

Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)» (МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчет по лабораторной работе №2 по курсу "Архитектура ЭВМ"

Тема Изучение принципов работы микропроцессорного ядра RISC-V

Студент Шавиш Тарек.

Группа ИУ7и-54Б

ПреподавательПопов А.Ю.

Оглавление

Введение														
1	Основные теоретические сведения	3												
2	Эксперименты	4												
	2.1 Задание 1	4												
	2.2 Задание 2	6												
	2.3 Задание 3	7												
	2.4 Задание 4	7												
	2.5 Задание 5	8												
3a	аключение	12												

Введение

Основной целью данной лабораторной работы является ознакомление с принципами функционирования, построения и особенностями архитектуры суперскалярных конвейерных микропроцессоров.

лью работы является знакомство с принципами проектирования и верификации сложных цифровых устройств с использованием языка описания аппаратуры SystemVerilog и ПЛИС.

1 Основные теоретические сведения

Изучение архитектуры суперскалярных конвейерных микропроцессоров используется синтезируемое описание микропроцессорного ядра Taiga, реализующего систему команд RV32I семейства RISC-V. Данное описание выполнено на языке описания аппаратуры SystemVerilog.

Термин RISC-V является названием для семейства различных систем команд, которые строятся вокруг базового набора команд, путем внесения в него различных расширений. В данной работе исследуется набор команд RV32I, который включает в себя основные команды 32-битной целочисленной арифметики кроме умножения и деления.

2 Эксперименты

2.1 Задание 1

Дизассемблировать программу по индивидуальному варианту.

```
1
            .section .text
 2
            .globl _start;
 3
            len = 8 #Размер массива
 4
            enroll = 4 #Количество обрабатываемых элементов за одну итерацию
 5
            elem sz = 4 #Размер одного элемента массива
7
     _start:
8
           la x1, _x
9
            addi x20, x1, elem_sz*len #Адрес последнего элемента
10
            add x31, x0, x0
11
    lp:
           lw x2, 0(x1)
12
13
           lw x3, 4(x1)
           add x31, x31, x2 #!
15
           add x31, x31, x3
           lw x4, 8(x1)
16
17
           lw x5, 12(x1)
           add x31, x31, x4
18
19
           add x31, x31, x5
20
           addi x1, x1, elem_sz*enroll
           bne x1, x20, lp
21
22
           addi x31, x31, 1
23 lp2: j lp2
24
           .section .data
25
26
    _x:
           .4byte 0x1
27
           .4byte 0x2
28
            .4byte 0x3
29
            .4byte 0x4
30
            .4byte 0x5
31
            .4byte 0x6
32
            .4byte 0x7
            .4byte 0x8
```

Рис. 2.1: Код неоптимизированной программы

Создается массив из 8 элементов. Он последовательно заполняется числами от 1 до 8. Потом все элементы суммируются и в регистр х31 записывается накопленное значение. Затем содержимое регистра х31 инкрементируется.

```
main.elf: file format elf32-littleriscv
 2
 3
     SYMBOL TABLE:
4
    80000000 l d
                     .text
                              00000000
                                          .text
 5
     80000040 1
                 d
                      .data
                              00000000
                                          .data
                     *ABS*
 6
    000000000 l df
                              00000000
                                          main.o
 7
     00000008 1
                     *ABS*
                              00000000
                                          len
 8
     00000004 1
                     *ABS*
                                          enroll
                              00000000
9
                     *ABS*
     00000004 1
                             00000000
                                          elem sz
10
     80000040 1
                      .data
                              00000000
                                          _X
11
     80000010 1
                      .text
                             00000000
                                          1p
12
     8000003c 1
                                          lp2
                      .text
                              00000000
13
     80000000 g
                      .text
                              00000000
                                          _start
14
     80000060 g
                      .data
                              00000000
                                          end
```

