

УДК 004.31:378.147.88
DOI 10.17513/snt.40216

ФУНКЦИОНАЛЬНАЯ МОДЕЛЬ УЧЕБНОГО ПОТОКОВОГО ПРОЦЕССОРА

Страбыкин Д.А.

ФГБОУ ВО «Вятский государственный университет», Киров, e-mail: Strabykin@mail.ru

Цель исследования: развитие технологии построения и применения в учебном процессе функциональных моделей вычислительных устройств с помощью Microsoft Excel для распространения ее на вычислительные структуры, состоящие из параллельно работающих блоков, с демонстрацией разработки модели учебного потокового процессора ЭВМ. Рассматривается представление алгоритма в виде информационного графа и списочной формы потокового процессора, отражающей не единственную последовательность операций, а общую структуру алгоритма. В потоковом процессоре переход к выполнению операции осуществляется, как только создаются условия, характеризующие возможность ее выполнения. Приводится структура и алгоритм работы учебного потокового процессора, содержащего следующие блоки: управления; памяти шаблонов; очередей шаблонов; выполнения коротких и длинных операций; записи результатов в память шаблонов. Рассматривается построение функциональной модели учебного потокового процессора, включающее: определение программистской структуры и формата шаблонов; разработку экранной формы со структурой процессора и дополнительными средствами модели (управления, ввода данных, анализа работы процессора); реализацию функций узлов и блоков процессора с помощью формул Microsoft Excel, обеспечивающую параллельную работу блоков в четырех режимах работы процессора (полутакт, такт, шаг и автомат). Приводится пример экспериментальных исследований решения задачи потоковым процессором, описываемой информационным графом.

Ключевые слова: потоковые процессоры, действующие функциональные модели, компьютерные практикумы по ЭВМ, применение Microsoft Excel

A FUNCTIONAL MODEL OF A TRAINING DATAFLOW PROCESSOR

Strabykin D.A.

Vyatka State University, Kirov, e-mail: Strabykin@mail.ru

Purpose of work: development of technology for construction and application of functional models of computing devices in the educational process using Microsoft Excel to extend it to computing structures consisting of parallel working units, with demonstration of creation and experimental study of a model of a training dataflow computer processor. The presentation of the algorithm in the form of an information graph and a list form of a dataflow processor, reflecting not a single sequence of operations, but the general structure of the algorithm, is considered. In a dataflow processor, an operation is transitioned as soon as conditions are created that characterize the possibility of its execution. The structure and algorithm of the training dataflow processor is given, which contains the following units: controls; template memory; template queues; performing short and long operations; writing results to template memory. It is considered the construction of a functional model of a training dataflow processor, including: determination by a programmer of the structure and format of templates; development of a screen form with a processor structure and additional model tools (control, data input, analysis of processor operation); implementation of functions of processor units and units using Microsoft Excel formulas, which provides parallel operation of units in four modes of processor operation (half-cycle, cycle, step and automatic). An example of experimental studies of solving the problem by a dataflow processor described by an information graph is given.

Keywords: dataflow processors, functional models, practical studies of computers, Microsoft Excel applications

Введение

Одно из основных направлений развития процессоров (ПР) современных вычислительных машин и систем связано с ориентацией на достижение максимальной производительности при ограничениях, накладываемых на другие технические характеристики. При этом широко используются различные подходы к организации параллельной обработки данных, реализуемые путем параллельного функционирования блоков ПР. Примером такого подхода может служить применение потокового управления вычислениями. В потоковых ПР в отличие от директивного управления

по программному счетчику команды выполняются, когда становятся доступными их операнды [1-3]. Важную роль при изучении ПР играет проведение обучающимися экспериментальных исследований не учебных и учебных процессоров с помощью программных средств моделирования. В этом случае могут быть использованы как программные средства, созданные для разработки и исследования вычислительных процессов и структур [4-6], так и средства, предназначенные для обучения [7-9]. В частности, представляют интерес учебные процессоры в виде программных моделей, отражающих наиболее существенные

аспекты рассмотрения ПР на конкретном этапе их изучения.

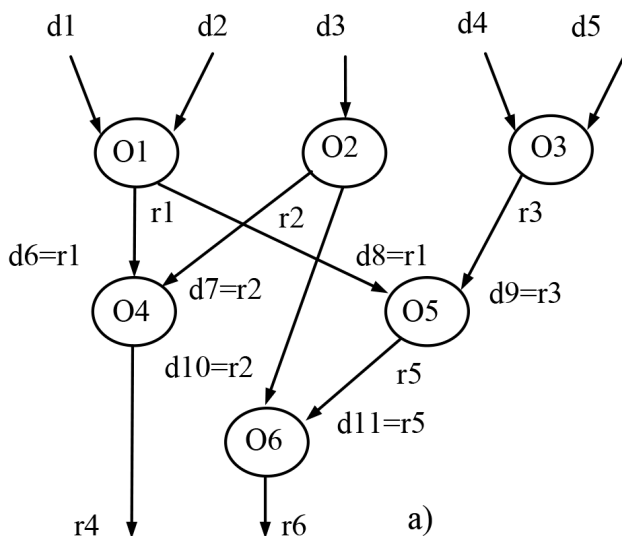
Цель исследования: развитие технологии построения и применения в учебном процессе функциональных моделей вычислительных структур, состоящих из параллельно работающих блоков, позволяющей создавать и экспериментально исследовать модели учебных потоковых процессоров ЭВМ с помощью Microsoft Excel.

