

# 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»

# Лабораторная работа №2 по дисциплине «Архитектура ЭВМ»

**Тема** Изучение принципов работы микропроцессорного ядра RISC-V

Студент Тузов Даниил Александрович

**Группа** <u>ИУ7-52Б</u>

Преподаватель Калитвенец Максим, Попов А.Ю.

## Введение

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.

Для достижения поставленных целей в настоящей лабораторной работе используется синтезируемое описание микропроцессорного ядра Taiga, реализующего систему команд RV32I семейства RISC-V. Данное описание выполнено на языке описания аппаратуры SystemVerilog.

В ходе лабораторной работы используется средство моделирования MentorGrap Modelsim для моделирования работы исследуемого микропроцессора в процессе выполнения программы и наблюдения формы внутренних сигналов.

#### 2 Задание 1

#### 2.1Исходный текст исследуемой программы

Листинг 1 – Исходный текст исследуемой программы

```
.section .text
        .globl _start;
        len = 9
        enroll = 2
        elem_sz = 4
_start:
        la x1, _x
        addi x20, x1, elem_sz*len
        1w x31, 0(x1)
        addi x1, x1, elem_sz*1
1p:
        1w x2, 0(x1)
                         #!
        1w x3, 4(x1)
        bltu x2, x31, lt1
        add x31, x0, x2
lt1:
        bltu x3, x31, lt2
        add x31, x0, x3
                                         1
```

#### 2.2 Дизассемблированный листинг

Листинг 2 – Дизассемблированный листинг программы

```
Disassembly of section .text:
80000000 <_start>:
80000000: 00000097
                              auipc x1,0x0
80000004: 03808093
                              addi x1,x1,56 # 80000038 <_x>
80000008: 02408a13
                               addi x20,x1,36
8000000c: 0000af83
                              lw x31,0(x1)
80000010: 00408093
                              addi x1,x1,4
80000014 <lp>:
80000014: 0000a103
                              lw x2,0(x1)
80000018: 0040a183
                              lw x3, 4(x1)
8000001c: 00808093
                               addi x1,x1,8
80000020: 01f16463
                               bltu x2,x31,80000028 <1t1>
80000024: 00200fb3
                               add x31,x0,x2
80000028 <lt1>:
80000028: 01f1e463
                              bltu x3,x31,80000030 <1t2>
8000002c: 00300fb3
                               add x31,x0,x3
80000030 <1t2>:
80000030: ff4092e3
                               bne x1, x20, 80000014 < lp >
80000034 <1p2>:
80000034: 0000006f
                               jal x0,80000034 < lp2 >
Disassembly of section .data:
80000038 <_x>:
80000038: 0001
                                .insn 2, 0x0001
8000003a: 0000
                                .insn 2, 0x
8000003c: 0002
                                 .insn 2, 0x0002
```

```
8000003e:
          0000
                                 .insn 2, 0x
80000040:
          0000003
                               1b x0,0(x0) # 0 < enroll - 0x2 >
80000044:
          0004
                                 .insn 2, 0x0004
80000046:
         0000
                                 .insn 2, 0x
80000048:
          8000
                                 .insn 2, 0x0008
                                 .insn 2, 0x
8000004a:
          0000
8000004c: 0006
                                 .insn 2, 0x0006
8000004e:
          0000
                                 .insn 2, 0x
80000050:
                               .insn 4, 0x0007
          00000007
80000054:
          0005
                                 .insn 2, 0x0005
80000056:
           0000
                                 .insn 2, 0x
80000058:
                                 .insn 2, 0x0004
          0004
```

