

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

**Московский институт электроники и математики
им. А.Н.Тихонова НИУ ВШЭ**

Департамент компьютерной инженерии

Курс «Системное проектирование цифровых устройств»

**ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

Тема работы: «Разработка и программирование Soft-процессорных ядер с
архитектурой однокластерный MIPS. Часть 1»

Выполнили:

Студенты группы БИВ174
Бригада №5

Подчерзцев Алексей Евгеньевич
Солодянкин Андрей Александрович

Принял:

асс. МИЭМ НИУ ВШЭ
Американов А.А.

Москва 2020 г.

Содержание

1	Задание	3
2	Выполнение работы	3
2.1	Моделирование счетчика	3
2.2	Моделирование последовательности Фибоначчи	4
2.3	Моделирование извлечения квадратного корня	6
3	Самостоятельная работа	9
4	Выводы по работе	9
	Список использованных источников	11

1 Задание

1. Изучить разделы 6.2, 6.3, 6.4, 6.7 и приложение В книги Н&Н. Добавить в микропроцессор в соответствии со своим вариантом поддержку следующих команд: j, xori, sllv, nor.
2. Разработать программу, продемонстрировать на модели и прототипе правильность их работы. Разработать в соответствии со своим вариантом программу, продемонстрировать на модели и прототипе правильность ее работы. Добавить ее в проект микропроцессора; добавить в папку с программой файл описания. Использовать только те команды, которые есть в процессоре.

6) Найти сумму геометрической прогрессии (количество членов прогрессии = ваш вариант % 30 + 3, знаменатель прогрессии = ваш вариант % 10 + 1)
3. Перейти в ветку проекта schoolMIPS 01_mmio. Скачать новую версию процессора и выполнить на вашей плате (или DE10-Lite) программы 00_counter, 01_fibonacci, 02_sqrt. Убедиться, что они работают также. Выполнить одну из программ (по вариантам):

2) 04_gpio

2 Выполнение работы

2.1 Моделирование счетчика

Ассемблерный код счетчика представлен на листинге ниже.

```
.text  
  
start:      move    $v0, $0  
counter:    addiu   $v0, $v0, 1  
            beqz    $0, counter
```

Моделирование программы проводилось в среде Icarus Verilog.

Ниже приведена часть логов из выполнения программы:

```
0 pc = x pcaddr = xxxxxxxX instr = 00001021 v0 = 0 addu $2, $0, $0  
1 pc = x pcaddr = xxxxxxxX instr = 00001021 v0 = 0 addu $2, $0, $0
```

2	pc = 0	pcaddr = 00000000	instr = 00001021	v0 = 0	addu \$2, \$0, \$0
3	pc = 0	pcaddr = 00000000	instr = 00001021	v0 = 0	addu \$2, \$0, \$0
4	pc = 0	pcaddr = 00000000	instr = 00001021	v0 = 0	addu \$2, \$0, \$0
5	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 0	addiu \$2, \$2, 1
6	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 1	beq \$0, \$0, -1
7	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 1	addiu \$2, \$2, 1
8	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 2	beq \$0, \$0, -1
9	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 2	addiu \$2, \$2, 1
10	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 3	beq \$0, \$0, -1
11	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 3	addiu \$2, \$2, 1
12	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 4	beq \$0, \$0, -1
13	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 4	addiu \$2, \$2, 1
14	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 5	beq \$0, \$0, -1
15	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 5	addiu \$2, \$2, 1
16	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 6	beq \$0, \$0, -1
17	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 6	addiu \$2, \$2, 1
18	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 7	beq \$0, \$0, -1
19	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 7	addiu \$2, \$2, 1
20	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 8	beq \$0, \$0, -1
21	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 8	addiu \$2, \$2, 1
22	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 9	beq \$0, \$0, -1
23	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 9	addiu \$2, \$2, 1
24	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 10	beq \$0, \$0, -1
25	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 10	addiu \$2, \$2, 1
26	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 11	beq \$0, \$0, -1
27	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 11	addiu \$2, \$2, 1
28	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 12	beq \$0, \$0, -1
29	pc = 0	pcaddr = 00000000	instr = 24420001	v0 = 12	addiu \$2, \$2, 1
30	pc = 0	pcaddr = 00000000	instr = 1000ffff	v0 = 13	beq \$0, \$0, -1

Вейвформа при моделировании программы (рис. 1).