Списочная форма потокового процессора

Алгоритм может быть представлен в виде информационного графа, являющегося однонаправленной сетью или ациклическим графом, узлы которого поставлены в соответствие единицам обработки данных (операциям), а однонаправленные дуги – путям передачи данных между ними – информационным связям. Программа представляет собой списочную форму описания информационного графа и состоит из последовательности команд. В потоковом процессоре переход к выполнению операции осуществляется, как только создаются условия, характеризующие возможность ее выполнения. Такой способ управления называют также управлением потоком данных. Операция может быть выполнена, если на соответствующую вершину информационного графа по входным дугам поступили все необходимые операнды. Говорят, что такая вершина «сработала», и после этого ее результат по направленным дугам передается на другие вершины. Результат решения задачи будет получен после срабатывания конечных вершин в информационном графе.

Списочная форма потокового ПР отражает не единственную последовательность операций, а общую структуру алгоритма. Элемент такого списка, часто называемый шаблоном, может быть представлен в следующем виде: $O, F, \{d\}, \{ar\}$, где O определяет примитивную операцию, соответствующую срабатыванию вершины данной операции в информационном графе (спусковая функция); $\{d\}$ – множество полей операндов, заполняемых по мере выполнения предшествующих операций (срабатывания предшествующих вершин в графе); эти операнды могут использоваться как для выполнения операции O , так и для вычисления спусковой функции F ; $\{ar\}$ – множество полей назначения, указывающих, каким шаблонам должны быть переданы промежуточные результаты выполнения операции или куда следует направить полученные конечные результаты.

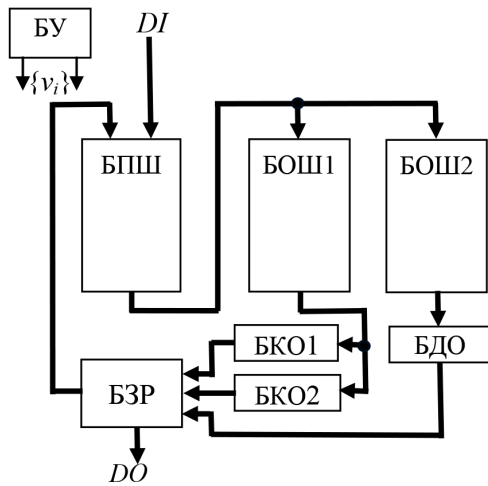
Пример информационного графа и списочной формы для потокового процессора приведены на рис. 1, где ri – результат операции, а ki – шаблон с номером i ($i=1,2,\dots,6$); $kj/n \in \{ar\}$ – n -е поле назначения операнда ($n \in \{1,2\}$) j -го шаблона ($j=1,2,\dots,J$), в который передается результат операции; $Ti(x,x)$ – тег (играет роль спусковой функции F) используется для фиксации наличия операндов, необходимых для выполнения i -го шаблона: $Ti(x,x)$ – операндов нет, $Ti(1,x)$ – получен только первый операнд, $Ti(x,1)$ – получен только второй операнд, $Ti(1,1)$ – получены оба операнда (шаблон готов к выполнению).



- $k1: T1(x,x) O1 (d1,d2), (k4/1,k5/1);$
 $k2: T2(1,x) O2 (d3), (k4/2,k6/1);$
 $k3: T3(x,x) O3 (d4,d5), (k5/2);$
 $k4: T4(x,x) O4 (r1,r2), (r4);$
 $k5: T5(x,x) O5 (r1,r3), (k6/2);$
 $k6: T6(x,x) O6 (r2,r5), (r6).$

б)

Рис. 1. Пример информационного графа (а) и списочной формы (б) потокового ПР



а)

H	k	T	O	(d)	(r)
1	k1	T1(1,1)	O1	(d1,d2)	(k4/1,k5/1)
1	k2	T2(1,1)	O2	(d3)	(k4/2,k6/1)
1	k3	T3(1,1)	O3	(d4,d5)	(k5/2);
1	k4	T4(1,1)	O4	(r1,r2)	(DO)
1	k5	T5(1,1)	O5	(r1,r3)	(k6/2)
1	k6	T6(1,x)	O6	(r2,r5)	(DO)
2	k4	T4(1,1)	O4	(r1,r2)	(DO)
2	k5	T5(1,1)	O5	(r1,r3)	(k6/2)
2	k6	T6(1,1)	O6	(r2,r5)	(DO)
3	k6	T(1,1)	O6	(r2,r5)	(DO)

б)

Рис. 2. Пример структуры (а) и рабочих циклов (б) потокового ПР

Структура и рабочий цикл учебного потокового процессора

Пример структуры учебного потокового ПР приведен на рис. 2а, где БУ – блок управления; БПШ – блок памяти шаблонов; БОШ1 – блок очереди шаблонов коротких, а БОШ2 – длинных операций; БКО1 и БКО2 – блоки выполнения коротких, а БДО – длинных операций; БЗР – блок записи результатов в память шаблонов. Подлежащие выполнению шаблоны предварительно записываются БПШ по входу DI, а результаты вычислений считываются из БПШ с помощью выхода DO. Блок управления вырабатывает множество необходимых управляющих сигналов {v_i}.

В каждом рабочем цикле потокового ПР выполняются следующие действия.

– В начале цикла в БПШ анализируется поле тегов всех шаблонов, и выделяются готовые к выполнению шаблоны, у которых T=(1,1). Выделенные шаблоны последовательно передаются в соответствующие очереди: шаблоны коротких операций – в БОШ1, а длинных – в БОШ2.

– При наличии свободного блока операций в него из соответствующей очереди помещается готовый к выполнению операции шаблон. После выполнения операции, как только окажется свободным БЗР, полученный в блоке операций результат вместе со значениями полей назначений пересылается в БЗР.

– БЗР записывает в БПШ результат операции, в соответствии с заданными полями назначений во входные поля операндов других шаблонов. Причем, при записи результата в поле операнда шаблона, устанавлива-

ется в единицу соответствующий полю бит тега данного шаблона.

