

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)»

ьный исследовательский университет (МГТУ им. Н. Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»	
КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»	

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

по курсу «Архитектура ЭВМ»

на тему:

«Изучение принципов работы микропроцессорного ядра RISC-V»

Студент Рунов К.А.
Группа <u>ИУ7-54Б</u>
Вариант 18
Преподаватели Попов А.Ю., Ибрагимов С.В.

СОДЕРЖАНИЕ

1	Цел	ІИ																		3	
2	Зад	ания																		3	
	2.1	Задание 1																		3	
	2.2	Задание 2							•	•										8	
	2.3	Задание 3																		10	
	2.4	Задание 4							•	•										11	
	2.5	Задание 5																	•	12	
3	Зак	лючение																		20	

1 Цели

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

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

2 Задания

Далее представлен ход выполнения заданий лабораторной работы.

2.1 Задание 1

Далее представлены листинги, полученные в процессе выполнения первого задания лабораторной работы.

```
.section .text
  .globl _start;
 2
            len = 9
            enroll = 2
 4
           elem sz = 4
 5
 6
   _{\mathrm{start}}:
8
            la x1, _x
           addi x20, x1, elem_sz*len
9
10
           lw x31, 0(x1)
11
            addi x1, x1, elem sz*1
12 lp:
13
           lw x2, 0(x1)
           lw x3, 4(x1)
14
15
           bltu x2, x31, lt1
           add x31, x0, x2
16
17 lt1:
            bltu x3, x31, lt2
           add x31, x0, x3
18
19 lt 2:
           add x1, x1, elem_sz*enroll
20
21
           bne x1, x20, lp
22|lp2: j lp2
23
24 section data
25
            .4byte 0x1
  x:
            .4byte 0x2
26
            .4byte 0x3
27
28
            .4byte 0x4
29
            .4byte 0x8
30
            .4byte 0x6
31
            .4byte 0x7
32
            .4byte 0x5
33
            .4byte 0x4
```

Листинг 1 – Код программы по варианту на языке ассемблера

```
1 Disassembly of section .text:
 2 \mid 800000000 < start >:
 3 80000000:
                        00000097
                                       auipc
                                                  x1,0x0
 4 80000004:
                        03808093
                                       addi
                                                  x1, x1, 56 \# 80000038 < x >
5 80000008:
                        02408\,\mathrm{a}13
                                       addi
                                                  x20, x1, 36
 6 8000000c:
                        0000\,\mathrm{af83}
                                       lw
                                                  x31,0(x1)
 7 80000010:
                        00408093
                                       addi
                                                  x1, x1, 4
8 \mid 80000014 < lp >:
9 80000014:
                                                  x2,0(x1)
                        0000a103
                                       lw
10 80000018:
                        0040\,\mathrm{a}183
                                       lw
                                                  x3,4(x1)
11 8000001 c:
                                                  x2,x31,80000024 < lt1>
                       01f16463
                                       bltu
12 80000020:
                        00200 fb3
                                       add
                                                  x31, x0, x2
13 \mid 80000024 < \text{lt } 1 >:
14 80000024:
                                       bltu
                        01 \, \text{f} \, 1 \, \text{e} \, 463
                                                  x3,x31,8000002c < lt2 >
15 80000028:
                        00300\,{\rm fb}3
                                       add
                                                  x31, x0, x3
16 \mid 8000002 \, \mathrm{c} \mid \mathrm{clt} \, 2 > :
17 8000002 c:
                                       addi
                        00808093
                                                  x1, x1, 8
                                                  x1,x20,80000014 < lp>
18 80000030:
                        ff4092e3
                                       bne
19 \mid 80000034 < lp2 >:
20 80000034:
                        0000006 \, \mathrm{f}
                                                  x0,80000034 < lp2 >
                                       jal
21
22 Disassembly of section .data:
23 \mid 80000038 < x>:
24 80000038:
                        0001
                                       c.addi
                                                  x0,0
                        0000
25 8000003a:
                                       unimp
                                       0x2
26 8000003 c:
                        0002
27 8000003e:
                        0000
                                       unimp
                                       lb
                                                  x0,0(x0) # 0 < enroll -0x2>
28 80000040:
                        00000003
29 80000044:
                                       c.addi4spn
                                                            x9, x2, 0
                        0004
30 80000046:
                        0000
                                       unimp
                                       c.addi4spn
31 80000048:
                                                            x10, x2, 0
                        0008
32 8000004a:
                        0000
                                       unimp
33 8000004 c:
                        0006
                                       0x6
34 8000004e:
                        0000
                                       unimp
35 80000050:
                        00000007
                                       0x7
36 80000054:
                        0005
                                       c.addi
                                                 x0,1
37 80000056:
                                       unimp
                        0000
38 80000058:
                                       c.addi4spn
                        0004
                                                            x9, x2, 0
```