Рис. 2.2: Таблица символов

```
17
    Disassembly of section .text:
    80000000 < start>:
18
                           auipc x1,0x0
19
    80000000:
                00000097
                           addi x1,x1,64 # 80000040 <_x>
    80000004:
                04008093
20
                           addi x20,x1,32
21
    80000008:
                02008a13
    8000000c: 00000fb3
                           add x31,x0,x0
22
23
    80000010 <lp>:
24
    80000010:
               0000a103
                           lw x2,0(x1)
                           lw x3,4(x1)
25
    80000014:
                0040a183
26
    80000018: 002f8fb3 add x31,x31,x2
27
    8000001c: 003f8fb3
                           add x31,x31,x3
                          lw x4,8(x1)
28
    80000020: 0080a203
29
                           lw x5,12(x1)
    80000024: 00c0a283
    80000028: 004f8fb3 add x31,x31,x4
30
31
    8000002c: 005f8fb3
                           add x31,x31,x5
32
    80000030: 01008093
                           addi x1,x1,16
33
    80000034:
                           bne x1,x20,80000010 <lp>
                fd409ee3
                           addi x31,x31,1
34
    80000038:
                001f8f93
35
    8000003c <1p2>:
    8000003c: 0000006f jal x0,8000003c <lp2>
36
```

Рис. 2.3: Дизассемблированная секция текста

```
Disassembly of section .data:
39
40
     80000040 < x>:
41
     80000040:
                   0001
                                c.addi x0,0
42
     80000042:
                                unimp
                   0000
43
     80000044:
                  0002
                                0x2
                                unimp
44
     80000046:
                  0000
45
     80000048:
                                1b \times 0,0(x0) # 0 < elem_sz-0x4>
                  00000003
46
     8000004c:
                  0004
                                c.addi4spn x9,x2,0
47
     8000004e:
                                unimp
                  0000
48
     80000050:
                  0005
                                c.addi x0,1
49
     80000052:
                                unimp
                  0000
50
     80000054:
                                0x6
                  0006
51
     80000056:
                  0000
                                unimp
52
     80000058:
                  00000007
                                0x7
                                c.addi4spn x10,x2,0
53
     8000005c:
                   0008
```

Рис. 2.4: Дизассемблированная секция данных

2.2 Задание 2

Получить снимок экрана, содержащий временную диаграмму выполнения стадий выборки и диспетчеризации команды с адресом 80000000 на 2-ой итерации.

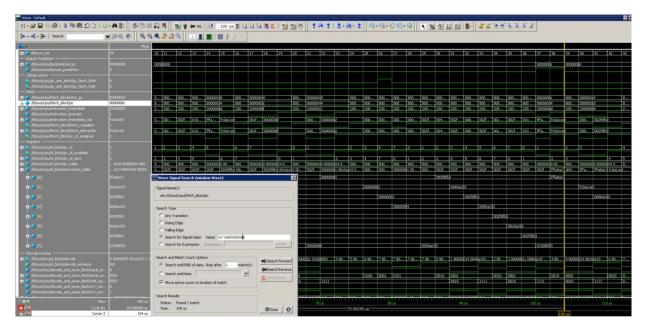


Рис. 2.5: Временная диаграмма выполнения стадий выборки и диспетчеризации команды 8000000с на 2-й итерации

2.3 Задание 3

Получить снимок экражердержащий временную диаграмму выполнения стадии декодирования и планирования на выполнение команды с адресом 80000018 на 2-ой итерации.

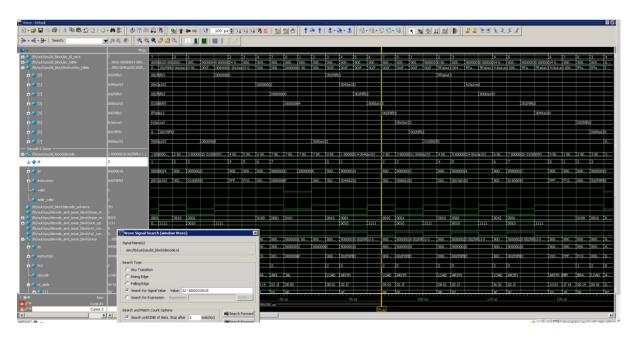


Рис. 2.6:Временная диаграмма выполнения стадий декодирования и планирования на выполнение команды 80000018 на 2-й итерации

2.4 Задание 4

Получить снимок экрана, содержащий временную диаграмму выполнения стадии выполнения команды с адресом 8000002с на 1-ой итерации.

