

УДК 004.31

**ФУНКЦИОНАЛЬНОЕ МОДЕЛИРОВАНИЕ ПОСЛЕДОВАТЕЛЬНОСТНЫХ КОМПЬЮТЕРНЫХ УЗЛОВ И БЛОКОВ С ПОМОЩЬЮ ТАБЛИЧНОГО ПРОЦЕССОРА MICROSOFT EXCEL****Страбыкин Д.А.***ФГБОУ ВО «Вятский государственный университет», Киров, e-mail: Strabykin@mail.ru*

Технология построения действующих функциональных моделей узлов и блоков ЭВМ при проведении практических занятий на компьютерах с использованием табличного процессора Microsoft Excel первоначально ориентированная на комбинационные схемы, развивается применительно к последовательным схемам. Для функционального моделирования последовательных схем предлагается изменить параметры Microsoft Excel, связанные с вычислением формул, путем отключения режима автоматических вычислений и включения режима ручных вычислений, в котором для выполнения пересчета формул необходимо нажимать клавишу F9. Показано, что ячейка листа Microsoft Excel позволяет моделировать функционирование простейших последовательных схем – триггеров. Моделирование триггеров осуществляется путем построения формул функций перехода с последующей их реализацией в виде логических функций электронной таблицы. Приводятся примеры моделей синхронных триггеров четырех типов: D, RS, DV и T. Моделирование более сложного узла с памятью рассматривается на примере 4-разрядного двоичного суммирующего счетчика, построенного на основе T триггеров с установкой в «0». Моделирование вычислительного блока с разработкой экранной формы для проведения экспериментальных исследований иллюстрируется на примере операции деления нацело 8-разрядных двоичных чисел. Вычислительный блок включает: регистры делимого и делителя, вычитатель, счетчик частного и логический элемент ИЛИ-НЕ (формирователь признака нуля в регистре делителя). Выявляется особенность моделирования последовательных схем с помощью Microsoft Excel, заключающаяся в необходимости учитывать порядок пересчета формул в ячейках электронной таблицы после нажатия клавиши F9. Отмечается, что достоинства технологии построения действующих функциональных моделей узлов и блоков ЭВМ с использованием Microsoft Excel, ориентированной на комбинационные схемы, сохраняются при переходе к последовательным схемам.

**Ключевые слова:** функциональные модели узлов и блоков, последовательные схемы, компьютерные практикумы по изучению ЭВМ, применение Microsoft Excel

**FUNCTIONAL MODELING OF SEQUENTIAL COMPUTING BLOCKS AND NODES WITH MICROSOFT EXCEL SPREADSHEET APPLICATION****Strabykin D.A.***Federal State Budgetary Educational Institution of Higher Education «Vyatka State University», Kirov, e-mail: Strabykin@mail.ru*

The article describes the advancement of the technology used to design working functional models for computer nodes and blocks for practical exercises in the Microsoft Excel spreadsheet application. Initially described for combinatorial logic the technology has been adapted for sequential logic. To model sequential logic schemes formulae computation mode in Microsoft Excel is changed from automatic to manual mode, when each time a user wants to re-compute values, they manually trigger it by pressing the F9 key. The paper shows how a spreadsheet cell in Microsoft Excel can be used to model triggers as trivial sequential logic schemes. Triggers are modeled by creating formulae and implementing them with logical functions available in the spreadsheet application. Models for 4 types of synchronous triggers D, RS, DV, and T are described. The paper uses a 4-bit binary cumulative counter built with T-triggers set to «0» as an example of a more complex scheme with memory. The division of two 8-bit integers is used to illustrate the modeling of a computational block together with the design of a screen form. The computational block includes the following components: two registers for the dividend and the divisor, subtractor, quotient counter, and OR-NOT logical element (used to test the divisor for 0). The paper indicates the specific of sequential logic modeling with a spreadsheet application that requires to consider the order in which cells formulae get re-evaluated after a user manually triggers the computation. All advantages of the technology for designing working functional models for computer nodes and blocks using Microsoft Excel previously described for combinatorial logic schemes persist in case of sequential logic schemes.