#### 2.3 Псевдокод, поясняющий работу программы

#### Листинг 3 – Псевдокод

```
#define len 9
#define enroll 2
#define elem_sz 4
unsigned _x[] = \{ 1, 2, 3, 4, 8, 6, 7, 5, 4 \};
void _start() {
    unsigned *x1 = _x; // la x1, _x
    unsigned *x20 = x1 + len; // addi x20, x1, elem_sz*len
    unsigned x31 = x1[0]; // lw x31, 0(x1)
    x1 += 1; // addi x1, x1, elem_sz*1
    do {
        // lp
        unsigned x2 = x1[0]; // lw x2, 0(x1)
        unsigned x3 = x1[1]; // lw x3, 4(x1)
        if (x2 < x31) { // bltu x2, x31, lt1
        else
            x31 = x2; // add x31, x0, x2
        // lt1
        if (x3 < x31) // bltu x3, x31, lt2
        else
            x31 = x3; // add x31, x0, x3
        // lt2
        x1 += enroll; // add x1, x1, elem_sz*enroll
    } while (x1 != x20); // bne x1, x20, lp
    for (;;); // lp2: j lp2
}
```

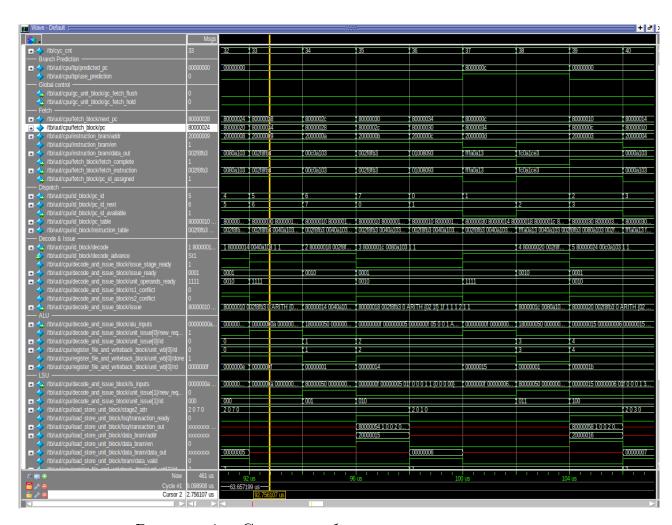


Рисунок 1 – Стадии выборки и диспетчеризации

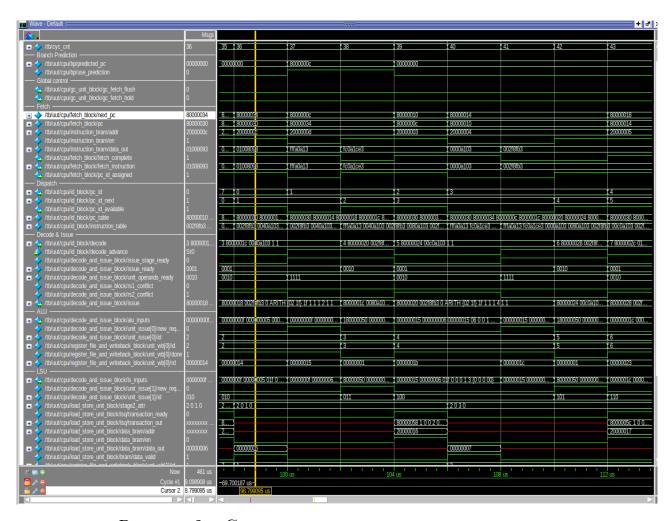


Рисунок 2 – Стадии декодирования и планирования

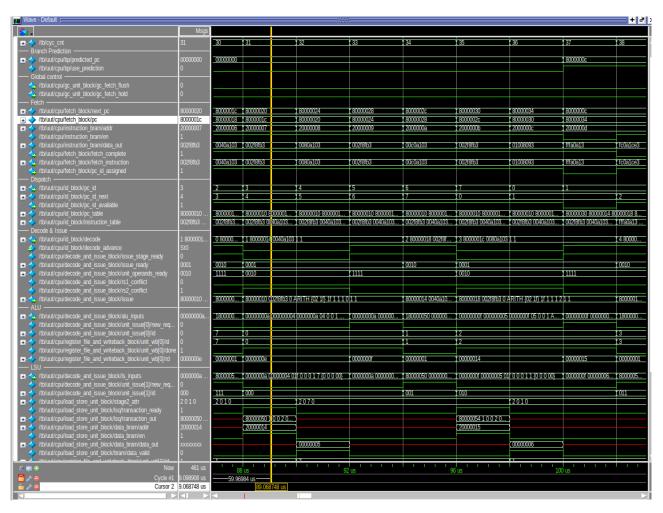


Рисунок 3 – Непосредственное выполнение комманды

## 6.1 Место, отмеченное решеткой

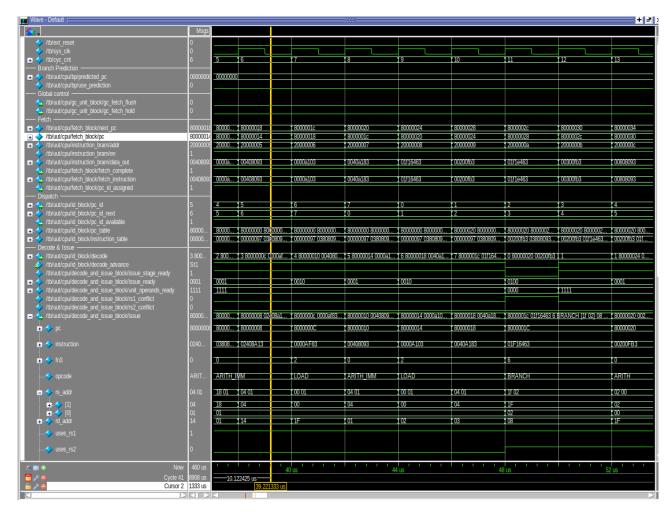


Рисунок 4 — Первая итерация команды, отмеченной #

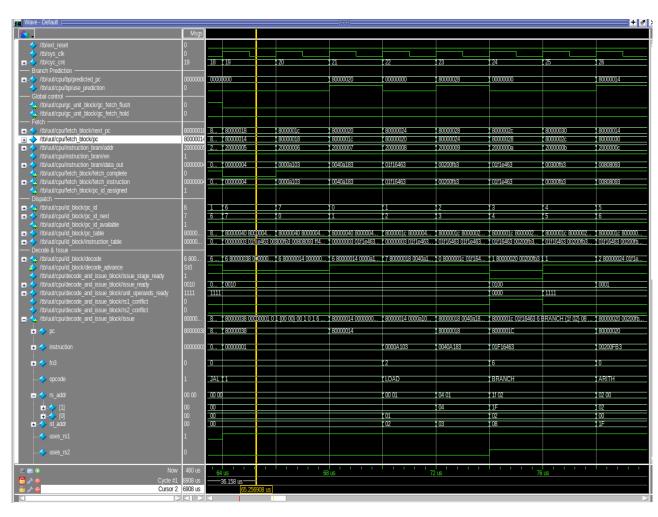


Рисунок 5 — Вторая итерация команды, отмеченной #

#### 6.2 Трасса исходной программы

	Код		$\overline{}$			_					_			_	_		_							_	_	_		Ном	on	такт	ra						_	_						_		_									_	
Адрес	команды	Команда	id	1 2	3	4	5 6	7	8	9	10	11	12	13	14	15 1	6 1	7 18	3 19	20	21	22	23 2	4 2	5 2	5 2						3 34	1 35	36	37	38	39	40	41	42	13 4	4 4	46	47	48	49	50 5	1 5	2 5	3 5	4 5	5 5	5	7 5	3 59	6
	00000097	auipc x1,0x0	0	F ID														Т								Т					Т																		Т	Т				П	П	Γ