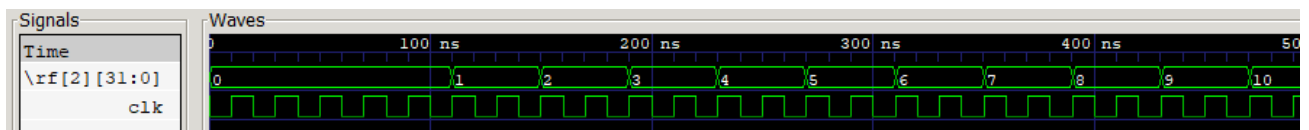


Рис. 1: Вейвформа для программы счетчика

2.2 Моделирование последовательности Фибоначчи

Ассемблерный код функции для подсчета значений последовательности Фибоначчи представлен на листинге ниже.

```
.text
```

```

start:      move    $t0, $0
            li      $t1, 1
            move    $v0, $t1

fibonacci:  addu    $t0, $t0, $t1
            move    $v0, $t0
            addu    $t1, $t0, $t1
            move    $v0, $t1
            beqz    $0, fibonacci

```

Моделирование программы проводилось в среде Icarus Verilog.

Ниже приведена часть логов из выполнения программы:

0	pc = x	pcaddr = xxxxxxxX	instr = 00004021	v0 = 0	addu \$8, \$0, \$0
1	pc = x	pcaddr = xxxxxxxX	instr = 00004021	v0 = 0	addu \$8, \$0, \$0
2	pc = 0	pcaddr = 00000000	instr = 00004021	v0 = 0	addu \$8, \$0, \$0
3	pc = 0	pcaddr = 00000000	instr = 00004021	v0 = 0	addu \$8, \$0, \$0
4	pc = 0	pcaddr = 00000000	instr = 00004021	v0 = 0	addu \$8, \$0, \$0
5	pc = 0	pcaddr = 00000000	instr = 24090001	v0 = 0	addiu \$9, \$0, 1
6	pc = 0	pcaddr = 00000000	instr = 00091021	v0 = 0	addu \$2, \$0, \$9
7	pc = 0	pcaddr = 00000000	instr = 01094021	v0 = 1	addu \$8, \$8, \$9
8	pc = 0	pcaddr = 00000000	instr = 00081021	v0 = 1	addu \$2, \$0, \$8
9	pc = 0	pcaddr = 00000000	instr = 01094821	v0 = 1	addu \$9, \$8, \$9
10	pc = 0	pcaddr = 00000000	instr = 00091021	v0 = 1	addu \$2, \$0, \$9
11	pc = 0	pcaddr = 00000000	instr = 1000ffffb	v0 = 2	beq \$0, \$0, -4
12	pc = 0	pcaddr = 00000000	instr = 01094021	v0 = 2	addu \$8, \$8, \$9
13	pc = 0	pcaddr = 00000000	instr = 00081021	v0 = 2	addu \$2, \$0, \$8
14	pc = 0	pcaddr = 00000000	instr = 01094821	v0 = 3	addu \$9, \$8, \$9
15	pc = 0	pcaddr = 00000000	instr = 00091021	v0 = 3	addu \$2, \$0, \$9
16	pc = 0	pcaddr = 00000000	instr = 1000ffffb	v0 = 5	beq \$0, \$0, -4
17	pc = 0	pcaddr = 00000000	instr = 01094021	v0 = 5	addu \$8, \$8, \$9
18	pc = 0	pcaddr = 00000000	instr = 00081021	v0 = 5	addu \$2, \$0, \$8
19	pc = 0	pcaddr = 00000000	instr = 01094821	v0 = 8	addu \$9, \$8, \$9
20	pc = 0	pcaddr = 00000000	instr = 00091021	v0 = 8	addu \$2, \$0, \$9
21	pc = 0	pcaddr = 00000000	instr = 1000ffffb	v0 = 13	beq \$0, \$0, -4
22	pc = 0	pcaddr = 00000000	instr = 01094021	v0 = 13	addu \$8, \$8, \$9
23	pc = 0	pcaddr = 00000000	instr = 00081021	v0 = 13	addu \$2, \$0, \$8
24	pc = 0	pcaddr = 00000000	instr = 01094821	v0 = 21	addu \$9, \$8, \$9
25	pc = 0	pcaddr = 00000000	instr = 00091021	v0 = 21	addu \$2, \$0, \$9
26	pc = 0	pcaddr = 00000000	instr = 1000ffffb	v0 = 34	beq \$0, \$0, -4
27	pc = 0	pcaddr = 00000000	instr = 01094021	v0 = 34	addu \$8, \$8, \$9
28	pc = 0	pcaddr = 00000000	instr = 00081021	v0 = 34	addu \$2, \$0, \$8
29	pc = 0	pcaddr = 00000000	instr = 01094821	v0 = 55	addu \$9, \$8, \$9
30	pc = 0	pcaddr = 00000000	instr = 00091021	v0 = 55	addu \$2, \$0, \$9

Вейвформа при моделировании программы (рис. 2).