**Keywords:** functional models of nodes and blocks, sequential logic schemes, practical studies of computers, Microsoft Excel applications

Важную роль в процессе изучения вычислительной техники играет практическая подготовка. Самостоятельная разработка простейших узлов и блоков ЭВМ с последующим построением и экспериментальным исследованием их программных моделей на занятиях в компьютерном классе способствует более глубокому усвоению изучаемого материала. Программные мо-

дели исследуемых узлов и блоков могут создаваться с помощью различных инструментальных средств [1–3] или представлять собой готовые программы [4]. Развитие технологий использования в учебном процессе программных моделей узлов и блоков ЭВМ является актуальной задачей, так как способствует повышению эффективности обучения.

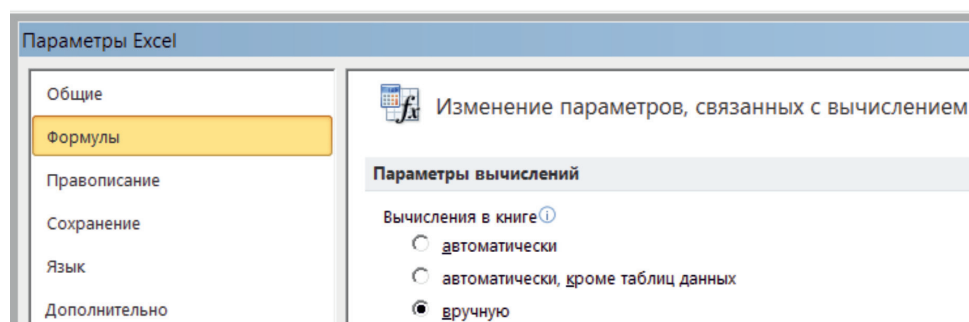


Рис. 1. Изменение параметров вычисления формул

Цель исследования: развитие технологии построения и использования действующих функциональных моделей применительно к узлам и блокам ЭВМ, содержащим память (являющимся последовательными схемами), на основе табличного процессора Microsoft Excel при проведении практических занятий в компьютерном классе.

В технологии построения действующих функциональных моделей узлов и блоков ЭВМ при проведении практических занятий на компьютере с использованием табличного процессора Microsoft Excel, ориентированной на комбинационные схемы, выделяются следующие этапы [5]: анализ задания по разработке функциональной модели; описание работы узла (блока) с помощью формул булевых функций; разработка схемы; разработка экранной формы для экспериментального исследования; построение функциональной модели; экспериментальные исследования функциональной модели; оформление отчета по результатам работы.

Переход от моделирования комбинационных схем к последовательным схемам вносит изменения в содержание этапов технологии построения действующих функциональных моделей узлов и блоков ЭВМ с использованием табличного процессора Microsoft Excel, обусловленные особенностью функционирования последовательных схем. Эта особенность проявляется в зависимости выходного сигнала не только от поступивших в данный момент на входы значений логических переменных, но и от тех значений переменных, которые поступали на входы в предыдущие моменты времени. Для выполнения этого условия значения переменных должны запоминаться узлом (блоком). Функцию запоминания значений логических переменных в последовательных узлах и блоках выполняют элементы памяти, представляющие собой триггеры, способные под воздействием входных сигна-

лов переходить в одно из двух устойчивых состояний и сохранять это состояние при отсутствии входных сигналов.

*Возможность моделирования  
последовательных схем  
в Microsoft Excel*

Электронные таблицы Microsoft Excel в обычном режиме работы хорошо подходят для моделирования комбинационных схем, так как в электронной таблице, как и в комбинационной схеме, изменения значений входных переменных автоматически приводят к изменению (пересчету) значений выходных переменных. Поскольку значения выходных переменных в электронной таблице зависят только от значений входных переменных, моделирование последовательных схем оказывается невозможным.

