{T1} \cdot T2 \cdot \overline{T3} \cdot x0 \cdot \overline{x1} + \overline{T1} \cdot T2 \cdot \overline{T3} \cdot x0 \cdot x1 + \overline{T1} \cdot \overline{T2} \cdot T3 \cdot \overline{x0} \cdot x1 + \overline{T1} \cdot \overline{T2} \cdot T3 \cdot x0 \cdot x1$$

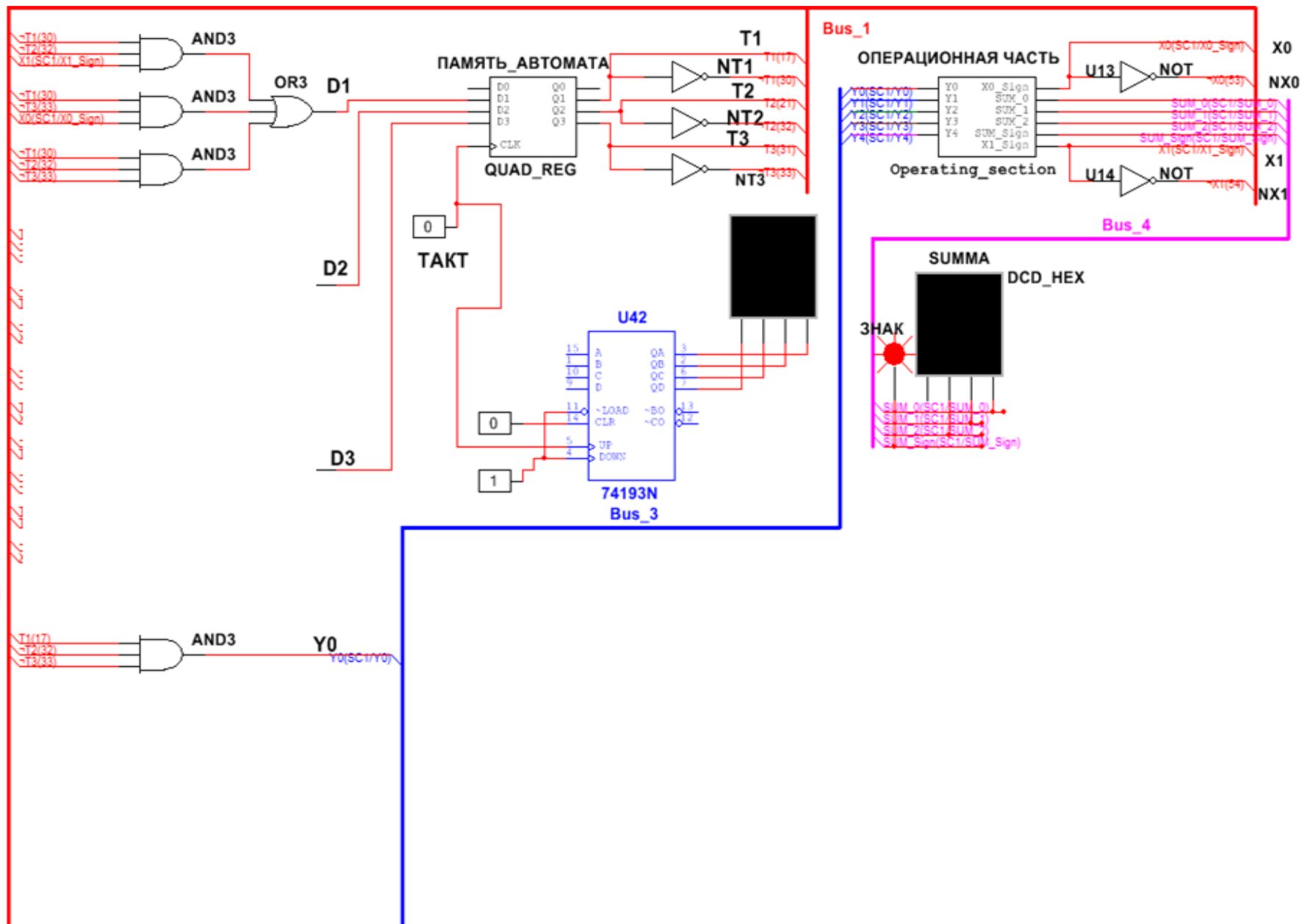
$$D2 = T1 \cdot \overline{T2} \cdot \overline{T3} \cdot \overline{x0} \cdot \overline{x1} + T1 \cdot \overline{T2} \cdot \overline{T3} \cdot \overline{x0} \cdot x1 + T1 \cdot \overline{T2} \cdot \overline{T3} \cdot x0 \cdot \overline{x1} + T1 \cdot \overline{T2} \cdot \overline{T3} \cdot x0 \cdot x1 + \\ \overline{T1} \cdot T2 \cdot \overline{T3} \cdot x0 \cdot \overline{x1} + \overline{T1} \cdot T2 \cdot \overline{T3} \cdot x0 \cdot x1$$

$$D3 = \overline{T1} \cdot T2 \cdot \overline{T3} \cdot \overline{x0} \cdot \overline{x1} + \overline{T1} \cdot T2 \cdot \overline{T3} \cdot \overline{x0} \cdot x1 + T1 \cdot T2 \cdot \overline{T3} \cdot x0 \cdot \overline{x1} + T1 \cdot T2 \cdot \overline{T3} \cdot x0 \cdot x1 + \\ \overline{T1} \cdot \overline{T2} \cdot T3 \cdot \overline{x0} \cdot x1 + \overline{T1} \cdot \overline{T2} \cdot T3 \cdot x0 \cdot x1$$