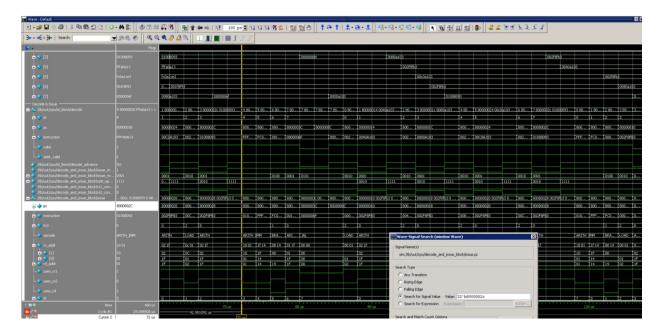


Рис. 2.7:Временная диаграмма выполнения стадии выполнения команды 8000002с на 1-й итерации

2.5 Задание 5

Оптимизировать программу.

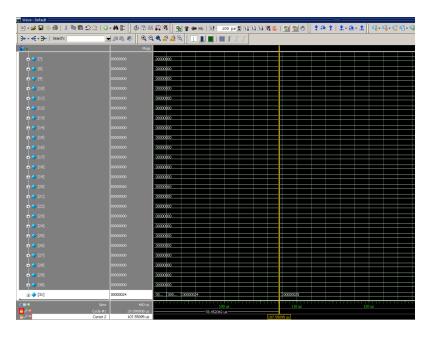


Рис. 2.8: Результат выполнения программы

Значение регистра x31 в конце выполнения программы равно 25h = 37, как и предполагалось ранее.

Адрес	Код команды	Команда	1d	1 2	3 4	5	6 7	7 8	9	10	11	12 1	3 14	15	16	17 1	8 10	20		0Me				27	28 2	9 3	B 3	1 32	33	34	35	36 3	7 38	39	40	41	42	43 4	44	45
80000000<-start>	00000097	auipc x1,0x0		F ID		L			Ť	- V										-	1	1				Ť		1					1		Ť					
8000004	03c08093	addi x1,x1,60#8000003c<_x>	1	F	ID D	AL		Т	Т			Т	Т		П	Т	Т			Т	Т	Т			Т	Т	Т	Т			Т	Т	Т				П	Т	П	П
8000008	00200a13	addi x20,x1,32	2		FI	DD	AL																																	
8000000c <lp></lp>	0000a103	lw x2,0(x1)	3		F	ID	D M	11 M	2 M3				Т		П		Т				Т	Т				Т	Т	Т				Т	Т				П			
8000010	0040a183	lw x3,4(x1)	4			F	ID () M	1 M2	МЗ																														
8000014	002f8fb3	add x31,x31,x2	5				FI	D D	C	AL		Т	Т		П	Т	Т			Т	Т	Т			Т	Т	Т	Т			Т	Т	Т				П	Т		П
80000018	003f8fb3	add x31,x31,x3	6				. 1		D W																															
800001c	0080a203	lw x4,8(x1)	7					F	ID	W	D	M1 N	2 M3													Т		Т					Т							
8000020	00с0а283	lw x5,12(x1)	Θ						F	ID	W	DN	1 M2	МЗ																										
80000024	004f8fb3	add x31,x31,x4	1							F	ID	W	0 0	AL																										
8000028	005f8fb3	add x31,x31,x5	2								F :	ID I	W W	D	AL											1														
8000002c	01008093	addi x1,x1,16	3									F I	D W	W	D	٩L																								
80000030	fd409ee3	bne x1,x20,8000000c <lp></lp>	4										ID	W	W	DI	В																							
8000034	001f8f93	addi x31,x31,1	5										F				D X																							
80000038	0000006f	jal x0,80000038 <lp2></lp2>	6											F	ID	W	N DX	d																						
800003c	00000001	<invalid operation=""></invalid>	7												F :	ID I	W X																							
8000040	00000002	<invalid operation=""></invalid>	Θ													FI	DX																							
8000044	00000003	<invalid operation=""></invalid>	1														FX																							