Между тем моделирование последовательных схем с помощью электронных таблиц Microsoft Excel становится возможным, если отказаться от автоматического вычисления формул при изменении значений входных переменных. Для этого необходимо изменить параметры Microsoft Excel, связанные с вычислением формул (Файл/Параметры/Формулы/Параметры вычислений/Вычисления в книге/вручную) (рис. 1).

В результате будет отключен режим автоматических вычислений и включен режим ручных вычислений, в котором для выполнения пересчета формул необходимо нажимать клавишу F9 (или выбирать команду «Пересчет»). При моделировании узлов и блоков клавиша F9 будет выполнять функции клавиши «Такт».

*Моделирование триггеров с помощью  
табличного процессора Microsoft Excel*

Ячейка листа Microsoft Excel может выступать в качестве элемента памяти, значения входных сигналов которого содержатся в других ячейках, а работа элемента памяти

описывается формулой, занесенной в соответствующую ему ячейку. Новое состояние элемента памяти, зависящее от текущего состояния и значения входных сигналов, формируется после пересчета формулы в соответствующей элементу памяти ячейке при нажатии клавиши F9.

Таким образом, ячейка листа Microsoft Excel позволяет моделировать функционирование простейших последовательностных схем – триггеров. Поскольку изменение состояния триггера происходит под действием специального сигнала, вырабатываемого при нажатии клавиши «Такт», то моделируется работа синхронных триггеров.

Пример синхронного D триггера приведен на рис. 2, где представлено условное графическое изображение (УГО), а также таблица переходов и функция переходов, описывающие функционирование триггера. При отсутствии сигнала синхронизации C ( $C = 0$ ) состояние Q триггера в момент времени  $(t + 1)$  совпадает с состоянием в предыдущий момент времени  $(t)$ :  $Q(t + 1) = Q(t)$ . При поступлении сигнала синхронизации C ( $C = 1$ ) триггер в момент времени  $(t + 1)$  переходит в состояние, определяемое информационным входом D:  $Q(t + 1) = D$ .

При моделировании триггера на листе Microsoft Excel состояние триггера Q представляется содержимым ячейки F3, а сигнал на информационный вход D поступает из ячейки A3. При этом моделирование сигнала синхронизации не связано с содержимым ячеек листа Microsoft Excel и реализуется путем нажатия на клавиатуре компьютера клавиши «Такт» (F9). Принимая во внимание особенность моделирования сигнала синхронизации, а также то, что при отсутствии этого сигнала ( $C = 0$ ) состояние последовательностных схем не изменяется, для упрощения логических формул далее используются формулы, описывающие работу схем при наличии сигнала синхронизации ( $C = 1$ ).

Моделирование триггеров осуществляется путем построения формул функций перехода с последующей их реализацией в виде логических функций Microsoft Excel. Для получения формул функций перехода по словесному описанию работы триггеров могут быть использованы таблицы или графы переходов. На рис. 3 приведены примеры моделей трех типов триггеров: RS, DV и T (с установкой в нулевое состояние).

Для экспериментального исследования работы триггера после ввода логических формул, описывающих его работу, необходимо задать в ячейках, соответствующих входам, значения входных сигналов и нажать клавишу «Такт» (F9). Например, T триггер при  $R = 0$  и  $T = 1$  при каждом нажатии на клавишу «Такт» переключается в противоположное состояние.

#### *Моделирование счетчика с помощью табличного процессора Microsoft Excel*

Моделирование узла с памятью рассмотрим на примере накапливающего счетчика. Счетчик может быть построен на основе рассмотренного выше T триггера с установкой в «0». Пример схемы 4-разрядного двоичного суммирующего счетчика, моделируемого на листе Microsoft Excel, приведен на рис. 4.

Функциональная модель счетчика описывается формулами булевых функций и соответствующими им логическими формулами Microsoft Excel, приведенными в табл. 1.