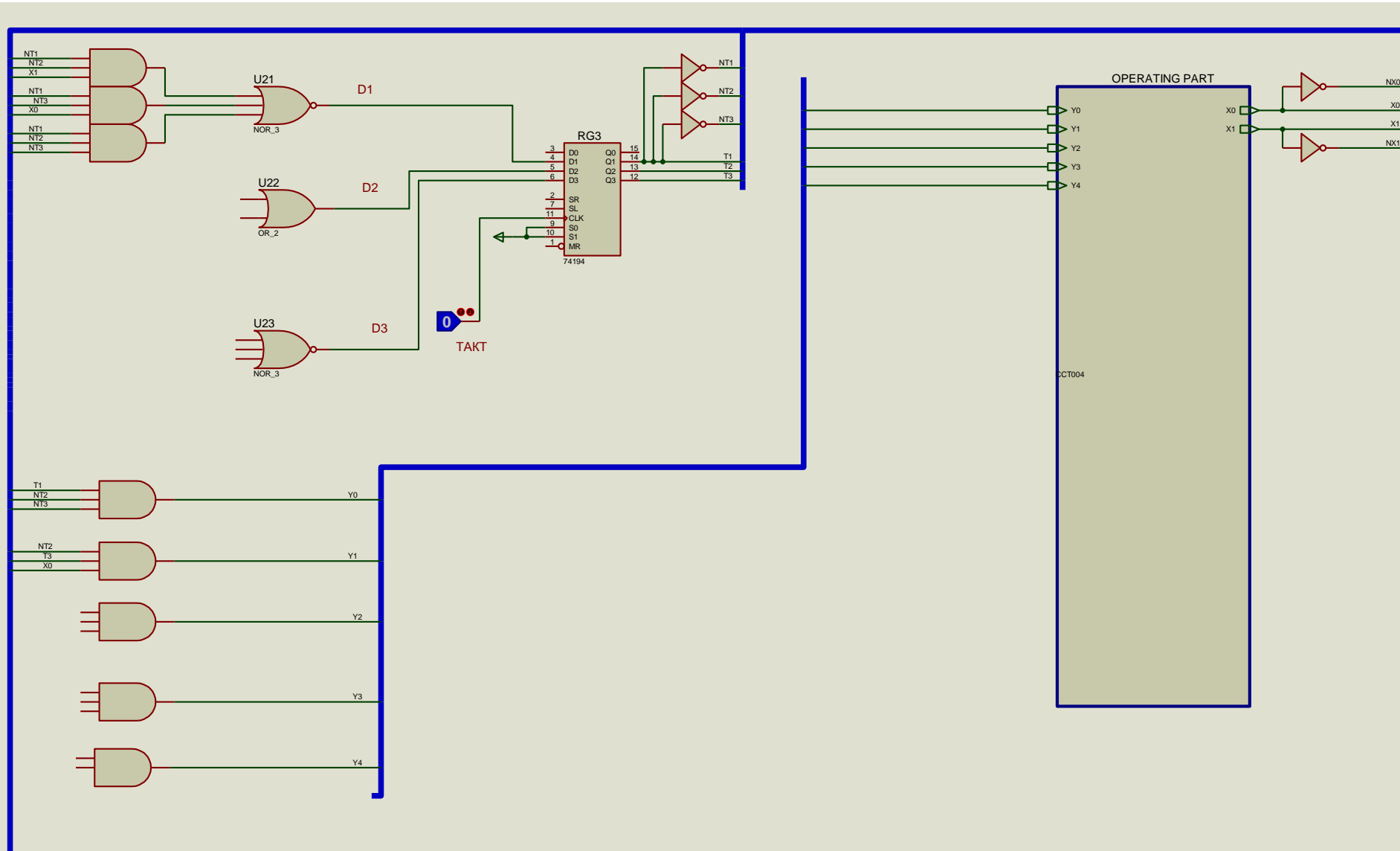
Управляющая часть



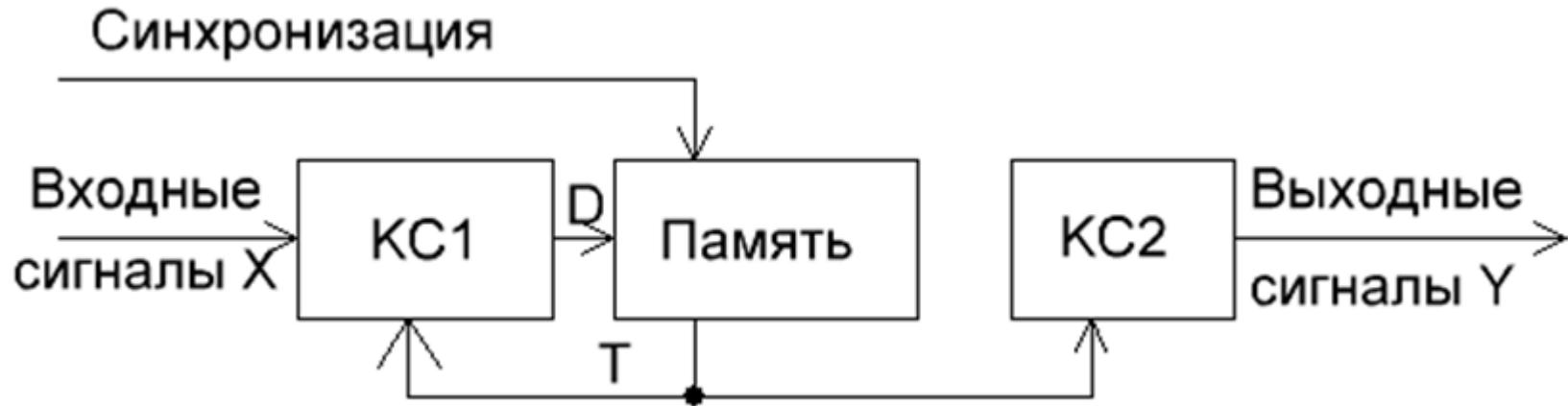
Управляющая часть



Управляющая часть



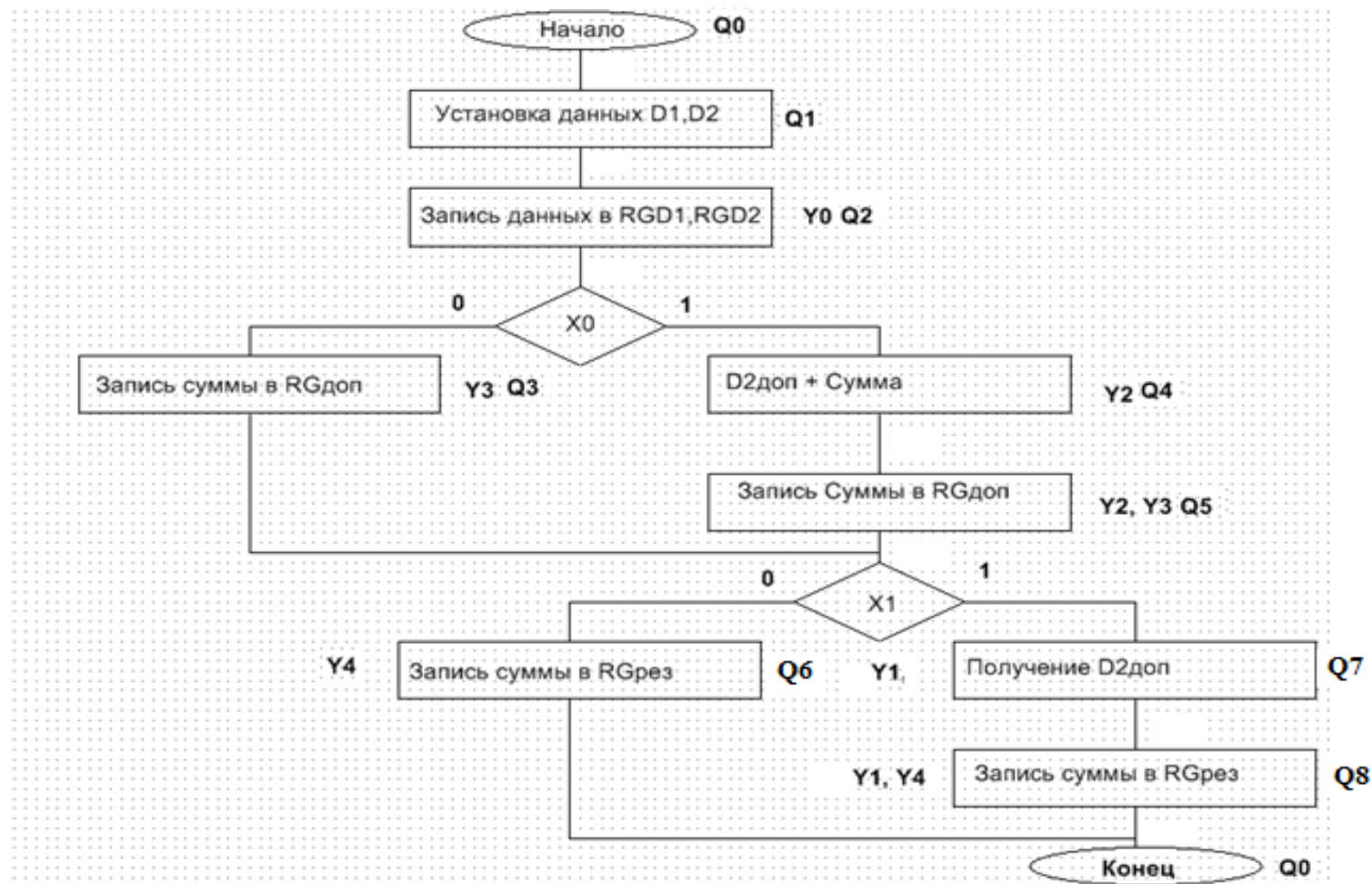
Синтез автомата Мура



- Две комбинационные схемы
- КС1 реализует функцию переходов $D = \delta(X, T)$,
- КС2 реализует функцию выходов $Y = \lambda(X, T)$.
- Для каждой комбинационной схемы строится своя таблица состояний.

У автомата Мура выход «развязан» от входа

Правила разметки ГСА для автомата Мура



1. Символом Q0 отмечается начальная и конечная вершина .
2. Символами Q1-Q5 отмечаются все **операторные вершины**.
3. Каждая операторная вершина помечается одним символом.