Листинг 2 – Дизассемблированный листинг кода программы по варианту

```
1 00000097
 2 03808093
 3 02408a13
 4 0000 af 83
5 00408093
 6 0000 a 103
 7 0040a183
8 01f16463
9 00200 fb3
10 01 f1 e 4 6 3
11 00300 fb3
12 00808093
13 ff4092e3
14 0000006 f
15 00000001
16 00000002
17 00000003
18 00000004
19 | 00000008
20 00000006
21 00000007
22 | 00000005
23 00000004
```

Листинг 3 — Код программы по варианту в шестнадцатеричном представлении

```
1 #define len 9
2 #define enroll 2
3 #define elem sz 4
5 | unsigned x[] = \{ 1, 2, 3, 4, 8, 6, 7, 5, 4 \};
6
  void start() {
      unsigned *x1 = _x; // la x1, _x
       unsigned *x20 = x1 + len; // addi x20, x1, elem_sz*len
9
       unsigned x31 = x1[0]; // lw x31, 0(x1)
10
      x1 += 1; // addi x1, x1, elem sz*1
11
12
13
      do {
14
           // lp
15
           unsigned x^2 = x^1[0]; // lw x^2, 0(x^1)
16
           unsigned x3 = x1[1]; // lw x3, 4(x1)
17
           if (x2 < x31) { // bltu x2, x31, lt1
18
19
           } else {
               x31 = x2; // add x31, x0, x2
20
21
           // lt1
22
           if (x3 < x31) { // bltu x3, x31, lt2
23
24
           } else {
25
               x31 = x3; // add x31, x0, x3
26
           }
           // lt2
27
           x1 += enroll; // add x1, x1, elem sz*enroll
28
       \} while (x1 != x20); // bne x1, x20, lp
29
30
31
       for (;;); // lp2: j lp2
32 }
```

Листинг 4 – Псевдокод программы по варианту

В результате выполнения программы, в регистре х31 будет содержаться число, соответствующее максимальному элементу массива х $_$, то есть 8.

2.2 Задание 2

```
1 Disassembly of section .text:
 2 \mid 800000000 < _start >:
 3 80000000:
                      00200\,\mathrm{a}13
                                    addi
                                              x20, x0, 2
 4 80000004:
                      00000097
                                              x1,0x0
                                    auipc
 5 80000008:
                                    addi
                                              x1, x1, 60 \# 80000040 < x >
                      03c08093
 6 \mid 8000000c < loop >:
 7 8000000c:
                                    lw
                      0000a103
                                              x^{2},0(x^{1})
8 80000010:
                                    add
                                              x31,x31,x2
                      002f8fb3
9 80000014:
                      0040a103
                                    lw
                                              x2,4(x1)
10 80000018:
                      002f8fb3
                                    add
                                              x31, x31, x2
11 8000001 c:
                      0080\,\mathrm{a}103
                                    lw
                                              x2,8(x1)
12 80000020:
                                              x31, x31, x2
                      002f8fb3
                                    add
13 80000024:
                                    lw
                                              x2,12(x1)
                      00 c0 a 103
14 80000028:
                                              x31, x31, x2
                      002 f8 fb3
                                    add
                                              x1, x1, 16
15 8000002 c:
                                    addi
                      01008093
16 80000030:
                      fffa0a13
                                    addi
                                              x20, x20, -1
17 80000034:
                      fc0a1ce3
                                    bne
                                              x20,x0,8000000c < loop >
18 80000038:
                      001f8f93
                                    addi
                                              x31,x31,1
19|8000003c < for ever >:
20 8000003 c:
                      0000006 f
                                    jal
                                              x0,8000003c <forever>
21
22 Disassembly of section .data:
23 \mid 80000040 < x>:
24 80000040:
                      0001
                                    c.addi
                                              x0,0
25 80000042:
                      0000
                                    unimp
26 80000044:
                      0002
                                    0x2
                      0000
27 80000046:
                                    unimp
28 80000048:
                      0000003
                                    1b
                                              x0,0(x0) \# 0 < elem sz-0x4>
29 8000004c:
                      0004
                                    c.addi4spn
                                                       x9, x2, 0
30 8000004e:
                      0000
                                    unimp
31 80000050:
                      0005
                                    c.addi
                                             x0,1
32 80000052:
                      0000
                                    unimp
33 80000054:
                      0006
                                    0x6
34 80000056:
                      0000
                                    unimp
35 80000058:
                      00000007
                                    0x7
36 8000005 c:
                      0008
                                    c.addi4spn
                                                       x10, x2, 0
```