В моделируемом счетчике используются три комбинации управляющих сигналов:  $R = 1$  – обнуление счетчика;  $R = 0$  и  $T = 1$  – счет;  $R = 0$  и  $T = 0$  – хранение текущего значения счетчика. Если тактовый сигнал, подаваемый с помощью клавиши F9, поступает при  $R = 1$  (содержимое ячейки AB12 Microsoft Excel равно единице), то все триггеры счетчика устанавливаются в нулевое состояние. Если тактовый сигнал поступает при  $R = 0$  и  $T = 1$ , то содержимое счетчика увеличивается на единицу.

F3		fx		=A3																																	
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q																				
1	УГО			Таблица переходов			Функция переходов																														
2																																					
3	0	<table border="1"><tr><td>D</td><td>T</td></tr><tr><td>C</td><td></td></tr></table>		D	T	C		Q	0	<table border="1"><tr><td>C</td><td>D</td><td>Q(t+1)</td></tr><tr><td>0</td><td>0</td><td>Q(t)</td></tr><tr><td>0</td><td>1</td><td>Q(t)</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	C	D	Q(t+1)	0	0	Q(t)	0	1	Q(t)	1	0	0	1	1	1	1) $Q(t+1)=Q(t) \& \neg C \vee D \& C$ или $Q(t+1)=Q(t)$ при $C=0$ и $Q(t+1)=D$ при $C=1$ 2) F3: =A3											
D	T																																				
C																																					
C	D	Q(t+1)																																			
0	0	Q(t)																																			
0	1	Q(t)																																			
1	0	0																																			
1	1	1																																			
4																																					
5	1																																				
6																																					
7																																					
8																																					

Рис. 2. Моделирование синхронного D триггера

[illegible]

a)

The screenshot shows an Excel spreadsheet with the following content:

- Formula Bar:** 
$$F19 = \text{ЕСЛИ}(\text{ИЛИ}(\text{И}(\text{A21}; \text{A19}); \text{И}(\text{НЕ}(\text{A21}); \text{F19}))); 1; 0)$$
- Worksheet Grid:**
  - Row 17: Column L contains the text "Функция переходов".
  - Row 18: Column L contains the text "C=1".
  - Row 19: Column L contains the text "1) Q(t+1)=V&D∨¬V&Q(t)".
  - Row 20: Column L contains the text "2) F19: =ЕСЛИ(ИЛИ(И(A21;A19);И(НЕ(A21);F19)));1;0)".
  - Row 21: Column L contains the text "(D в ячейке A19, V – A21, Q – F19)".
  - Row 22: Column L contains the text "Возможна более короткая формула:".
  - Row 23: Column L contains the text "F19: =ЕСЛИ(A21;A19;F19)".

6)

А27      fx      1

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
25																		
26																		
27	1																	
28	1																	
29	0																	
30																		
31																		
32																		
33																		

УГО

Триггер

с установкой в «0»

Таблица переходов

C=1

	R	T	Q(t+1)
0 0	0	0	Q(t)
0 1	0	1	Q(t)
1 0	1	0	0
1 1	1	1	0

Функция переходов

C=1

1)  $Q(t+1) = \overline{R} \& T \& \overline{Q(t)} \vee R \& \overline{T} \& Q(t) = \overline{R} \& (T \oplus Q(t))$

2) F27: =ЕСЛИ(И(НЕ(A29);ИСКЛЮЧИЛИ(A27;F27)));1;0)

(R в ячейке A29, Q – F27)

6)

Рис. 3. Модели триггеров: RS (а), DV (б) и T (в)