Кодирование состояний и таблица выходов для КС2

Всего 9 состояний – надо 4 триггера

Номер состояния	<u>Код состояния</u>				Значения выходных сигналов Y
	T1	T2	T3	T4	
					<u>Y0-Y4=0</u>
Q0	0	0	0	0	<u>Y0-Y4=0</u>
Q1	1	0	0	0	<u>Y0-Y4=0</u>
Q2	0	1	0	0	<u>Y1-Y4=0</u> <u>Y0=1</u>
Q3	1	1	0	0	<u>Y0,Y1,Y2,Y4=0</u> <u>Y3=1</u>
Q4	0	0	1	0	<u>Y0,Y1,Y3,Y4=0</u> <u>Y2=1</u>
Q5	1	0	1	0	<u>Y0,Y1,Y4=0</u> <u>Y2,Y3=1</u>
Q6	0	1	1	0	<u>Y0,Y1,Y2,Y3=0</u> <u>Y4=1</u>
Q7	1	1	1	0	<u>Y0,Y2,Y3,Y4=0</u> <u>Y1=1</u>
<u>Q8</u>	0	0	0	1	<u>Y0,Y2,Y3=0</u> <u>Y1,Y4=1</u>

$$Y0 = Q2 = \overline{T1} \cdot T2 \cdot \overline{T3} \cdot T4$$

$$Y1 = Q7 + Q8 = \overline{T1} \cdot T2 \cdot T3 \cdot \overline{T4} + T1 \cdot T2 \cdot T3 \cdot \overline{T4}$$

Кодирование состояний и таблица переходов для КС1

Исх. сост.	Код исходного состояния				Условия перехода		След. Сост.	Код следующего состояния			
	T1	T2	T3	T4	X0	X1		D1	D2	D3	D4
Q0	0	0	0	0	*	*	Q1	1	0	0	0
Q1	1	0	0	0	*	*	Q2	0	1	0	0
Q2	0	1	0	0	1	*	Q4	0	0	1	0
Q2	0	1	0	0	0	*	Q3	1	1	0	0
Q3	1	1	0	0	*	0	Q6	0	1	1	0
Q3	1	1	0	0	*	1	Q7	1	1	1	0
Q4	0	0	1	0	1	*	Q5	1	0	1	0
Q5	1	0	1	0	*	0	Q6	1	0	1	0
Q5	1	0	1	0	*	1	Q7	1	1	1	0
Q6	0	1	1	0	*	*	Q0	0	0	0	0
Q7	1	1	1	0	*	*	Q8	0	0	0	1
Q8	0	0	0	1	*	*	Q0	0	0	0	0

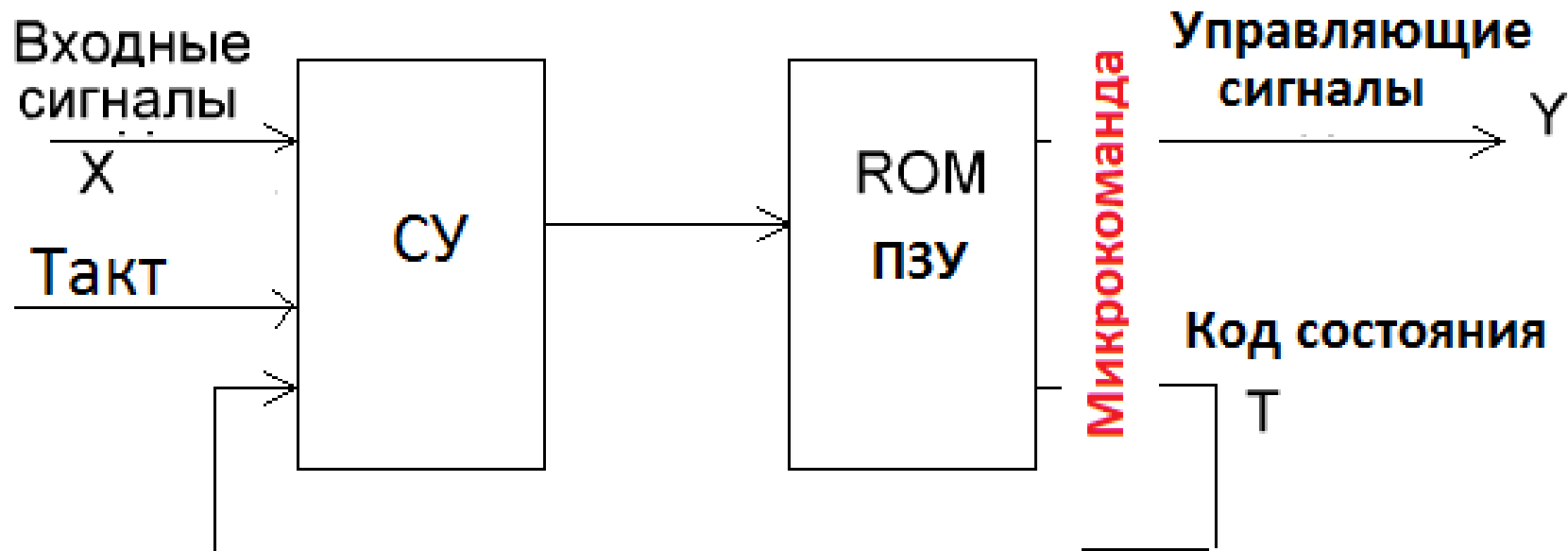
$$D4 = Q7 = T1 \cdot T2 \cdot T3 \cdot \overline{T4}$$