Листинг 5 – Дизассемблированный листинг кода тестовой программы test.s

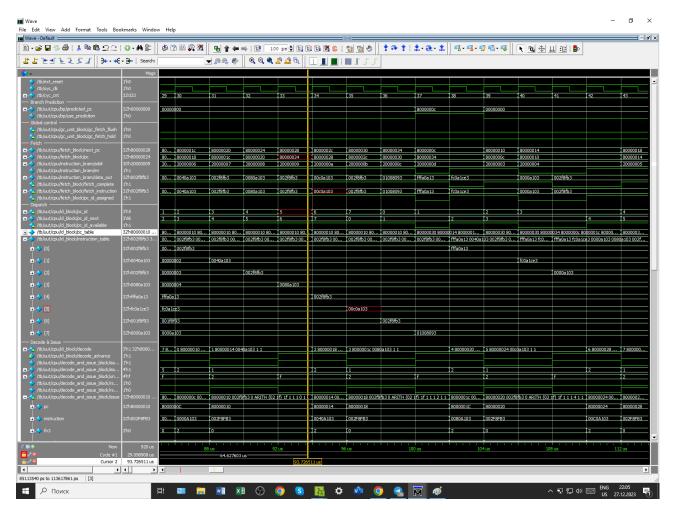


Рисунок 1 — Временная диаграмма выполнения стадий выборки и диспечеризации команды по адресу 80000024, 2-я итерация

2.3 Задание 3

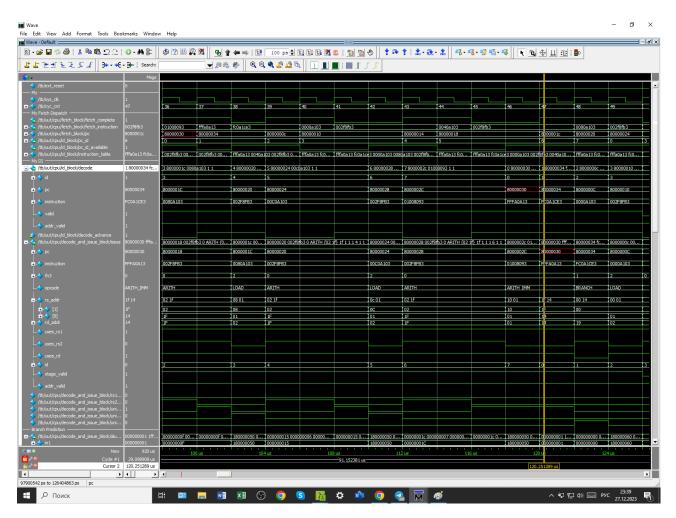


Рисунок 2 — Временная диаграмма выполнения стадии декодирования и планирования на выполнение команды по адресу 80000030, 2-я итерация

2.4 Задание 4

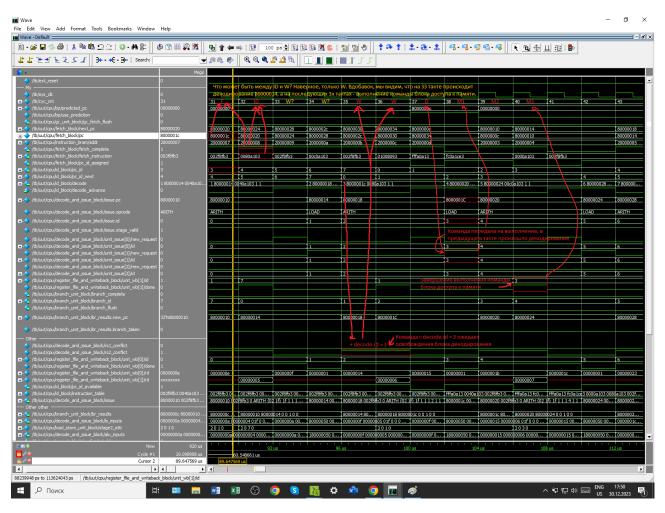


Рисунок 3 — Временная диаграмма выполнения стадии выполнения команды по адресу 8000001с, 2-я итерация