– После записи результатов операций, выделенных в начале цикла готовых к выполнению шаблонов, эти шаблоны удаляются из БПШ. Проверяется, все ли шаблоны удалены из БПШ. Если в БПШ есть шаблоны, то выполняется следующий рабочий цикл. В противном случае процесс вычислений завершается.

Выполнение списочной формы потокового ПР, приведенной на рис. 1б, требует три рабочих цикла (H=1,2,3) (рис. 2б). В начале первого цикла (H=1) в БПШ анализируется поле тегов всех шаблонов, и выделяются готовые к выполнению шаблоны (k1, k2, k3), у которых T=(1,1). Выделенные шаблоны последовательно передаются в соответствующие очереди: шаблоны коротких операций (k2, k3) – в БОШ1, а длинных (k1) – в БОШ2. Поскольку блоки операций свободны, то в них из соответствующих очередей помещаются готовые к выполнению операции шаблоны (k1, k2, k3). После выполнения операций, по мере освобождения БЗР, полученные в блоках операций результаты со значениями полей назначения (r1,k4/1,k5/1; r2,k4/2,k6/1; r3,k5/2) пересылаются в БЗР. Этот блок последовательно записывает в БПШ пересланный результат операции в соответствии с заданными полями назначений во входные поля операндов других шаблонов: результат r1 – в первые поля данных шаблонов k4 и k5; результат r2 – во второе поле данных шаблона k4 и первое поле данных шаблона k6; результат r3 – во второе поле данных шаблона k5. При этом теги шаблонов, в которые был

записан результат, принимают следующие значения: T4(1,1), T5(1,1) T6(1,x). После записи результатов операций, выделенных в начале цикла готовых к выполнению шаблонов (k1, k2, k3), эти шаблоны удаляются из БПШ. Далее ПР выполняет второй (H=2) и третий (H=3) рабочий цикл, пока из БПУ не будут удалены все шаблоны.

Структура процессора и экранная форма для проведения экспериментальных исследований

Разработка функциональной модели процессора осуществляется на основе структуры и рабочего цикла учебного потокового ПР. Архитектура процессора включает программистскую (логическую) структуру и внутренний язык, используемый для составления списочной формы (программы потокового процессора). Программистская структура представляет со-

бой 16 32-разрядных регистров шаблонов, в которые помещается списочная форма (программа) процессора. Основу внутреннего языка составляет шаблон, формат которого приведен на рис. 3, где O – код операции, в котором нулевое значение в старшем разряде определяет код короткой, а единичное – длинной операции; d1 – поле первого, а d2 – второго операнда; ar1=j1.n1 и ar2=j2.n2 – поля назначения, в каждом из этих полей старшие четыре разряда (j1 и j2) содержат номер регистра шаблона, в который следует записать результат операции, а младший разряд (n1 и n2) определяет поле данных в этом шаблоне (если n=0, то d1, если n=1, то d2); T=t1.t2 – двухразрядное поле тега, в котором разряд t1 соответствует полю операнда d1, а t2 – d2, при единичном значении разряда тега операнд находится в шаблоне в соответствующем поле операнда, а при нулевом – отсутствует.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O				d1								d2								ar1				ar2				T			
4				8								8								5				5				2			

Рис. 3. Формат шаблона

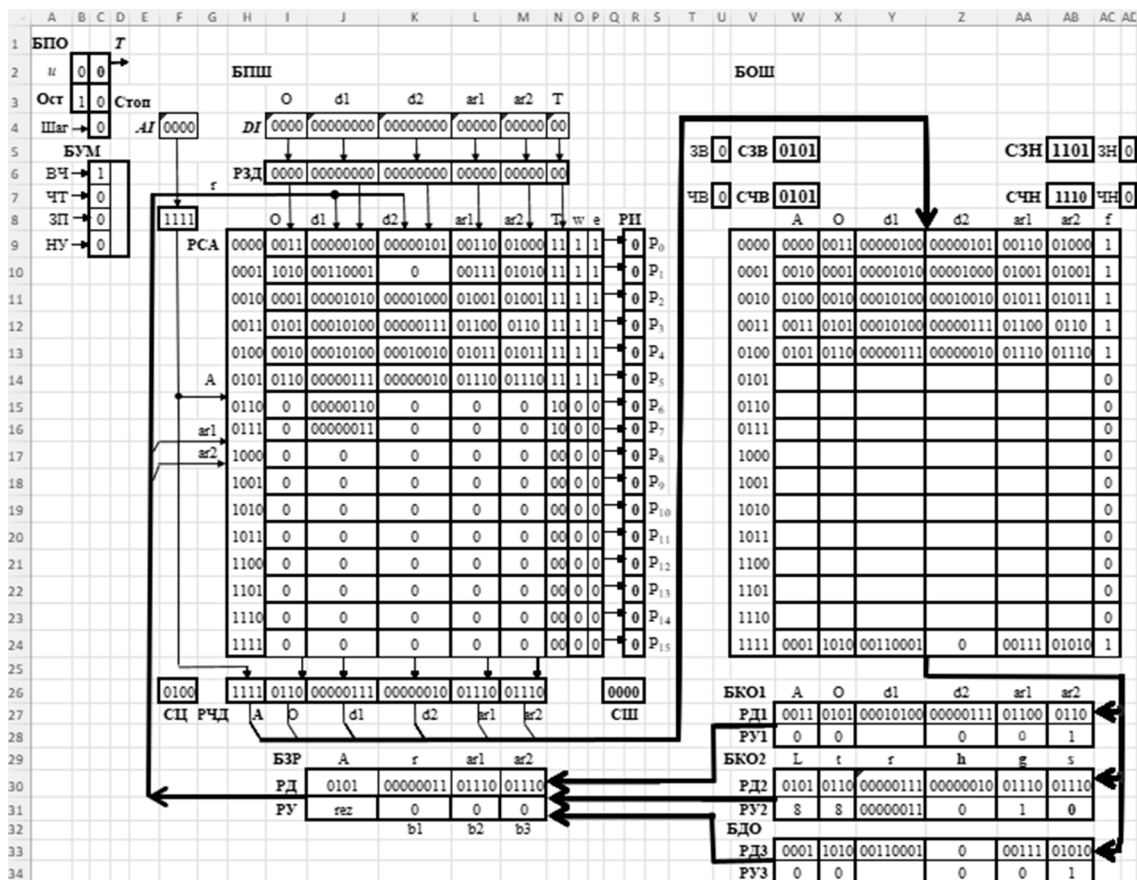


Рис. 4. Экранная форма для экспериментального исследования процессора (левая часть)