- Минимизируем и разрабатываем схему

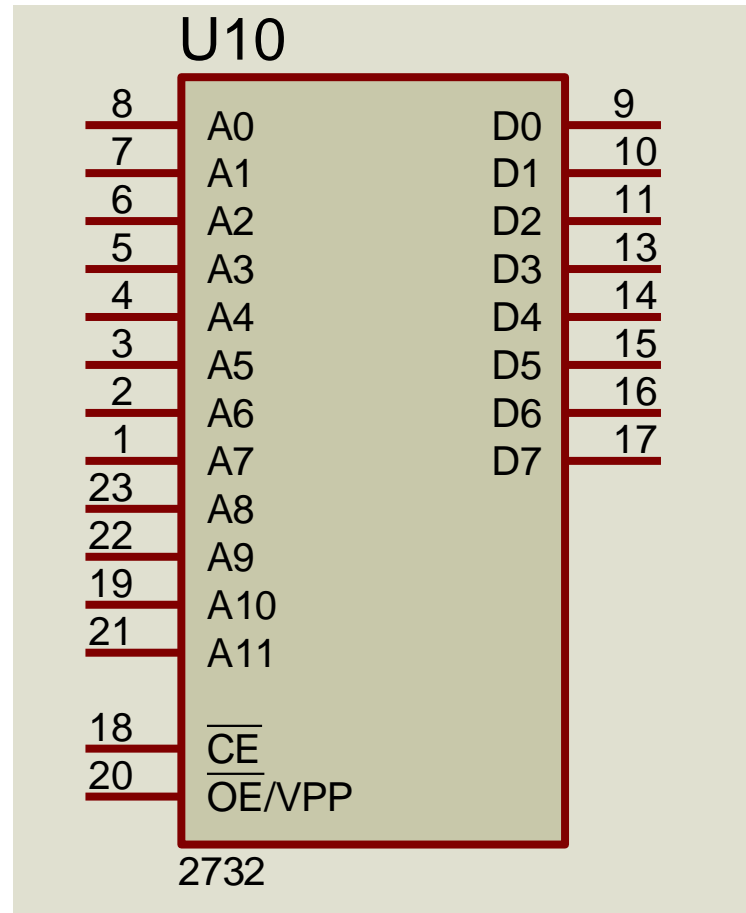
Недостатки УА на основе жесткой логики

- 1. При необходимости внесения любых, даже небольших изменений алгоритма работы схему автомата надо полностью пересинтезировать.
- 2. При большом числе входных и выходных сигналов схема автомата сильно разрастается, а синтез сильно усложняется.

Микропрограммный автомат (общая схема)



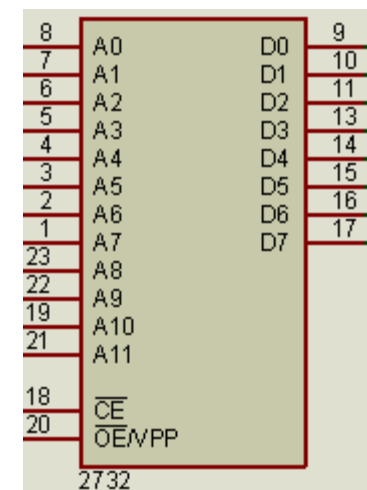
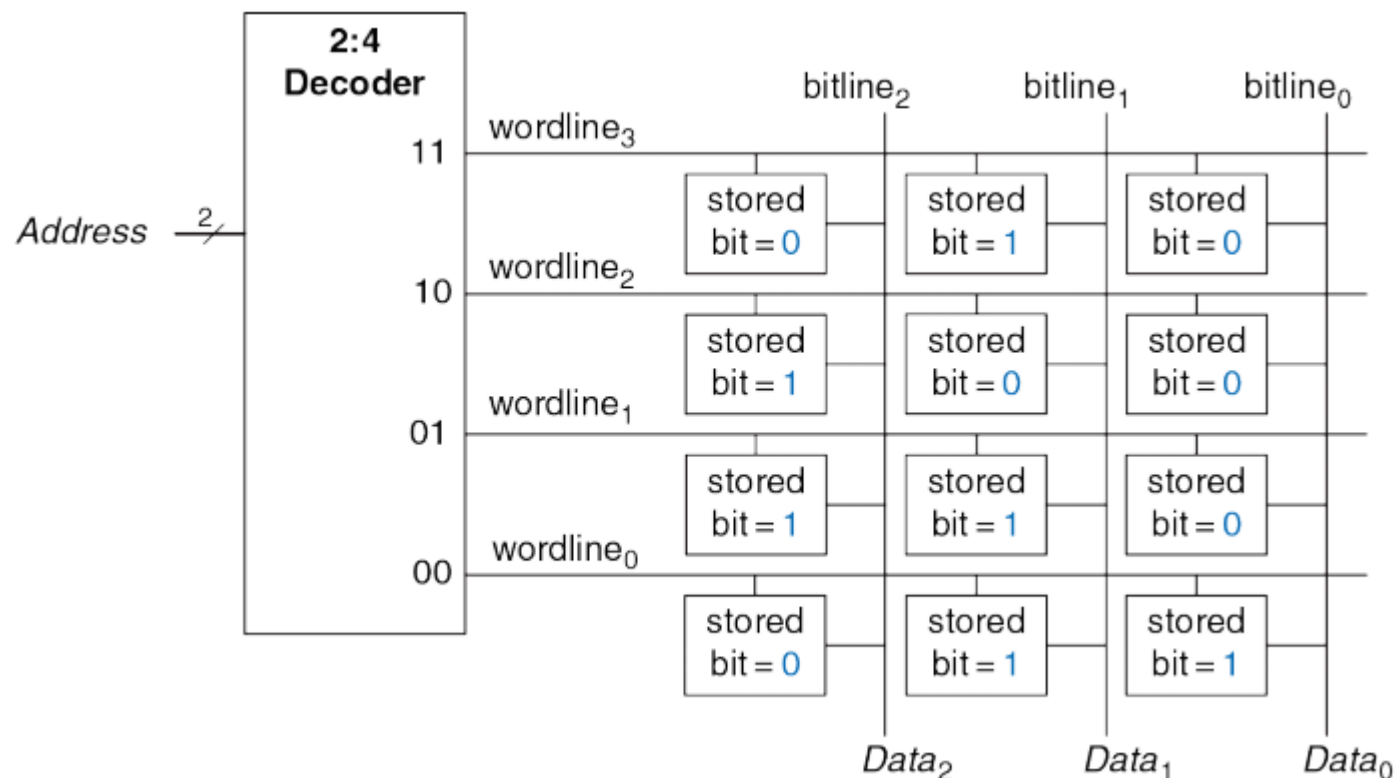
- Каждая ячейка ПЗУ хранит биты микрокоманды (МК)
- Микрокоманда содержит поля:
 - Y – набор бит управляющих сигналов для каждого состояния
 - T – код следующего состояния (адрес ячейки ПЗУ).
- Схема Управления ПЗУ – формирует новый адрес ячейки ПЗУ для перехода в следующее состояние