2.5 Задание 5

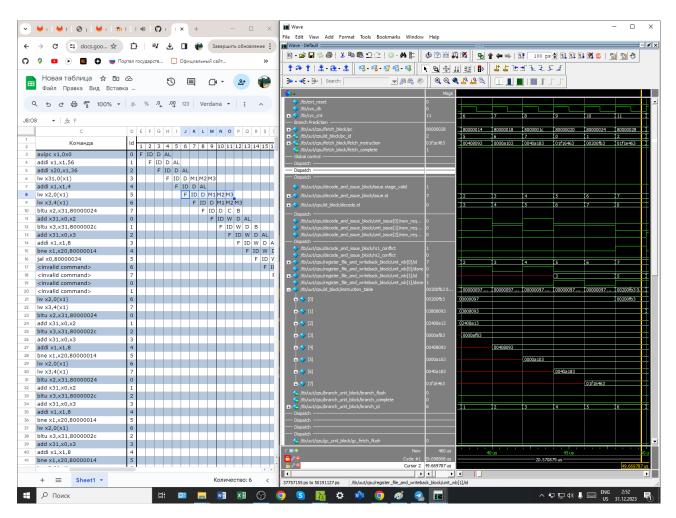


Рисунок 4 — Временная диаграмма всех стадий выполнения команды, обозначенной в тексте программы #!

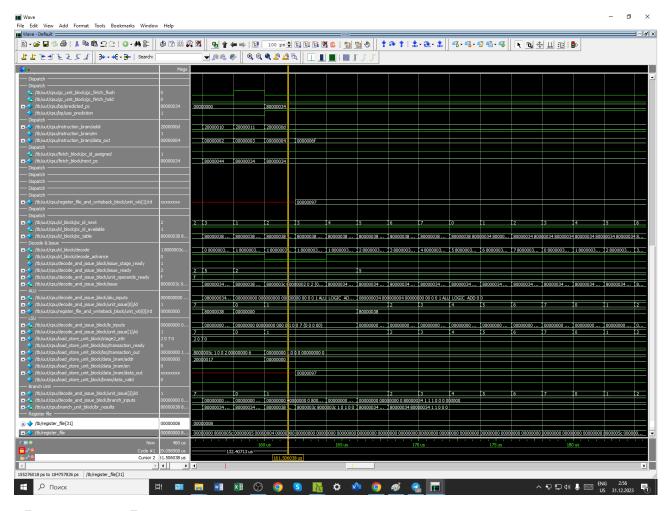


Рисунок 5 — В результате выполнения программы, регистр х
31 принимает значение 8, как и ожидалось