Структура моделируемого процессора приведена на разработанной экранной форме (рис. 4) и отличается от структуры учебного процессора (рис. 2а) наличием блока пуска-останова (БПО), более детальным представлением состава БПШ, БОШ, БКО1, БКО2, БДО и БЗР, а также введением блока управления моделью (БУМ). Кроме того, БОШ представляет собой две совмещенные на одном накопителе из 16 регистров линейные очереди: верхнюю (заполняется от младших номеров регистров к старшим) и нижнюю (заполняется от старших номеров регистров к младшим). Из верхней очереди шаблоны поступают в БКО1 или БКО2, а из нижней – в БДО.

На рис. 4 использованы следующие обозначения узлов и сигналов (в квадратных скобках указаны ссылки на ячейки Excel, в которых отображаются их состояния).

БПО: Ост[V3] и Стоп[C3] – осведомительные сигналы, принимающие единичное значение при завершении выполнения всей программы и одного рабочего цикла соответственно. Шаг[C4] – поле управляющего сигнала, позволяющего перевести процессор в режим, при котором после выполнения одного рабочего цикла ПР формирует сигнал Стоп[C3] и останавливается. Для выполнения ПР следующего шага пользователю необходимо ввести с клавиатуры в ячейку С4 в десятичной системе счисления номер следующего шага и нажать клавишу F9. Для отмены пошагового режима в ячейку С4 вводится 0.

БУМ: ВЧ[C6] – управляющий сигнал, переводящий модель ПР в режим вычислений; ЧТ[C7] – чтения содержимого БПШ; ЗП[C8] – записи в БПШ; НУ[C9] – начальной установки всех блоков. Значения всех сигналов в БУМ задаются пользователем с клавиатуры, путем записи нуля или единицы в соответствующую ячейку Excel.

БПШ: РСА[F8] – регистр-счетчик адреса, позволяющий адресовать один из 16 регистров шаблонов для записи и чтения; РЗД[I6:N6] – регистр записи данных в регистр шаблона по адресу в РСА при подаче управляющего сигнала ЗП[C8]; АI[F4] – поле ввода данных с клавиатуры в РСА, а ДI[I4:N4] – в РЗД; w, e – дополнительные разряды в регистре шаблона, устанавливаемые в единичное состояние при копировании шаблона в БОШ (w) и после завершения выполнения шаблона при записи его результата в регистры БПШ (e); РИ[R9:R24] – регистр индикаторов, каждый разряд которого (p_0, p_1, \dots, p_{15}) соответствует одноименному регистру шаблона и устанавливается в единичное состояние в режиме вычислений, когда тег

шаблона в этом регистре равен коду «11» (шаблон готов к выполнению); СЦ[F26] – счетчик рабочих циклов; РЧД[H26:M26] – регистр чтения данных из регистра шаблона по адресу в РСА при подаче управляющего сигнала ЧТ[C7], а также в режиме вычислений; СШ[Q26] – счетчик шаблонов, готовых в текущий момент к выполнению, но выполнение которых еще не завершено.

БОШ: СЗВ[W5] – счетчик записи верхний с сигналом ЗВ[U5] – записи, СЧВ[W7] – счетчик чтения верхний с сигналом ЧВ[U7] – чтения для верхней очереди, растущей от младших номеров регистров к старшим; СЗН[AB5] – счетчик номера нижний с сигналом ЗН[AD5] – записи, СЧН[AB7] – счетчик чтения нижний с сигналом ЧН[AD7] – чтения для нижней очереди, растущей от старших номеров регистров к младшим; А – поле адреса шаблона в БПШ; f – дополнительный разряд в регистре очереди, который устанавливается в единичное состояние, когда шаблон пересылается в блок операций.

БКО1, БКО2, БДО – блоки операций. БКО1 содержит: регистр данных РД1[W27:AB27], в который поступает шаблон из БОШ, и регистр управления РУ1[W28:AB28], имеющий следующие поля: L – число тактов, требуемое для выполнения операции; t – текущее число выполненных тактов; r – результат операции; h – признак поступления шаблона для вычисления в блок операций (при h=1 шаблон поступил в РД); g – признак готовности результата (g=1 при поступлении результата в поле r); s – признак готовности блока операций принять новый шаблон (при s=1 блок готов принять новый шаблон). БКО2 и БДО содержат аналогичные регистры: БКО2 – РД2[W30:AB30] и РУ2[W31:AB31], БДО – РД3[W33:AB33] и РУ3[W34:AB34].

БЗР: РД[J30:M30] – регистр данных, в который из блока операций поступают: А – адрес выполненного шаблона в БПШ, r – результат операции, а также содержимое полей назначения ar1 и ar2; РУ[J31:M31] – регистр управления, включающий следующие поля: rez – резерв; b1 – признак поступления в РД новых данных из БКО1, b2 – из БКО2 и b3 – из БДО (признаки устанавливаются в единичное состояние при поступлении новых данных). Результат r может одновременно записываться в поля данных одного или двух шаблонов регистров БПШ в соответствии с содержимым полей назначения ar1 и ar2 в РД.