8000004	03808093	addi x1,x1,56	1	F	ID	D .	٩L				l																																													ı
8000008	02408a13	addi x20,x1,36	2		F	ID	D A	L																																																ı
800000c	0000af83	lw x31,0(x1)	3		П	F	ID [	) M:	1 M2	М3				П				Т	П				П			Т	П				Т										П	Т							Т	Т	Т		Т	Т	Т	Г
80000010	00408093	addi x1,x1,4	4				FI	D D	AL																																															ı
80000014 <lp></lp>	0000a103	lw x2,θ(x1)	5		П		F	11	D D	M1	M2	МЗ		П	Т	Т	Т	Т	Т				Т	Т	Т	Т	Т		П	Т	Т	Т	Г	П	Т			П			Т	Т	Т					Т	Т	Т	Т	Т	Т	Т	Т	Г
80000018	0040a183	lw x3,4(x1)	6					F	II	D	M1	M2	МЗ																																					ı				ı		t
8000001c	01f16463	bltu x2,x31,80000024	7		П	$\neg$	Т	Т	F	ID	D	С	В	Т	Т	т	т	Т	Т				т	Т	Т	т	т		П	т	Т	Т	Т	Т	т				П	$\neg$	т	Т	Т	П			Т	т	т	Т	Т	Т	т	Т	т	Г
80000020	00200fb3	add x31,x0,x2	Θ							F	ID	W	D	AL																																				ı				ı		t
80000024	01fle463	bltu x3,x31,8000002c	1		П				т	П	F	ID	W	D	в	т	т	т	т				т	т	т	т	т				Т	т	Т	Т	т				$\neg$	╛	т	т		П				т	т	Т	Т	т	Т	Т	т	Г
80000028	00300fb3		2									F	ID	W	D A	AI .																																							ė.	L
8000002c <lt2></lt2>	00808093	addi x1,x1,8	3			_			т							D A			т				т			т	т					т		т	т						т								т	т	т		т	т	т	Т
80000030	ff4092e3	bne x1.x20.80000014	4													W																																						۰	÷	ŀ
80000034	0000006f	jal x0,80000034	5															x																							-									т			т	т	т	t
80000038			6															/ X																																٠				۰	٠	ŀ
8000003c	00000001	invalid command	7															D D					7			П															7								Т	Т	T		T	Т	1	f
80000030		invalid command	0															: X																																٠	٠			٠	٠	Ł
80000040	00000003	invalid command	1															F)																																۰			۳	Н	-	۰