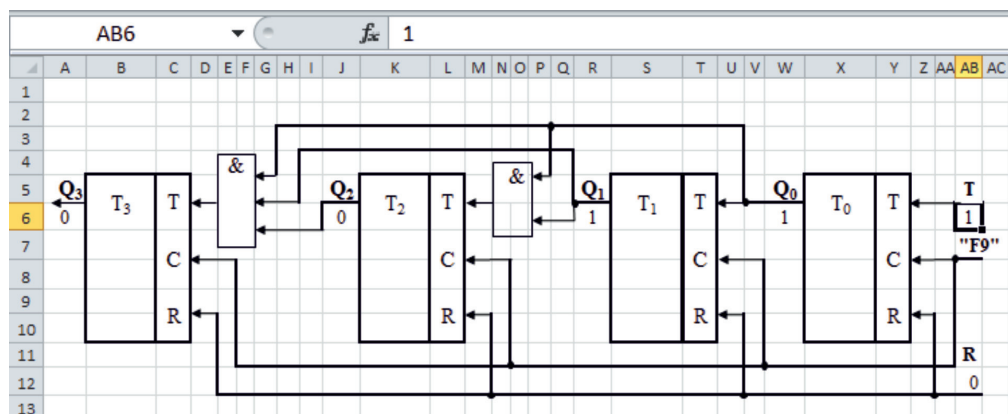


Рис. 4. Экранная форма функциональной модели 4-разрядного двоичного суммирующего счетчика



Таблица 1

## Описание функционирования счетчика

Формула булевой функции	Лист Microsoft Excel	
	Адрес ячейки	Формула
$Q_0(t+1) = \neg R \& (Q_0(t) \oplus T)$	W6	=ЕСЛИ(И(НЕ(Z11);ИСКЛИЛИ(W6;Z6));1;0)
$Q_1(t+1) = \neg R \& (Q_1(t) \oplus Q_0(t))$	R6	=ЕСЛИ(И(НЕ(Z11);ИСКЛИЛИ(R6;W6));1;0)
$Q_2(t+1) = \neg R \& (Q_2(t) \oplus (Q_1(t) \& Q_0(t)))$	J6	=ЕСЛИ(И(НЕ(Z11);ИСКЛИЛИ(J6;И(R6;W6)));1;0)
$Q_3(t+1) = \neg R \& (Q_3(t) \oplus (Q_2(t) \& Q_1(t) \& Q_0(t)))$	A6	=ЕСЛИ(И(НЕ(Z11);ИСКЛИЛИ(A6;И(J6;R6;W6)));1;0)

Необходимо отметить особенность моделирования последовательностных схем с помощью табличного процессора Microsoft Excel, которая обусловлена порядком пересчета формул в ячейках электронной таблицы после нажатия клавиши F9. Пересчет формул в ячейках производится слева направо, сверху вниз. Это означает, что в счетчике на рис. 4 в начале будет вычислено значение старшего разряда ( $T_3$ ), а затем последовательно разрядов  $T_2$ ,  $T_1$ ,  $T_0$ . Несмотря на такой порядок формирования новых значений разрядов, счетчик работает корректно, поскольку используется параллельное формирование переносов. При моделировании счетчика (сумматора) с последовательным переносом ячейки электронной таблицы, соответствующие младшим разрядам, должны располагаться слева от ячеек старших разрядов. В этом случае порядок пересчета формул в ячейках будет обеспечивать распространение переноса от младшего к старшему разряду.

*Моделирование вычислительного блока  
с помощью табличного процессора  
Microsoft Excel*

Моделирование вычислительного блока, содержащего узлы с памятью, рассмотрим на примере операции деления чисел нацело. Для деления нацело 8-разрядного двоичного числа  $X$  на 8-разрядное двоичное число  $Y$  используется следующий алгоритм. Если число  $Y = 0$  (специальный признак  $z = 1$ ), то деление не выполняется. В противном случае из делимого  $X$  вычитается делитель  $Y$ , и если знак разности  $s = 0$  (разность положительное число), то частное  $Z$  увеличивается на единицу, а разность выступает в качестве нового уменьшаемого на следующем шаге. Процесс вычитания продолжается до тех пор, пока не будет получена отрицательная разность ( $s = 1$ ). В этом случае  $Z$  будет представлять собой искомое частное от деления  $X$  на  $Y$  нацело.