Адрес	Код	Команда	id	ᆫ			_	_	_					_	_										_				р та		_		_													_		_	_		_		_	_		_	_
	команды		_					6	7	8	9 1	0 11	12	13	14 1	5 16	17	18	19 2	0 2:	1 22	2 23	3 24	25	26	27	28 2	9 30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55 !	56 5	57 5	8 5	<u> 9</u>
30000000<_start>	00000097	auipc x1,0x0	0) Al																																																			
30000004	03808093	addi x1,x1,56	1			D D					_				_								_																														_	_	_	_	
30000008	02408a13	addi x20,x1,36	2			: IC																																																			
3000000с	0000af83	lw x31,0(x1)	3			F				M2N	13																					Ш	Ш										Ш													_	_
30000010	00408093	addi x1,x1,4	4				F	ID																																																	
30000014 <lp></lp>	0000a103	lw x2,0(x1)	5					F		D M																						Ш	Ш										Ш													_	_
30000018	0040a183	lw x3,4(x1)	6							ID I																																															
3000001c	01f16463	bltu x2,x31,80000024	7							FI) C																																												\perp	
30000020	00200fb3	add x31,x0,x2	0								FI	O W																																													
30000024	01f1e463	bltu x3,x31,8000002c	1								- 1	ID	W	D	В																																									\perp	
30000028	00300fb3	add x31,x0,x3	2									F	ID																																												
3000002c	00808093	addi x1,x1,8	3	ш) AL																																									-
30000030	ff4092e3	bne x1,x20,80000014	4											F 1	D V	V D	В																																								
30000034	0000006f	jal x0,80000034	5		Т	Т	Т		П	Т	Т	Т		\neg	F II	D W	D	Х		Т	Т	Т	Т	П	П		Т	Т	Т													П											\neg	П	Т	Т	П
80000038	00000001	<invalid command=""></invalid>	6												F	ID	W	Х																																							
3000003c	00000002	<invalid command=""></invalid>	7			Т			П		Т					F	ID	Х			Т	Т	Т		П																												\neg	П	Т	Т	П
80000040	00000003	<invalid command=""></invalid>	0														F	х																																							
80000044	00000004	<invalid command=""></invalid>	1			Т			П		Т							FX			Т	Т	Т		П																												\neg	П	Т	Т	
0000014 <lp></lp>	0000a103	lw x2.0(x1)	6																FII	D D	M:	1 M2	2 M	3																																	
0000018	0040a183	lw x3,4(x1)	7		т	т	т				т	т			т			П		= 10								Т	Т																								7	7	т	т	
000001c	01f16463	bltu x2.x31.80000024	0																	F	IC	D	C	В																																	
0000020	00200fb3	add x31.x0.x2	1		т	т	Т				т	т			т			П		т	F	ID	o w	D	AL			Т	Т																								7	7	т	т	ı
0000024	01f1e463	bltu x3.x31.8000002c	2																		Ė			W		В																															
0000028	00300fb3	add x31.x0.x3	3																				F	ID	w	D	AI .																										7	7		т	Ŧ
000002c	00808093	addi x1,x1,8	4																				Ĺ				D A	1																													
80000030	ff4092e3	bne x1.x20.80000014	5																								N C																										_			т	
80000030 80000014 <lp></lp>	0000a103	lw x2,0(x1)	6																										M1	M2	мз																										
80000014 (1)2	0040a183	lw x3.4(x1)	7																								F II		D																											7	
3000001c	01f16463	bltu x2.x31.80000024	0																										W																												
80000020	00200fb3	add x31,x0,x2	1																								-		ID				v																							4	٠
80000024	00200IB3 01f1e463	bltu x3.x31.8000002c	2		-	-																						-				W										Н														-	
80000024	00300fb3	add x31,x0,x3	3																													W																								4	
80000028 8000002c	00808093	addi x1.x1.8	4		-						-											-	٠							r		ID																					-			-	-
8000002c	ff4092e3	bne x1,x20,80000014	5							-																						F																								4	
0000030 0000014 <lp></lp>	0000a103	lw x2,0(x1)	6		-	-	-				-	_		_	-	-			_		-	-	-	-									FX									-											_		-	-	-
																																	FX																							4	
0000024	01f1e463	bltu x3,x31,8000002c	2		-	-					-				-								-											F		D																	_			_	-
0000028	00300fb3	add x31,x0,x3	3																																F	ID																				4	
000002c	00808093	addi x1,x1,8	4		_	_	_				-			_	_	_			_	_	_	_	_	_				_				ш					ID					_								_		_	_	_	_	_	
0000030	ff4092e3	bne x1,x20,80000014	5																																		F	Х																		4	
0000014 <lp></lp>	0000a103	lw x2,0(x1)	6		_	_					_			_	_						_	_	_									ш	ш					FX					ш										_	_	_	_	
000002c	00808093	addi x1,x1,8	4																																				F			AL															
0000030	ff4092e3	bne x1,x20,80000014	5					Ш			_		ш	_	_																									F		D											_	_	_	_	
0000014 <lp></lp>	0000a103	lw x2,0(x1)	6																																						F	ID															
0000018	0040a183	lw x3,4(x1)	7	Ш			1	Ш					Ш																													F				М2										\perp	
8000001c	01f116463	bltu x2,x31,80000024	0																																											С											
0000020	00200fb3	add x31,x0,x2	1																																									F	ID	D	AL	Х									
0000024	01f1e463	bltu x3,x31,8000002c	2																																											ID		х									
Адрес	Код	Команда	Т.,	1	2 :	3 4	5	6	7	8	9 11	0 11	112	13	14 1	5 16	17	118	1912	0 2	1 22	2 23	3 24	125	26	27	8 2	9 3	131	132	33	34	35	36	137	138	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	551	56	7 7	8 5	;9