8000048	00000004	<invalid operation=""></invalid>	2														F)																							
8000000c <lp></lp>	0000a103	lw x2,0(x1)	6					Т	Т				Т		П	П	Т	F	ID	D M	1 M2	2 M3				Т	Т	Т					Т				П	П		
8000010	0040a183	lw x3,4(x1)	7																F	ID D) M	M2	M3			1														
8000014	002f8fb3	add x31,x31,x2	Θ																	FI	D D	C	Αl																	
8000018	003f8fb3	add x31,x31,x3	1																	F	I	W	D	AL		1														
800001c	0080a203	lw x4,8(x1)	2																		F	ID	W	D	M1 M	12 M	3													
8000020	θθcθa283	lw x5,12(x1)	3																			F	ID	W	DM	11 M	2 M	3												
8000024	004f8fb3	add x31,x31,x4	4																				F		W		C A													
8000028	005f8fb3	add x31,x31,x5	5																					F	ID V	W	V D	AL.												
8000002c	01008093	addi x1,x1,16	6																						FI															
8000030	fd409ee3	bne x1,x20,8000000c <lp></lp>	7																							FI	D W	W	D	В										
8000000c <lp></lp>	0000a103	lw x2,0(x1)	Θ																							1	E IE	W	W	D	Х									
8999919	0040a183	lw x3,4(x1)	1																							н	F	ID	W	WE	XC									
8999914	002f8fb3	add x31,x31,x2	2																									F	ID	W	X									
8000018	003f8fb3	add x31,x31,x3	3																							1			F	ID										
8000001c	0080a203	lw x4,8(x1)	4																											F	Х									
8000020	θθcθa283	lw x5,12(x1)	5																							1					FX									
8000034	001f8f93	add x31,x31,1	1																													FI								
8000038	0000006f	jal x0,80000038 <lp2></lp2>	2																							1						F	ID	D	В					
800003c	00000001	<invalid operation=""></invalid>	3																							1							F	ID	D					
8000040	00000002	<invalid operation=""></invalid>	4																															F	ID	DX				
8000044	00000003	<invalid operation=""></invalid>	5																																	X				
8000048	00000004	<invalid operation=""></invalid>	6																																	FX				
80000038<1p2>	0000006f	jal x0,8000003c <forever></forever>	4																																		F			
80000038 <lp2></lp2>	0000006f	jal x0,8000003c <forever></forever>	5	1 2																																		F		