Кроме блоков, приведенных на рис. 4, в состав модели ПР входят еще два блока: ПД БПШ – пультовый дублер блока памяти шаблонов и БНО – блок настройки операций (рис. 5).

	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	AT	AU	AV
1																
2	ПД БПШ						БНО									
3																
4																
5																
6																
7																
8	A	O	d1	d2	ar1	ar2	T		O	L	r	rd	S			
9	0000	0011	00000100	00000101	00110	01000	11		0000	1	rez	rez				
10	0001	1010	00110001	0	00111	01010	11		0001	2						+
11	0010	0001	00001010	00001000	01001	01001	11		0010	2						-
12	0011	0101	0	0	01100	0110	00		0011	8						x
13	0100	0010	0	0	01011	01011	00		0100	4	rez	rez				/
14	0101	0110	0	0	01110	01110	00		0101	8						%
15	0110	0	0	0	0	0	00		0110	8		0000011				//
16	0111	0	0	0	0	0	00		0111	2	rez	rez				
17	1000	0	0	0	0	0	00		1000	2	rez	rez				
18	1001	0	0	0	0	0	00		1001	2	rez	rez				
19	1010	0	0	0	0	0	00		1010	16		rez				√
20	1011	0	0	0	0	0	00		1011	2	rez	rez				
21	1100	0	0	0	0	0	00		1100	2	rez	rez				
22	1101	0	0	0	0	0	00		1101	2	rez	rez				
23	1110	0	0	0	0	0	00		1110	2	rez	rez				
24	1111	0	0	0	0	0	00		1111	2	rez	rez				

Рис. 5. Экранная форма для экспериментального исследования процессора (правая часть)

ПД БПШ представляет собой таблицу для пультового ввода данных в БПШ, в которой заданы номера всех регистров (поле A). Предварительно введенные в таблицу двоичные коды шаблонов переносятся в отдельные регистры БПШ после задания режима начальной установки всех блоков модели (НУ[С9]=1) и нажатия клавиши F9. При этом в неиспользуемые поля могут быть внесены символы «0».

БНО отображается в виде таблицы, в которой заданы коды операции (поле O). В столбце L указывается число тактов, необходимых БКО1 (БКО2) и БДО для выполнения каждой используемой в процессоре

операции. В столбце r помещаются результаты вычислений операций с заданным кодом операции в БКО1 (0000-0111) и БДО (1000-1111), а в столбце rd – в БКО2 (0000-0111). Столбец S может быть использован для указания символа реализуемой операции. Временно неиспользуемые ячейки таблицы помечаются символами «rez». Операции выполняются с помощью заранее введенных в ячейки столбцов r и rd формул Microsoft Excel над данными, находящимися в соответствующем блоке операций. Например, формулы для умножения целых чисел с кодом операции O=0011 для БКО1 и БКО2 имеют вид:

AT13: =ЕСЛИ(И(X27=>»0011»;W28=X28;W28>0);ОСНОВАНИЕ(ПРОИЗВЕД(ДЕС(Y27;2);ДЕС(Z27;2));2;8);» «);

AU13: =ЕСЛИ(И(X30=>»0011»;W31=X31;W31>0);ОСНОВАНИЕ(ПРОИЗВЕД(ДЕС(Y30;2);ДЕС(Z30;2));2;8);» «).

Результаты вычислений из БНО передаются в поля r РУ соответствующих блоков операций.

Реализация функций узлов и блоков с помощью формул Microsoft Excel

Разработка функциональной модели процессора производится с использованием подхода, основанного на пословном моделировании узлов и блоков процессора с помощью стандартных функций Excel, когда в ячейках могут находиться многоразрядные двоичные коды, а вычисления производятся в десятичной системе счисления. При этом сохраняется представление состояний входов и выходов узлов и блоков на экранных формах в двоичной системе счисления. Такой подход в отличие от поразрядного моделирования работы узлов и блоков в двоичной системе счисления позволяет значительно уменьшить объем работы по моделированию ПР за счет сокращения числа и сложности формул, требует меньшего количества ячеек Excel [9]. Рассмотрим реализацию функций узлов и блоков потокового ПР более подробно.

В состав БПО входит генератор тактовых импульсов (ГТИ). Используется известная модель ГТИ на основе Т триггера: $T(t+1)=u(\tau \oplus T(t))$, где $T(t)$ – состояние триггера до подачи сигнала τ , формируемого при нажатии клавиши F9, а $T(t+1)$ – после подачи этого сигнала, u – состояние управляющего входа (при $u=0$ состояние выхода $T(t+1)$ генератора принимает нулевое значение и сохраняет его при нажатиях клавиши F9) [9]. Логическая формула Т триггера представляется следующей формулой Excel: C2: =ЕСЛИ(И(B2;НЕ(C2));1;0). Управляющий вход u принимает нулевое значение, прекращающее формирование тактовых сигналов, в одном из трех случаев: при начальной установке (НУ[C9]=1); получении сигнала Ост[B3]=1 или Стоп[C3]=1. Формирование этого значения осуществляется в ячейке В2 с помощью формулы: В2: =ЕСЛИ(ИЛИ(C9;В3;C3);0;1)). В свою очередь значения сигналов Ост[B3] и Стоп[C3] определяются следующими формулами:

В3: =ЕСЛИ(C9;0;ЕСЛИ(И(F8="1111";Q26="0000");1;В3));

С3: =ЕСЛИ(ИЛИ(C9;НЕ(C4=ДЕС(F26;2)));0;ЕСЛИ(И(C4>0;F8="0000";C4=ДЕС(F26;2));1;C3)).