Рисунок 6 – Трасса выполнения неоптимизированной программы

Конфликты возникают из-за того, что происходит обращение к регистру, запись в который ещё не была завершена. Как можно заметить по псевдокоду, строка x1 += enroll (add x1, x1, $elem_sz^*enroll$) будет выполнена вне зависимости от предшествующих условных переходов, а также значение регистра x1 не используется после строки lw x3, 4(x1) в пределах выполнения тела цикла. Следовательно, если поместить строку add x1, x1, $slem_sz^*enroll$ сразу после lw x3, 4(x1), это не повлияет на результат выполнения программы, зато даст дополнительную задержку в один такт между загрузкой значения в x2 и чтением из него, что позволит устранить конфликты.

```
.section .text
 1
 2
            .globl _start;
 3
            len = 9
            enroll = 2
 4
            elem sz = 4
 5
 6
 7
   _start:
8
            la x1, _x
9
            addi x20, x1, elem_sz*len
10
            lw x31, 0(x1)
            addi x1, x1, elem sz*1
11
12 lp:
13
            lw x2, 0(x1) #!
            lw x3, 4(x1)
14
            add x1, x1, elem_sz*enroll
15
            bltu x2, x31, lt1
16
            add x31, x0, x2
17
            bltu x3, x31, lt2
18 lt1:
19
            add x31, x0, x3
20 lt 2:
            bne x1, x20, lp
22| lp2: j lp2
23
24
            .section .data
25
            .4byte 0x1
  x:
26
            .4byte 0x2
27
            .4byte 0x3
28
            .4byte 0x4
29
            .4byte 0x8
30
            .4byte 0x6
31
            .4 \, \text{byte} \, 0 \text{x} 7
32
            .4byte 0x5
33
            .4byte 0x4
```

Листинг 6 – Оптимизированный код программы по варианту на языке ассемблера

```
1 Disassembly of section .text:
 2 \mid 800000000 < start >:
 3 80000000:
                       00000097
                                      auipc
                                                x1,0x0
 4 80000004:
                       03808093
                                      addi
                                                 x1, x1, 56 \# 80000038 < x >
 5 80000008:
                                      addi
                       02408\,\mathrm{a}13
                                                x20, x1, 36
 6 8000000c:
                                      lw
                                                 x31,0(x1)
                       0000 \, af83
 7 80000010:
                       00408093
                                      addi
                                                 x1, x1, 4
 8 \mid 80000014 < lp >:
 9 80000014:
                       0000\,\mathrm{a}103
                                      lw
                                                 x2,0(x1)
10 80000018:
                                                 x3,4(x1)
                       0040\,\mathrm{a}183
                                      lw
11 8000001 c:
                       00808093
                                      addi
                                                x1,x1,8
12 80000020:
                                      bltu
                                                 x2, x31, 80000028 < lt1 >
                       01f16463
13 80000024:
                       00200fb3
                                      add
                                                 x31, x0, x2
14 \mid 80000028 < lt1 >:
15 80000028:
                       01 \, \text{f} \, 1 \, \text{e} \, 463
                                      bltu
                                                x3,x31,80000030 < lt2 >
16 8000002 c :
                       00300fb3
                                      add
                                                x31, x0, x3
17 \mid 80000030 < \text{lt } 2 >:
18 80000030:
                       ff4092e3
                                      bne
                                                 x1,x20,80000014 < lp>
19 \mid 80000034 < lp2 >:
20 80000034:
                       0000006 f
                                                x0,80000034 < lp2 >
                                      jal
21
22 Disassembly of section .data:
23 \mid 80000038 < x>:
24 80000038:
                       0001
                                      c.addi
                                                x0,0
25 8000003a:
                       0000
                                      unimp
26 8000003 c:
                       0002
                                      0x2
27 8000003 e:
                       0000
                                      unimp
28 80000040:
                       0000003
                                      lb
                                                 x0,0(x0) \# 0 < enroll -0x2>
29 80000044:
                       0004
                                      c.addi4spn
                                                          x9, x2, 0
30 80000046:
                       0000
                                      unimp
                                      c.addi4spn
31 80000048:
                       0008
                                                           x10, x2, 0
32 8000004a:
                                      unimp
                       0000
33 8000004c:
                       0006
                                      0x6
34 8000004 e:
                       0000
                                      unimp
35 80000050:
                       00000007
                                      0x7
                                      c.addi
36 80000054:
                       0005
                                                x0,1
37 80000056:
                       0000
                                      unimp
38 80000058:
                       0004
                                      c.addi4spn
                                                           x9, x2, 0
```