80000044 80000014 <lp></lp>	00000004 0000a103	lw x2.0(x1)	6		ш					Н				-	-		-	F/	F	TO	-	M1 I		-							-										-	-							-	٠	+			۰	٠	Ł
			0																F																															۰				н	-	۰
80000018 8000001c	0040a183	lw x3,4(x1) bltu x2.x31.80000024	/		ш				-	-				-			-	-	-	F		D I					-				-	-									-	+							-	۰	+		۰	۰	+	Ł
			0																		F																													۰				н	-	۰
80000020	00200fb3	add x31,x0,x2	11		ш				_				_	_	_		_	_						N E							_	_								_	_	_							_	1	_		_	_	_	L
80000024		bltu x3,x31,8000002c	2																					D V																										۰			ı	ı		L
80000028	00300fb3		3		Ш								_	_	_			_									AL				_									_	_	_						_	_	_	_		_	_	┸	L
8000002c <lt2></lt2>		addi x1,x1,8	4																					F			D																							۰				н	۰	L
80000030	ff4092e3	bne x1,x20,80000014	5		Ш									_											F		W		В												_	_							_	1	1			1	Т	L
80000014 <lp></lp>	0000a103		6																							F	ID			11 M																										L
80000018	0040a183	lw x3,4(x1)	7		Ш																						F			D M																									┸	L
		bltu x2,x31,80000024	Θ																									F		W D																								ı		L
80000020	00200fb3	add x31,x0,x2	1																											ID W																									L	L
80000024	01fle463	bltu x3,x31,8000002c	2		Ш																									F II	D W	W	DX	(																н			Н	н	ш	ı
80000028	00300fb3	add x31,x0,x3	3		Ш						l																			F	11	) W	X																							ı
8000002c <lt2></lt2>	00808093	addi x1,x1,8	4																												F	II	X																							ı
80000030	ff4092e3	bne x1,x20,80000014	5		П				Т					П				Т					П			Т	П				Т	F	Х		Т						П	Т							Т	Т	Т		Т	Т	Т	Г