Примеры описаний функций узлов БПШ, БОШ, БКО и БЗР приведены в табл. 1.

Таблица 1

Описание функционирования блоков процессора с помощью формул Microsoft Excel

Блок\узел	Формула Microsoft Excel
БПШ\РСА[F8]	=ЕСЛИ(C9;F4;ЕСЛИ(И(C2=0;C6;В3=0;C3=0);ПСТР(ОСНОВАНИЕ(ДЕС(F8;2)+1;2;5);2;5);F8))
БПШ\РЗД-О[I6]	=ЕСЛИ(C9;»0000»;ЕСЛИ(И(C2=1;C8=1);I4;I6))
БПШ\РШО-О[I9]	=ЕСЛИ(\$C\$9;АН9;ЕСЛИ(И(\$C\$2;\$C\$8;\$F\$8=\$H9);I\$6;I9))
БПШ\РШО-ar1[L9]	=ЕСЛИ(\$C\$9;АК9;ЕСЛИ(И(\$C\$2;\$C\$8;\$F\$8=\$H9);L\$6;L9))
БОШ\СЗВ[W5]	=ЕСЛИ(C9;»0000»;ЕСЛИ(W5=»10000»;»0000»;ЕСЛИ(И(ЛЕВСИМВ(I26)=»0»;ВПР(\$H\$26;H9:R24;11);ВПР(\$H\$26;H9:R24;9)=0;В2;C2;ИЛИ(И(F26=»0000»;F8>W5);НЕ(F26=»0000»)))));ПСТР(ОСНОВАНИЕ(ДЕС(W5;2)+1;2;5);2;5);W5))
БОШ\ЯПО-А[W9]	=ЕСЛИ(\$C\$9;» «;ЕСЛИ(И(ИЛИ(И(\$U\$5;ЛЕВСИМВ(\$I\$26)=»0»; \$W\$5=\$V9);И(\$AD\$5;ЛЕВСИМВ(\$I\$26)=»1»; \$AB\$5=\$V9));\$B\$2;\$C\$2=0);H\$26;W9))
БКО1\РД1-А[W27]	=ЕСЛИ(C9;» «;ЕСЛИ(И(U7;C2;АВ28);ВЫБОР(ДЕС(W7;2)+1;W9;W10;W11;W12;W13;W14;W15;W16;W17;W18;W19;W20;W21;W22;W23;W24);W27))
БКО1\РУ1-L[W28]	=ЕСЛИ(ИЛИ(C9;АВ28);0;ЕСЛИ(И(C2=0;Z28;НЕ(X27=" "));ВЫБОР(ДЕС(X27;2)+1;A9;A10;A11;A12;A13;A14;A15;A16;A17;A18;A19;A20;A21;A22;A23;A24);W28))
БКО1\РУ1-t[X28]	=ЕСЛИ(ИЛИ(C9;АВ28);0;ЕСЛИ(И(C2=0;Z28;X28<W28);X28+1;X28))
БКО1\РУ1-r[Y28]	=ЕСЛИ(ИЛИ(C9;АВ28);» «;ЕСЛИ(И(W28=X28;W28>0;НЕ(X27=» «));ВЫБОР(ДЕС(X27;2)+1;AT9;AT10;AT11;AT12;AT13;AT14;AT15;AT16);Y28))
БЗР\РД-r[K30]	=ЕСЛИ(C9;» «;ЕСЛИ(И(C2;АА34);Y34;ЕСЛИ(И(C2;АА28);Y28;ЕСЛИ(И(C2;АА31);Y31;K30))))
БЗР\РД-ar1[L30]	=ЕСЛИ(C9;» «;ЕСЛИ(И(C2;АА34);АА33;ЕСЛИ(И(C2=1;АА28);АА27;ЕСЛИ(И(C2;АА31);АА30;L30))))

Особенности рабочего цикла процессора

Моделируется работа процессора по тактам, вырабатываемым ГТИ, входящим в БПО. Процессор выполняет вычисления в соответствии со списочной формой, представленной в виде шаблонов заданного формата и записанной в БПШ. Вычислениям предшествует начальная установка ПР, которая производится путем нажатия клавиши F9 после ввода следующих значений управляющих сигналов БУМ: $VЧ[C6]=ЧТ[C7]=ЗП[C8]=0$, $НУ[C9]=1$. В процессе начальной установки предварительно введенная в пультовый дублиер БПШ списочная форма копируется в БПШ. Для перевода модели ПР в режим вычислений в БУМ задается следующий набор управляющих сигналов: $VЧ[C6]=1$, $ЧТ[C7]=ЗП[C8]=НУ[C9]=0$.

В каждом рабочем цикле в модели потокового ПР выполняются следующие действия.

Блок памяти шаблонов. В БПШ анализируется поле тегов T всех шаблонов, и выделяются готовые к выполнению шаблоны: одноименные разряды РИ шаблонов, у которых $T=(1,1)$, устанавливаются в единичные состояния. Выделенные шаблоны последовательно считываются из БПШ и записываются в БОШ1 или БОШ2. При этом дополнительный разряд w регистра БПШ, из которого был считан шаблон, устанавливается в единичное состояние, фиксируя факт передачи шаблона из БПШ в БОШ. Кроме того, в БПШ записывается результат операции из БЗР, в соответствии с заданными полями назначений во входные поля операндов, и записываются единицы в соответствующие поля битов тегов T данных шаблонов. В дополнительный разряд регистра шаблона e , в котором находится выполнявшийся шаблон, записывается единица. Таким образом фиксируется факт выполнения шаблона.