Листинг 7 — Дизассемблированный листинг оптимизированного кода программы по варианту

```
1 00000097
 2 | 03808093
 3 02408a13
 4 0000 af 83
 5 00408093
 6 0000 a 103
 7 0040a183
8 00808093
9 01 f 1 6 4 6 3
10 00200 fb3
11 01 f1 e 4 6 3
12 00300 fb3
13 ff4092e3
14 0000006 f
15 00000001
16 00000002
17 00000003
18 00000004
19 00000008
20 00000006
21 00000007
22 | 00000005
23 00000004
```

Листинг 8 — Оптимизированный код программы по варианту в шестнадцатеричном представлении

```
1 #define len 9
2 #define enroll 2
3 #define elem sz 4
5 | unsigned _x[] = \{ 1, 2, 3, 4, 8, 6, 7, 5, 4 \};
6
  void _start() {
      unsigned *x1 = _x; // la x1, _x
       unsigned *x20 = x1 + len; // addi x20, x1, elem_sz*len
9
       unsigned x31 = x1[0]; // lw x31, 0(x1)
10
      x1 += 1; // addi x1, x1, elem sz*1
11
12
13
      do {
14
           // lp
15
           unsigned x^2 = x^1[0]; // lw x^2, 0(x^1)
16
           unsigned x3 = x1[1]; // lw x3, 4(x1)
17
18
           x1 += enroll; // add x1, x1, elem sz*enroll
19
           if (x2 < x31) { // bltu x2, x31, lt1
20
21
           } else {
               x31 = x2; // add x31, x0, x2
22
23
           // lt1
24
25
           if (x3 < x31) { // bltu x3, x31, lt2
26
           } else {
27
               x31 = x3; // add x31, x0, x3
28
           // lt 2
29
30
       \} while (x1 != x20); // bne x1, x20, lp
31
32
       for (;;); // lp2: j lp2
33 }
```