80000014 <lp></lp>	0000a103	lw x2,θ(x1)	6						L										L							L	L					L	FX	(																ı				ı	P	ı
80000024	01fle463	bltu x3,x31,8000002c	2	Т	П	П	Т	Т		П			П	Т	T	Т	Т	Г					Т	Т	Т	Г	Г		П	Т				F	ID	D	В	Г		Т	Т	Т		Г			Т	Т	Т	Τ	Τ	Т	Г	Г	Г	
80000028	00300fb3	add x31,x0,x3	3																																F	ID	D	Х												ı				ı		1
8000002c <lt2></lt2>	00808093	addi x1,x1,8	4						Т						Т								Т	Т	Т	Т	Т					Т	Г	П	Т	F	ΙD	DX				Т		Г						Т	Т	Т	Т	Т	Т	Г
80000030	ff4092e3	bne x1,x20,80000014	5																																		F	Х												İ	İ			i	t e	١
80000014 <lp></lp>	0000a103		6						Т						Т										Т	Т	Т					Т		Т	Т			FX				Т								Т	Т	Т	Т	Т	Т	Г
80000014 -tp=			4																																			i	F	ID	D A	L								ı				ı	b	L
80000030	ff4092e3	bne x1,x20,80000014	5																				7		Т	П															ED I									Т	Т	Т	Т	Т	т	Г
80000014 <lp></lp>			6																																								M1	M2	мз					ı				ı	b	L
80000014	0040a183	lw x3.4(x1)	7																																								D			мз				Т	т		Т	Т	т	f
80000016 8000001c	01f16463	bltu x2,x31,80000024	0																																							_	ID	-	_					ŀ	ı		i	ŀ	b	t
80000010	00200fb3	add x31,x0,x2	1														Т						7			П															7	1	_	_	W		х		Т	Т	T		T	Т	1	1
80000020	01fle463	bltu x3,x31,8000002c	2																																								1		ID		DX				H				٠	£
	0111e4b3 Код			1 2	-	4	E 6	1 7		9	10	77	12	12	4	E 3	6 1	7 10	110	20	21	22	22 2	4 2	E 3	6 3	7 20	20	20	31 3	2 2	2 2 4	1 25	26	27	20	20	40	41	42	12 4	4 4	46					2 5	2 5	2 5	4 5	E E	c	7 5	-	1
Апрес	код команды	Команда	id	1   2	3	4	) (	)   /	8	9	10	11	12	13.	.49	L) L	υĮ	/   18	119	120	21	44	23 2	4 2	3   Z	υ   Z .				11   3. Takt		3 34	133	130	1   3 /	158	29	40	41	4Z	10 4	4 4.	140	14/	48	49	JU :	1 5	4 3	2 2	4 5	اداد	כוט	/  5	1126	10