– **Блок очередей шаблонов.** В БОШ шаблоны записываются во время модификации РСА в БПШ, что позволяет совместить во времени работу БОШ и БПШ. При этом в регистр БОШ кроме шаблона также помещается его адрес в БПШ. Перед записью шаблона в БОШ анализируется старший разряд поля O . Если в этом разряде ноль, то шаблон записывается по СЗВ в верхнюю очередь коротких операций, растущую от младших номеров регистров к старшим, а если ноль, то используется СЗН, и запись осуществляется в нижнюю очередь длинных операций, растущую от старших номеров регистров к младшим. Чтение из верхней очереди БОШ производится

по СЧВ при поступлении от БКО сигнала $s=1$ готовности блока операций принять новый шаблон. Такой сигнал могут формировать БКО1 и БКО2. При этом сигнал от БКО1 имеет более высокий приоритет. Чтение из нижней очереди производится по СЧН при поступлении сигнала готовности $s=1$ от БДО, который имеет наивысший приоритет. После чтения шаблона из БОШ в блок операций, дополнительный разряд f в регистре БОШ, из которого был считан шаблон, устанавливается в единичное состояние, фиксируя факт передачи шаблона из БОШ в блок операций.

Блоки операций. В БКО1, БКО2 и БДО шаблоны копируются из БОШ вместе с их адресами в БПШ последовательно в соответствии с приоритетами сигналов готовности этих блоков. Копируемый шаблон помещается в РД блока операции. При этом в РУ блока разряд признака h поступления шаблона устанавливается в единичное состояние. Блоки операций могут работать параллельно. При этом в каждом блоке после поступления шаблона выполняются следующие действия:

- в РУ из БНО по коду операции, заданному в шаблоне, считывается число тактов L , необходимых для выполнения операции;
- в поле t РУ каждый такт добавляется по единице, пока не выполнится условие: $L=t$;
- по истечении заданного числа тактов в поле r РУ записывается результат, полученный после выполнения операции (O) над данными ($d1, d2$), указанными в шаблоне. Результат операции вычисляется в БНО в соответствии с заданным кодом операции;
- в РУ разряд признака готовности результата g устанавливается в единичное состояние.

Блок записи результата. По мере получения результатов в блоках операций осуществляется формирование признаков готовности результата $g=1$. Если БЗР свободен, то из блока операции, признак готовности результата которого имеет наивысший приоритет, в РД БЗР записываются следующие данные: адрес выполненного шаблона в БПШ (A), результат (r) и поля назначения ($ar1, ar2$). Наивысший приоритет имеет признак готовности результата БДО, низший – БКО2. После записи данных из блока операций в РД БЗР в единичное состояние устанавливается соответствующий передавшему данные блоку операций признак поступления данных b в РУ БЗР ($b1$ – БКО1, $b2$ – БКО2 и $b3$ – БДО). При единичном значении признака поступления данных b производится запись в БПШ результата операции из БЗР во входные поля операндов шаблонов по адресам, заданным

в полях назначений в БЗР. Запись результата в БПШ осуществляется одновременно по двум адресам. Роль дополнительных регистров адреса БПШ играют разряды полей назначений (ar1 и ar2) в РД БЗР. При этом в процессе записи результата соответствующие разряды тегов Т шаблонов, в которые записывается результат, устанавливаются в единичное состояние. В случае, если после этого $T=(1,1)$, то в соответствующий шаблону разряд РИ БПШ записывается единица. После записи результата в БПШ признак поступления данных b в РУ БЗР принимает нулевое значение, а БЗР переходит в состояние готовности принять следующий результат из блока операций.

Вычисления завершаются, когда по окончании очередного рабочего цикла (РСА=1111) в БПШ отсутствуют готовые к выполнению шаблоны, выполнение которых не завершено (СШ=0000). В этом случае устанавливается признак Ост=1 и формирование тактовых сигналов прекращается.

Экспериментальные исследования

Проведению экспериментальных исследований предшествует изменение параметров Excel, связанных с вычислением формул. Выбираются вычисления в книге «вручную» и включаются итеративные вычисления с предельным числом итераций. При этом задаваемое предельное число итераций (I) определяет один из следующих режимов работы ПР.

1. Полутакт (I=1). После начальной установки всех блоков процессора при первом нажатии клавиши F9 в БПШ, БОШ, БКО1, БКО2, БДО и БЗР первым выполняется полутакт $T=1$, а затем $T=0$. Несовместимые действия, такие как запись или чтение по адресу в счетчике адреса и изменение этого адреса, выполняются в блоках процессора в разных полутактах. В разных полутактах также выполняется чтение и запись в один и тот же блок.

2. Такт (I=2). Каждое нажатие клавиши F9 вызывает выполнение двух полутактов.

3. Шаг (I=16). Для работы ПР в данном режиме необходимо последовательно вводить с клавиатуры номер следующего шага в ячейку Шаг[C4] в десятичной системе счисления и нажимать клавишу F9. Нажатие клавиши F9 вызывает выполнение одного рабочего цикла процессора, в течение которого шаблоны с тегом $T=11$ из БПШ копируются в БОШ. После выполнения одного рабочего цикла ПР формирует сигнал Стоп[C3]=1 и останавливается. При этом в СЦ БПШ отображается число выполненных рабочих циклов. В конце каждого рабочего цикла анализируется содержимое СШ, в котором подсчитывается число шаблонов в БПШ, готовых в текущий момент к выполнению, но выполнение которых еще не завершено. Если СШ=0000, то формируется сигнал Ост[B3]=1 и ПР останавливается. Для выключения пошагового режима необходимо занести 0 в ячейку Шаг[C4].

4. Автомат ($I \geq M$, где M – максимальное число тактов, необходимых для выполнения «программы»). После начальной установки всех блоков процессора при нажатии клавиши F9 полностью выполняется списочная форма, предварительно занесенная в БПШ, формируется сигнал Ост[B3]=1 и ПР останавливается. При этом в СШ отображается код 0000, а СЦ – число выполненных рабочих циклов. Результаты вычислений записываются в свободные регистры шаблонов после списочной формы в первые поля данных.