- При $OE = 1$ выводы находятся в третьем состоянии,
- $OE=0$ и $CE=0$ режим чтения байта по адресу A0-A11

Байт ориентированная память

- Матрица памяти (прямоугольная)



- Ячейки выбираются построчно

Микропрограммный автомат (структура)

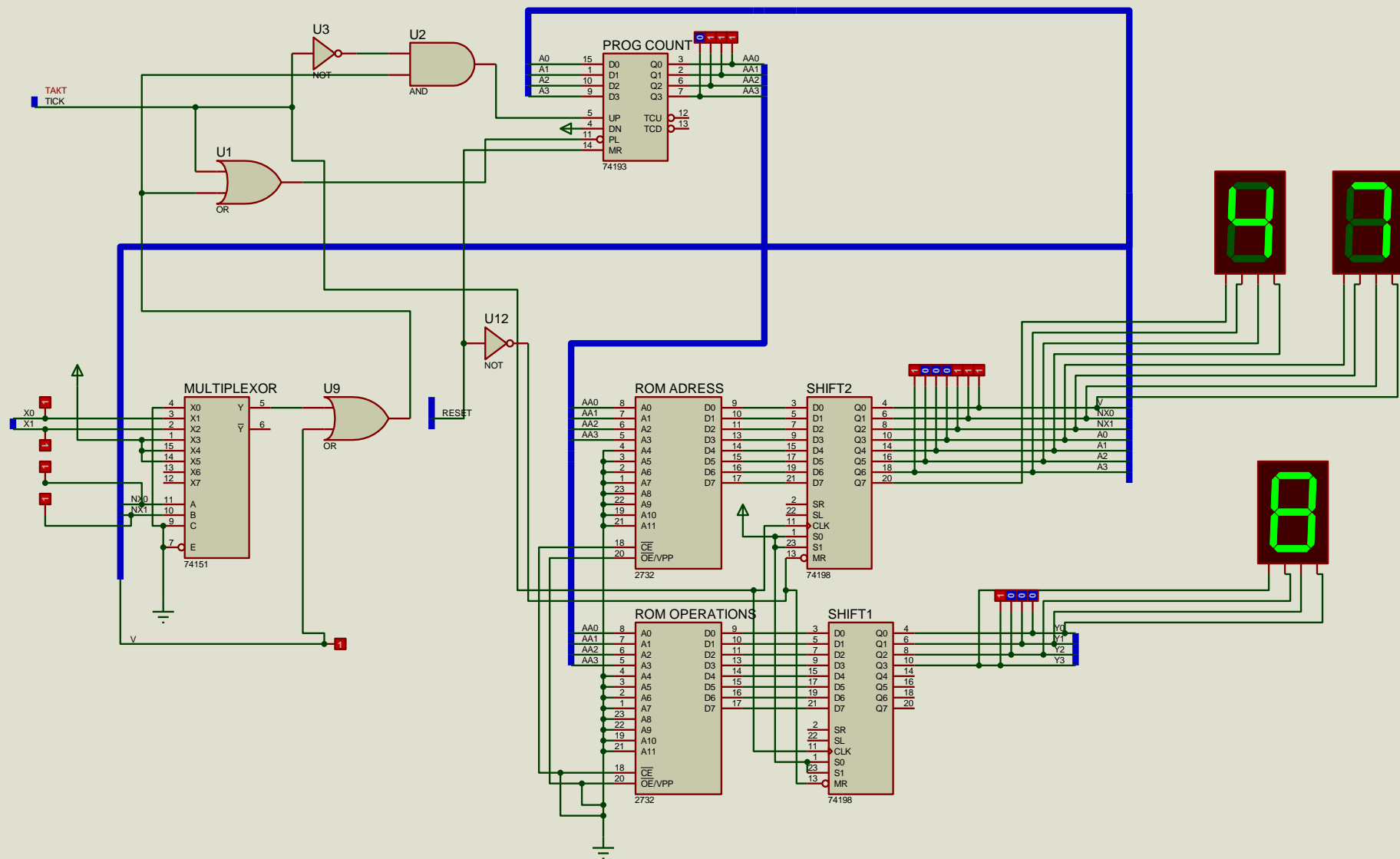


Структура микропрограммного автомата

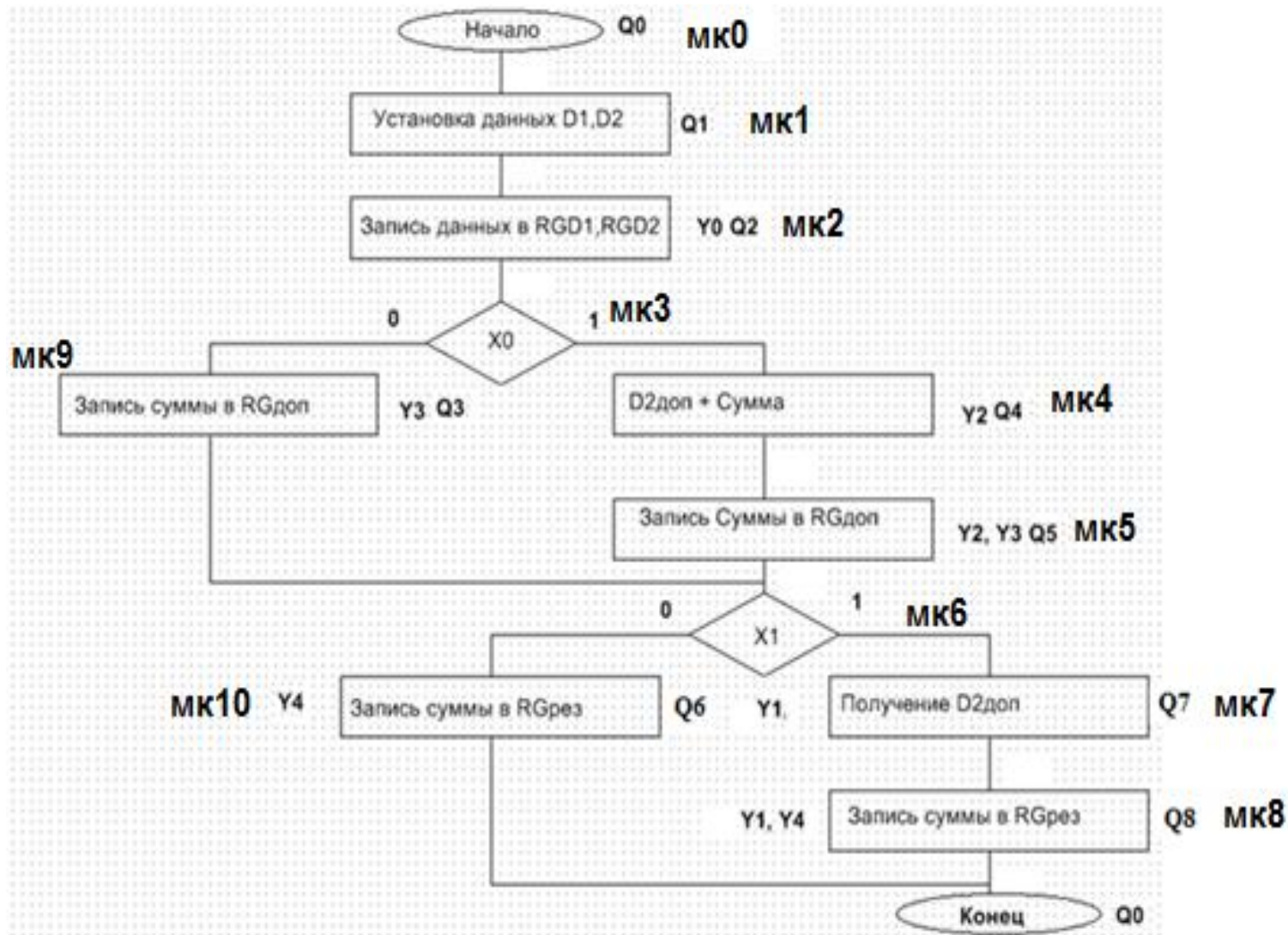


- **ПЗУ** – хранит коды микрокоманд
- **Счётчик-РАМК**(регистр **адреса** микрокоманды), при $V=0$ выполняет загрузку адреса команды со входов D , при $V=1$ выполняется счёт по тактовым импульсам на входе $+1$.
- **РАМК**-регистр микрокоманды, временно хранит **микрокоманду**
- **Бит V** определяет тип микрокоманды
- $V=1$ – **ОМК** – операционная микрокоманда, содержащая сигналы управления $Y - Y_k$.
- $V=0$ – **УМК** – управляющая микрокоманда (команда перехода)
- **Поле N_x** – Закодированный номер проверяемой входной переменной. Это значение подается на адресные входы мультиплексора и разрешает прохождение на его выход значения соответствующей переменной X_i .
- **Поле A** – адрес перехода, подаётся на входы предзагрузки счётчика и в зависимости от значения анализируемой переменной X_i на входе V может записывать в счетчик адрес следующего перехода
- Безусловный переход реализуется путем фиксирования лог. 0 на одном из входов MS .