Пример схемы 8-разрядного вычислительного блока для выполнения операции деления чисел нацело, моделируемого на листе

Microsoft Excel, приведен на рис. 5. Вычислительный блок включает: регистр делимого  $RX$ , регистр делителя  $RY$ , сумматор (вычитатель)  $SM$ , счетчик частного  $RZ$  и логический элемент ИЛИ-НЕ (для формирования признака  $z$  равенства нулю содержимого регистра  $RY$ ). В регистр  $RX$  при подаче управляющего сигнала  $v_1$  заносится делимое:  $RX = X$ , а при подаче сигнала  $v_3$  – содержимое сумматора:  $RX = SM$ . В регистр  $RY$  при подаче управляющего сигнала  $v_2$  заносится делитель:  $RY = Y$ . Результат вычитания из содержимого регистра  $RX$  содержимого регистра  $RY$  формируется в сумматоре при подаче управляющего сигнала  $v_5$ :  $SM = RX - RY$ . При этом формируется признак знака разности  $s$ , который равен единице, если результат отрицательный и нулю в противном случае. Все разряды сумматора, включая разряд  $s$ , устанавливаются в нулевые состояния при подаче управляющего сигнала  $v_4$ :  $SM = 0$ . К содержимому счетчика  $RZ$  можно добавить единицу с помощью управляющего сигнала  $v_7$ :  $RZ = RZ + 1$ , а с помощью сигнала  $v_6$  – установить в нулевое состояние ( $RZ = 0$ ).

В вычислительном блоке используется сумматор (вычитатель) с последовательным переносом (заемом), работа которого описывается следующими булевыми функциями:

$$r_{n-1} = (x_{n-1} \oplus y_{n-1} \oplus p_{n-1}),$$

$$g_n = \neg x_{n-1} \& y_{n-1} \vee \neg x_{n-1} \& g_{n-1} \vee y_{n-1} \& g_{n-1},$$

где  $x_{n-1}$  и  $y_{n-1}$  – разряды двоичных кодов уменьшаемого и вычитаемого соответственно;  $r_{n-1}$  – разряд разности, а  $g_n$  – разряд заема;  $n = 1, \dots, 8$ .  $g_0 = 0$ . Разряд знака разности  $s = g_8$ .

На рис. 5 над вычитателем ( $SM$ ) отображается 8-разрядное слово разности  $r_0 r_1 r_2 r_3 r_4 r_5 r_6 r_7$ , над которым выводится 8-разрядное слово заемов  $g_0 g_1 g_2 g_3 g_4 g_5 g_6 g_7$ . Для корректной работы вычитателя с последовательным распространением заема разряды вычитателя по отношению к разрядам регистров располагаются в противоположном порядке, который затем восстанавливается в отображении разрядов сумматора ( $SM$ ).

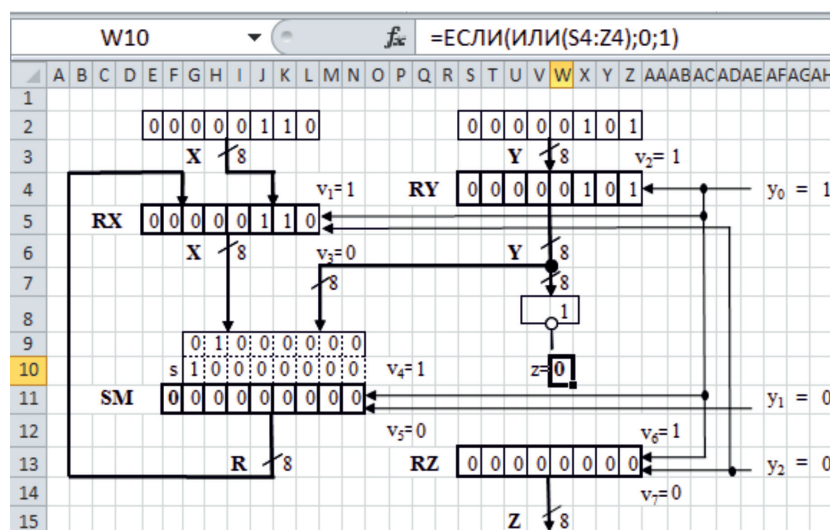


Рис. 5. Экранная форма функциональной модели 8-разрядного вычислительного блока деления чисел нацело