Рисунок 6 – Трасса выполнения неоптимизированной программы

Конфликты возникают из-за того, что происходит обращение к регистру, запись в который ещё не была завершена (регистр x2).

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.

### 6.3 Исходной текст оптимизированной программы

Листинг 4 – Исходный текст оптимизированной программы

```
.section .text
        .globl _start;
        len = 9
        enroll = 2
        elem_sz = 4
_start:
        la x1, _x
        addi x20, x1, elem_sz*len
        lw x31, 0(x1)
        addi x1, x1, elem_sz*1
lp:
        lw x2, 0(x1) #!
        lw x3, 4(x1)
        add x1, x1, elem_sz*enroll
        bltu x2, x31, lt1
        add x31, x0, x2
        bltu x3, x31, lt2
lt1:
        add x31, x0, x3
lt2:
        bne x1, x20, lp
1p2: j 1p2
       .section .data
        .4byte 0x1
_x:
        .4byte 0x2
        .4byte 0x3
        .4byte 0x4
        .4byte 0x8
        .4byte 0x6
        .4byte 0x7
        .4byte 0x5
        .4byte 0x4
```

## 6.4 Дизассемблированный листинг оптимизированной программы

Листинг 5 – Дизассемблированный листинг оптимизированной программы

```
Disassembly of section .text:
80000000 <_start>:
80000000: 00000097
                              auipc x1,0x0
80000004: 03808093
                              addi x1,x1,56 # 80000038 <_x>
80000008: 02408a13
                              addi x20,x1,36
8000000c: 0000af83
                              lw x31,0(x1)
80000010: 00408093
                              addi x1,x1,4
80000014 <lp>:
80000014: 0000a103
                              lw x2,0(x1)
                              lw x3,4(x1)
80000018: 0040a183
8000001c: 01f16463
                              bltu x2,x31,80000024 <1t1>
80000020: 00200fb3
                              add x31,x0,x2
80000024 <lt1>:
80000024: 01f1e463
                              bltu x3,x31,8000002c <1t2>
80000028: 00300fb3
                              add x31,x0,x3
8000002c <1t2>:
8000002c: 00808093
                              addi x1,x1,8
                              bne x1,x20,80000014 <1p>
80000030: ff4092e3
80000034 <1p2>:
                              jal x0,80000034 <1p2>
80000034: 0000006f
Disassembly of section .data:
80000038 <_x>:
80000038: 0001
                                .insn 2, 0x0001
8000003a: 0000
                                .insn 2, 0x
8000003c: 0002
                               .insn 2, 0x0002
8000003e: 0000
                               .insn 2, 0x
80000040: 00000003
                             1b x0,0(x0) # 0 < enroll - 0x2 >
80000044: 0004
                               .insn 2, 0x0004
80000046: 0000
                               .insn 2, 0x
80000048: 0008
                                .insn 2, 0x0008
8000004a: 0000
                               .insn 2, 0x
8000004c: 0006
                                .insn 2, 0x0006
8000004e: 0000
                                .insn 2, 0x
                             .insn 4,0x0007
80000050: 00000007
80000054:
          0005
                                .insn 2, 0x0005
80000056:
          0000
                                .insn 2, 0x
80000058: 0004
                                .insn 2, 0x0004
```

#### 6.5 Трасса оптимизированной программы

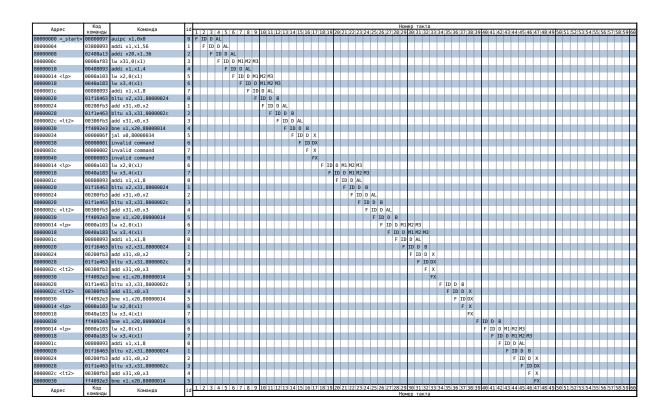


Рисунок 7 – Трасса выполнения оптимизированной программы

Можно заметить, что конфликтов больше нет, а трасса стала на два такта короче (на 4%).

## 7 Вывод

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

На основе изученных материалов удалось оптимизировать программу.