Микропрограммный автомат



Граф - схема алгоритма сложения



Кодировка входных сигналов автомата x0, x1

Входы данных мультиплексора Адресные входы мультиплексора

Вход	Поле <u>Nx</u> микрокоманды
0 · (лог. 0) Безусловный переход	00
1 · (x0)	01
2 · (x1)	10

Таблица переходов / микропрограмма

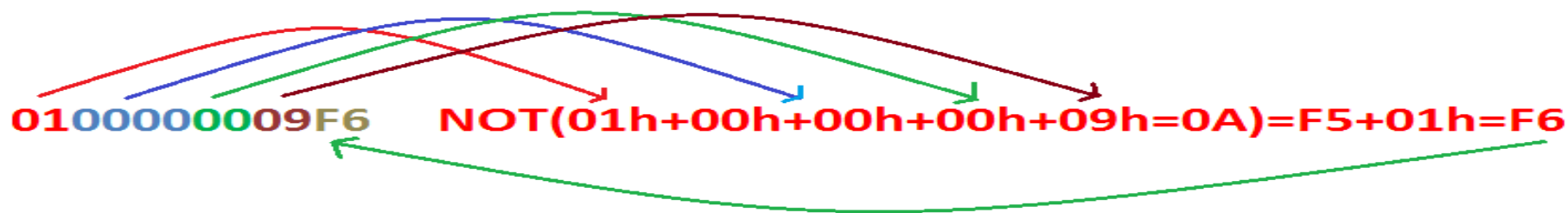
№ МК	Тек. сост.	Адрес на входе ПЗУ				Адрес следующей МК				Поле Nx		V	hex	Выходные сигналы				
		A3	A2	A1	A0	A3	A2	A1	A0	Nx1	Nx0			Y4	Y3	Y2	Y1	Y0
						ПЗУ 1									ПЗУ 2			
Мк0	Q0	0	0	0	0	0	0	0	1	1	1	1	F	0	0	0	0	0
Мк1	Q1	0	0	0	1	0	0	0	0	1	1	1	17	0	0	0	0	0
Мк2	Q2	0	0	1	0	0	0	1	1	1	1	1	1F	0	0	0	0	1
Мк3	Q2	0	0	1	1	0	0	0	0	0	1	0	52	0	0	0	0	1
Мк4	Q4	0	1	0	0	0	0	0	1	1	1	1	2F	0	0	1	0	0
Мк5	Q5	0	1	0	1	0	0	0	0	1	1	1	37	0	1	1	0	0
Мк6	Q5	0	1	1	0	0	0	0	0	1	0	0	64	0	1	1	0	0
Мк7	Q7	0	1	1	1	0	0	0	0	1	1	1	47	0	0	0	1	0
Мк8	Q8	1	0	0	0	0	0	0	1	1	1	1	4F	1	0	0	1	0
Бл	Q8	1	0	0	1	0	0	0	1	0	0	0	08	1	0	0	1	0
Мк9	Q3	1	0	1	0	0	0	0	1	1	1	1	5F	0	1	0	0	0
Бл	Q3	1	0	1	1	0	0	0	0	0	0	0	30	0	1	0	0	0
Мк10	Q6	1	1	0	0	0	0	0	1	1	1	1	6F	1	0	0	0	0
Бл	Q6	1	1	0	1	0	0	0	1	0	0	0	08	1	0	0	0	0

МК

- Микропрограмма занимает 14 ячеек памяти. Разрядность выходных данных ПЗУ определяется: $V + N_x + A + Y = 1 + 2 + 4 + 5 = 12$. надо две ПЗУ.

Создание HEX-файла

- **LLAAATTDCC**
- **LL** - поле длины — показывает количество байт данных (DD) в записи;
- **AA AA**- поле адреса — представляет адрес записи в памяти
- **TT** - поле типа. Оно может принимать следующие значения:
 - 00 запись содержит данные двоичного файла.
 - 01 запись является концом файла.
- **DD** - поле данных.
- **CC** — контрольная сумма.
 - Вычисляется $CC = 01h + NOT(LLh + AAh + AAh + TTh + DDh)$.
- Байт данных 0001001 по адресу 0000 можно представить в hex-формате



Формат hex - файла

:0100000000FF0
:0100010017E7
:010002001FDE
:0100030052AA
:010004002FCC

:01000000000FF
:0100010000FE
:0100020001FC
:0100030001FB
:0100040004F7

Запись микропрограммы

Редактирование компонента

Компонентная ссылка: U2

Компонентное значение: 2732

Image File: 1.hex

File Base Address (hex): 0x0000

File Address Shift (bits): 0

PCB Package: DIL24

Advanced Properties:

Address Access Time: 200ns

Other Properties:

☐ Исключить из Моделирования

☐ Исключить из PCB Layout

☐ Отредактировать свойства как текст

☐ Присоединить модуль иерархии

☐ Скрыть общие пины

Скрыть: ☐

Скрыть: ☐

Hide All

Hide All

Hide All

Hide All

Hide All

OK

Справка

Скрытые пины

Отмена