Таблица 2

Выполнение микропрограммы деления чисел нацело

Такт	МК	$y_1 y_2 y_3$	Микрооперации	z	s	RX	RY	SM	RZ
1	1	100	$RX := X; RY := Y; SM := 0; RZ := 0$	0	0	00001111	000001001	00000000	00000000
2	2	010	$SM := RX - RY$	0	0	00001111	000001001	00000110	00000000
3	3	001	$RX := SM; RZ := RZ + 1$	0	0	00000110	000001001	00000110	00000001
4	2	010	$SM := RX - RY$	0	1	00000110	000001001	11111101	00000001
5	4	000	—	0	1	00000110	000001001	11111101	00000001

Микропрограмма деления чисел нацело для схемы, приведенной на рис. 5, состоит из трех различных микрокоманд (МК) и может быть представлена в следующем виде.

#### НАЧАЛО

**МК 1:**  $y_1 = 1; y_2 = 0; y_3 = 0$  ( $v_1$ :  $RX := X$ ;  $v_2$ :  $RY := Y$ ;  $v_4$ :  $SM := 0$ ;  $v_6$ :  $RZ := 0$ );

если  $z = 1$ , то переход к **КОНЕЦ**, иначе

**МК 2:**  $y_1 = 0; y_2 = 1; y_3 = 0$  ( $v_5$ :  $SM := RX - RY$ );

если  $s = 1$ , то переход к **КОНЕЦ**, иначе

**МК 3:**  $y_1 = 0; y_2 = 0; y_3 = 1$  ( $v_3$ :  $RX := SM$ ;  $v_7$ :  $RZ := RZ + 1$ ); переход к **МК 2**

#### КОНЕЦ

В процессе экспериментальных исследований функциональной модели вычислительного блока осуществляется проверка правильности выполнения микропрограммы для всех сочетаний значений логических условий. В табл. 2 приведен пример выполнения микропрограммы по тактам для случая, когда делимое  $X = 00001111$ , а делитель  $Y = 000001001$ .

В примере в процессе деления чисел нацело при первом вычитании делителя из делимого (такт 2, МК 2) получается положительный результат ( $s = 0$ ) и частное увеличивается на единицу (МК 3). Второе вычитание делителя из полученной ранее разности (такт 4, МК 2) приводит к отрицательному результату ( $s = 1$ ), и процесс деления завершается.

#### Заключение

Предлагаемый подход к функциональному моделированию узлов и блоков ЭВМ, представляющих собой последовательные схемы, с помощью табличного процессора Microsoft Excel позволяет включить в учебный процесс самостоятельную разработку обучающимися простейших узлов и блоков ЭВМ с последующим построением и экспериментальным исследованием их программных моделей. В качестве достоинств такого подхода следует отметить: использование популярной инструментальной программы (Microsoft Excel); широкие возможности для формирования

различных вариантов индивидуальных заданий при построении функциональных моделей; открытость для последующего развития и адаптации в зависимости от особенностей учебного процесса и предпочтений преподавателя.

#### Список литературы

1. Алехин В.А. Развитие учебного комплекса по электротехнике, электронике и микроконтроллерам с моделированием в программной среде TINA // Открытое образование. 2017. Т. 21. № 6. С. 57–69.
2. Тюрин С.Ф. Анализ настроек логических элементов при проектировании автомата в системе QuartusII // В мире научных открытий. 2015. № 4 (64). С. 437–453.
3. Лемешко Н.В. Подходы к моделированию цифровых электронных схем // Труды научно-исследовательского института радио. 2008. № 4. С. 65–72.
4. Шогенов А.А., Тлостанов Ю.К. Имитационная модель четырехразрядного регистра сдвига. Свидетельство о регистрации программы для ЭВМ RUS № 2017618381. Заявка № 2017615054 от 30.05.2017. Опубл. 28.07.2017.
5. Страбыкин Д.А. Построение функциональных моделей узлов и блоков в компьютерном практикуме по ЭВМ // Современные наукоемкие технологии. 2019. № 8. С. 188–193.