Листинг 9 – Псевдокод оптимизированной программы по варианту

Адрес	Код команды	Команда	id	1 2 3	TATE		7 0	101	10 11	112	12 1	4 1 1 5	116	17 10	10	2012	11 22	122	2415	25 24	127			ер та		22	24 2	E 24	127	201	201	10 4	142	42	44 4	5146	147	100	4015	ole:	1 5 2	IE2	EA	Icc.	56	2710		olen
80000000< start> (00000097	auipc x1.0x0		F ID D		, 0	/ 0	9 .	10 11	12	13 1	7 1.	10	17 10	12	20 2	1 22	23	24 2	20 20	121	20	25 .	30 3.	1 32	22	J4 J	3 30))/	50	39 .	10 4.	42	40	17 7	J 40	7 47	40	77.	0 3.	1 32	. 33	54	33	50 .	77 3	0 3:	100
	03808093	addi x1,x1,56	1	F ID					-																																					-	-	
	02408a13	addi x1,x1,36	2		ID D													Н																													-	-
	0000af83	lw x31,0(x1)	3				M1 M2	2 42																																						4	4	
	00408093	addi x1.x1.4	4			F ID			-									н									-	-							-	-									_	-	-	-
					1																																									4	4	
	0000a103 0040a183	lw x2,0(x1)	5			F			M2 M3 M1 M2		_	_				_	_	н		_	-				-		_	-	-						_	-	-			-	-	_		\vdash	_	_	-	-
		lw x3,4(x1)	7																																											4	4	
	00808093	addi x1,x1,8					F		D AL		_	_	ш			_		_	ш	_	_					ш	_	_	_						_	_	_			_	_	_		\perp	_	_	_	_
	01f16463	bltu x2,x31,80000028	0						ID D																																					4	4	
	00200fb3	add x31,x0,x2	1			\perp		\perp	F ID				ш					ш	ш	_							_	_					\perp		_					_				\perp	_	_	_	_
	01f1e463	bltu x3,x31,80000030	2						F	ID																																						
	00300fb3	add x31,x0,x3	3			\perp		\perp			ID [ш							Ш	_	_							_					_				ш	Ш	_	_	_
	ff4092e3	bne x1,x20,80000014	4								F II																																					
	0000006f	jal x0,80000034	5					\perp	\perp		F		D	X												ш				Ш									-1					ш	Ш	_		1
	00000001	<invalid operand=""></invalid>	6									F	ID	X																																		
	00000002	<invalid operand=""></invalid>	7											Х																															Ш		\perp	
	00000003	<invalid operand=""></invalid>	0											FX																																		
	0000a103	lw x2,0(x1)	6									T		F			11 M2							T			T	T				T			Т	T			1	T					┙	\perp		
80000018	0040a183	lw x3,4(x1)	7												F	ID [D M1	M2	МЗ																													
8000001c	00808093	addi x1,x1,8	0					П	Т			Т	П		П	F I	D D	AL			Т				Т	П	Т	Т	Т	П				П		Т	Т	П	Т	Т	Т				П	Т	Т	Т
80000020	01f16463	bltu x2,x31,80000028	1														F ID	D	В																													
80000024	00200fb3	add x31,x0,x2	2					П	т	П			П		П		F	ID	D A	AL.	т								т																П	т	т	
80000028	01f1e463	bltu x3,x31,80000030	3															F	ID I	D B																												
8000002c	00300fb3	add x31.x0.x3	4					П	т	П			П		П			П	FΙ	D D	AL								т																П	т	т	
80000030 f	ff4092e3	bne x1,x20,80000014	5																	F IC	D	В																										
80000014 <lp></lp>	0000a103	lw x2,0(x1)	6					П	_	П			П							F	ID	D	M1N	12 M	3																				\neg	т	_	
	0040a183	lw x3.4(x1)	7																					11 M																								
8000001c	00808093	addi x1,x1,8	0					П	_	П			П									F	ID	D AI																					\neg	т	_	
	01f16463	bltu x2,x31,80000028	1																					D D																							die.	
	00200fb3	add x31.x0.x2	2						_																D	Х		_																	\neg	_	_	
	01f1e463	bltu x3.x31.80000030	3																						ID																						die.	
	00300fb3	add x31,x0,x3	4						_															-	F	Х		_																	\neg	_	_	
	ff4092e3	bne x1.x20.80000014	5																							FX																					ale:	
	01f1e463	bltu x3,x31,80000030	3						-																		E 1	D D	В																	-	-	
	00300fb3	add x31.x0.x3	4																									F IC		v																	ale:	
	ff4092e3	bne x1,x20,80000014	5						-																		- "	F																		-	-	
	0000a103	lw x2,0(x1)	6																									ď	F																			
	0040a183	Iw x3,4(x1)	7						-																					FX																-	-	
	ff4092e3	IW X3,4(X1) bne x1.x20.80000014	5																												-	D D	0														-	-
	0000a103	Iw x2,0(x1)	6																														D	м1	42 14	2										4	4	
	0040a103	IW x2,0(x1) Iw x3.4(x1)	7																													1 110	ID														-	-
	0040a183 00808093		0																								7					F					2									4	4	
		addi x1,x1,8	1						-									-															F		DA	L B						-		ш			-	-
	01f16463	bltu x2,x31,80000028																																												4	4	
	00200fb3	add x31,x0,x2	2					\perp	_																		-									D D			-	-				ш	_	4	4	1
	01f1e463	bltu x3,x31,80000030	3						4																										F	ID										4	4	
	00300fb3	add x31,x0,x3	4					\perp	_																											F								ш		4	_	\perp
80000030 f	ff4092e3	bne x1,x20,80000014	5	4 0 -					101	10				17 (20 -		100	24		10-	20	20	20 5	1 2-	22	24-	-	10-		20	10 0		-		-	FX		10	0.5		-	-				1	1
Адрес	Код команды	Команда	id	1 2 3	4 5	5 6	/ 8	19]	10 11	112	13 1	4 15	16	1/ 18	119	20 2	1122	123	24 2	25 26	s 27			30 3: ер та		33	34 3	5 36	37	38	39	10 4	142	43 4	94 4	5 46	147	48	49 5	U [5]	1 52	153	154	55	56	2/ 5	8 55	1160

Рисунок 7 – Трасса выполнения оптимизированной программы

3 Заключение

В результате данной работы были изучены принципы функционирования, построения и особенности архитектуры суперскалярных конвейерных микропроцессоров.

На основе изученных материалов был найден способ оптимизировать программу.

Цель работы была достигнута.