Рис. 2.9: Трасса работы неоптимизированной программы

Конфликты возникают, когда после выполнения инструкции lw выполняется инструкция addтимизируем программеренеся все команды lw выше инструкций add.

Рис. 2.10: Код оптимизированной программы

Адрес	Код команды	Команда	id		0			-	-			1.0		10					7 7		Hon					los	0.0	0.7	0.0	0.0	20	2.1	20	201	2.4	ne l	201	27/2	0010		10	
80000000< start>	00000097	auipc x1,0x0		1 F 1				6	7	8	9	10	11	12	13	14	15]]	6 1	/ 18	8 19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	54	35	36	37 3	38	1914	10	+1
800000004_start>	03c08093	addi x1,x1,60#8000003c<_x>	1			DD															Н																			٠		٩
80000004	00200a13	addi x20,x1,32	7			FI																																		4		
800000008 80000000c <lp></lp>	00200a13	lw x2,0(x1)	2				I			MO	MO										Н				Н												7	7	-	۲	7	7
8000000C <tp></tp>	0000a103	lw x3,4(x1)	4								M2	Ma																												d		
80000010	0040a103	lw x4,8(x1)	5		-			100	450	\$100	M1	200	MO				-				Н				Н			Н								-	7	-		۲	7	7
80000014	θθοθα283	lw x5,12(x1)	6					-	-	-	D			Ma																										d		
8000001s	00C0a263 002f8fb3	add x31,x31,x2	7								ID			rıs							Н				Н											-	7	-	-	۲	7	7
80000010	002181b3	add x31,x31,x3	Θ							-			D	ΔI																										d		
80000024	003181B3 004f8fb3	add x31,x31,x3 add x31,x31,x4	1						н		F			D							Н				Н											-	-			۲	7	9
80000024	004101b3 005f8fb3	add x31,x31,x4 add x31,x31,x5	2									Г	_	ID	_	A 1																								d		
80000026 8000002c	01008093	addi x1,x1,16	3													D A					Н				Н												-			۲	7	7
80000020	fd409ee3	bne x1,x20,8000000c <lp></lp>	_													ID		D			Н																			d		
80000030	001f8f93	addi x31,x31,1	5												-	200	-	D X	,																							
80000034	00000006f	jal x0,80000038 <lp2></lp2>	6															D X			н																			4		
8000003c	00000001		_															FX			Н				Н															٠	7	9
8000003C 80000040		<invalid operation=""></invalid>	7 0															E			н												_							4		
	00000002	<pre><invalid operation=""></invalid></pre>	_															F.		ID			140																	٠	٠	9
8000000c <lp></lp>	0000a103	lw x2,0(x1)	6			-	-		н										F															-		-		-	-	4		
80000010	0040a183	lw x3,4(x1)	7					н	н								-			F	ID	ID															-	-		٠	٠	4
80000014	0080a203	lw x4,8(x1)	0			-											-				F						M3												-	4		
80000018	00c0a283	lw x5,12(x1)	1																		Н	F		ID	400	1000	1000													٠		4
8000001c 80000020	002f8fb3 003f8fb3	add x31,x31,x2	2				-														Н		F			AL D			-				_	-			-	-	-	4		
		add x31,x31,x3	3					н													Н			F	455	195	1000									-	-	-		٠	٠	4
80000024	004f8fb3	add x31,x31,x4	4		-	-	-		н								-			-	н		L		F			AL.						-		-	-	-	-	4	4	
80000028	005f8fb3	add x31,x31,x5	5		-			н													Н				Н	F	-	-	-	-										٠		4
8000002c	01008093	addi x1,x1,16	6		-	-	-	-							_	-	-		-	-	Н		L		н	-	F	ID	ID				_	_		-	4	-	-	4	4	
80000030	fd409ee3	bne x1,x20,8000000c <lp></lp>	7		-			н	н												Н			Н	н	н		F	100		-			-	-	-	-	-	-	4	۰	4
8000000c <lp></lp>	0000a103	lw x2,0(x1)	Θ		-	-	-										-		-	-	Н		L	-	Н	-		-	F		D		_	-		-	-	-	-	4	4	
80000010	0040a183	lw x3,4(x1)	1		-				н								-				Н				Н	-				F	ID						-	-		٠	٠	4
80000014	0080a203	lw x4,8(x1)	2		-	-	-	-									-		-	-	н		L		н	-	н	н	Н			X	_			-	-	-	-	4	4	
80000018	00c0a283	lw x5,12(x1)	3		-			н								-	-				Н				н	н						FX					-			4	۹	4
80000034	001f8f93	add x31,x31,1	1		4	-	-	-			ш	_				_	4		-	4	н		L		н			ш	ш					ID				4	4	4	4	_
8000038	0000006f	jal x0,80000038 <lp2></lp2>	2		-			н									-				Н				Н									F :		-				4	۰	4
8000003c	00000001	<invalid operation=""></invalid>	3		4	-	-	-								_	_		-	-	н		L		н	-		_					_			ID			4	4	4	
8000040	00000002	<invalid operation=""></invalid>	4		4												-				Н				н											F	-	200	-	4	۹	4
80000044	00000003	<invalid operation=""></invalid>	5																		L				L			-	L								F			4		
80000048	00000004	<invalid operation=""></invalid>	6																																			FX				
80000038 <lp2></lp2>	0000006f	jal x0,8000003c <forever></forever>	4				1																																F]	_	_	_
80000038 <lp2></lp2>	0000006f	jal x0,8000003c <forever></forever>	5		4				Ļ	L											1			_																		D
Адрес	Код команды	Команда	id	1	2	3 4	5	6	7	8	9	10	11	12	13	14	15]]	6 1	7 18		Hon					25	26	27	28	29	30	31	32	33 .	34	35	36	37 3	38 3	19	10	41

Рис. 2.11: Трасса работы оптимизированной программы

Благодаря оптимизации программы получилось избавиться от конфликтов.

Заключение

В ходе лабораторной работы были изучены принципы функционирования и построения, а также особенности архитектуры суперскалярных конвейерных микропроцессоров на примере микропроцессорного ядра Taiga, реализующего систему команд семейства RTS (СИМ) образомцель данной работы была достигнута.