Пример решения задачи, описываемой информационным графом, изображенным на рис. 1а, представлен на экранной форме (рис. 4). В примере используются следующие исходные данные: $d1=4$; $d2=5$; $d3=49$; $d4=9$; $d5=7$. Операции и результаты вычислений (в десятичной системе счисления) приведены в табл. 2.

Таблица 2

Операции и результаты примера

Операция	Символ	Код	Функция Excel	Формула	Результат
Умножение	x	0011	ПРОИЗВЕД	$r_1=d_1r$	$r_1=4r$
Извлечение квадратного корня	$\sqrt{\quad}$	1010	КОРЕНЬ	$r_2=\sqrt{d_3}$	$r_2=\sqrt{49}=7$
Суммирование	+	0001	СУММ	$r_3=d_4+d_5$	$r_3=10+8=18$
Вычисление остатка от деления	%	0101	ОСТАТ	$r_4=r_1\%r_2$	$r_4=20\%7=6$
Вычитание	-	0010	СУММ	$r_5=r_1-r_3$	$r_5=20-18=2$
Деление нацело	//	0110	ЧАСТНОЕ	$r_6=r_2//r_5$	$r_6=7//2=3$

Анализ работы ПР в режимах полутакт и такт позволяет наблюдать параллельную работу блоков процессора, включая передачу данных между блоками. Можно задавать различные варианты параллельной работы БКО1, БКО2 и БДО, изменяя с помощью БНО время выполнения операций.

Заключение

Разработанная функциональная модель потокового ПР демонстрирует возможность использования технологии построения действующих функциональных моделей учебных вычислительных структур, состоящих из параллельно работающих блоков, с помощью Microsoft Excel. Для обеспечения поддержки проведения экспериментальных исследований функциональная модель ПР содержит дополнительные средства. Можно выделить следующие виды таких средств: управления; ввода данных; анализа функционирования ПР. К средствам управления относятся поля подачи сигналов в БУМ (ВЧ, ЧТ, ЗП, НУ) и поле ввода номера шага рабочего цикла процессора в БПО (Шаг), а также поля для занесения числа тактов, необходимых для выполнения операций (L) и формулы для вычисления результатов (r и rd) в БНО. Средства ввода данных включают поля ввода данных в РСА (AI) и в РЗД (DI) в БПШ, а также ПД БПШ. Для анализа работы процессора при проведении экспериментальных исследований используются отображаемые на экранной форме состояния узлов и блоков процессора, включая СЦ и СШ, разряды РИ, дополнительные разряды регистров шаблонов (w, e) в БПШ, а также (f) в БОШ; содержимое РД и РУ в блоках операций и БЗР. Особую роль при проведении экспериментальных исследований ПР играет БНО, позволяющий менять состав и время выполнения операций. Благодаря БНО можно отказать от реального выполнения операций над данными, ограничившись зада-

нием их условного обозначения и времени выполнения.

Список литературы

1. Дикарев Н.И., Шабанов Б.М., Шмелёв А.С. Преимущества и недостатки использования метода векторов указателей в векторном потоковом процессоре // Программные системы: теория и приложения. 2021. Т. 12, № 4(51). С. 65–83. DOI: 10.25209/2079-3316-2021-12-4-65-83.
2. Баканов В. М. Управление динамикой вычислений в процессорах потоковой архитектуры для различных типов алгоритмов // Программная инженерия. 2015. № 9. С. 20–24.
3. Толмачев А.А., Викторов Д.С., Хапёрский А.А., Дергунов А.М. Способ распределения данных по монофункциональным блокам процессоров вычислительной системы с управлением потоком данных. Патент РФ № 2820032. Патентообладатель Федеральное государственное казённое военное образовательное учреждение высшего образования «Военная академия воздушно-космической обороны имени Маршала Советского Союза Г.К. Жукова» Министерства обороны Российской Федерации. 2024. Бюл. № 16.
4. Орлов С.П., Ефимушкина Н.В. Имитационные модели параллельных вычислительных структур // Международная конференция по мягким вычислениям и измерениям: материалы конференции (г. Санкт-Петербург, 25–27 мая 2016 г.). Санкт-Петербург: Издательство Санкт-Петербургского государственного электротехнического университета, 2016. Т. 1. С. 217–220.
5. Хилько Д.В., Шикун Ю.И., Орлов Г.А., Степченко Ю.А. Программный комплекс моделирования потоковой рекуррентной многоядерной вычислительной системы (ПК ПОТОК). Свидетельство о государственной регистрации программы для ЭВМ. 2022. № 2022667594.
6. Змеев Д.Н. Средства проектирования высокопроизводительных потоковых вычислительных систем // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2016. № 2. С. 159–163.
7. Коннов Н.Н., Пчелинцев А.И., Федулов В.Д., Бычков А.С. Учебный кроссплатформенный эмулятор микропрограммируемой ЭВМ // Информатика: проблемы, методы, технологии: материалы XXII Международной научно-практической конференции им. Э.К. Алгазина (Воронеж, 10–12 февраля 2022 г.) / Под редакцией Д.Н. Борисова. Воронеж: Вэлборн, 2022. С. 1403–1411.
8. Жмакин А.П. Программные модели процессоров и ЭВМ // Auditorium. Электронный научный журнал Курского государственного университета. 2016. № 3 (11). С. 77–84. URL: <https://auditorium.kursksu.ru/magazine/archive/number/64> (дата обращения: 25.03.24).
9. Страбыкин Д.А. Функциональная модель учебного процессора с микропрограммируемой архитектурой в Microsoft Excel // Современные наукоемкие технологии. 2023. № 2. С. 102–120. DOI: 10.17513/snt.39530.