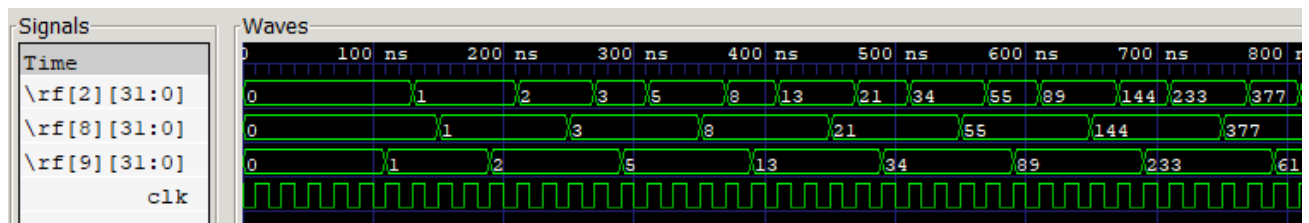


Рис. 2: Вейвформа для последовательности Фибоначчи

2.3 Моделирование извлечения квадратного корня

Ассемблерный код функции для вычисления квадратного корня представлен на листинге ниже.

```
## unsigned isqrt (unsigned x) {
##     unsigned m, y, b;
##     m = 0x40000000;
##     y = 0;
##     while (m != 0) { // Do 16 times
##         b = y | m;
##         y >>= 1;
##         if (x >= b) {
##             x -= b;
##             y |= m;
##         }
##         m >>= 2;
##     }
##     return y;
## }

.text

init:    li        $a0, 82          ## x = 82
         li        $v0, 0          ## calculation result reset

sqrt:    lui        $t0, 0x4000     ## m = 0x40000000
         move       $t1, $0         ## y = 0

L0:      or         $t2, $t1, $t0    ## b = y | m;
         srl        $t1, $t1, 1     ## y >>= 1
         sltu       $t3, $a0, $t2    ## if (x < b)
         bnez       $t3, L1          ## goto L1
         ## else
         subu       $a0, $a0, $t2    ## x -= b
         or         $t1, $t1, $t0    ## y |= m

L1:      srl        $t0, $t0, 2      ## m >>= 2
```

```

        bnez    $t0, L0          ## if(m != 0) goto L0
        move    $v0, $t1        ## return y

end:     beqz    $0, end         ## while(1);

```

Моделирование программы проводилось в среде Icarus Verilog.

Лог выполнения программы:

```

0  pc = x  pcaddr = xxxxxxxX  instr = 24040052  v0 = 0  addiu $4, $0, 82
1  pc = x  pcaddr = xxxxxxxX  instr = 24040052  v0 = 0  addiu $4, $0, 82
2  pc = 0  pcaddr = 00000000  instr = 24040052  v0 = 0  addiu $4, $0, 82
3  pc = 0  pcaddr = 00000000  instr = 24040052  v0 = 0  addiu $4, $0, 82
4  pc = 0  pcaddr = 00000000  instr = 24040052  v0 = 0  addiu $4, $0, 82
5  pc = 0  pcaddr = 00000000  instr = 24020000  v0 = 0  addiu $2, $0, 0
6  pc = 0  pcaddr = 00000000  instr = 3c084000  v0 = 0  lui    $8, 16384
7  pc = 0  pcaddr = 00000000  instr = 00004821  v0 = 0  addu   $9, $0, $0
8  pc = 0  pcaddr = 00000000  instr = 01285025  v0 = 0  or     $10, $9, $8
9  pc = 0  pcaddr = 00000000  instr = 00094842  v0 = 0  srl    $9, $0, $9
10 pc = 0  pcaddr = 00000000  instr = 008a582b  v0 = 0  sltu   $11, $4, $10
11 pc = 1  pcaddr = 00000004  instr = 15600002  v0 = 0  bne    $11, $0, 3
12 pc = 1  pcaddr = 00000004  instr = 00084082  v0 = 0  srl    $8, $0, $8
13 pc = 1  pcaddr = 00000004  instr = 1500fff8  v0 = 0  bne    $8, $0, -7
14 pc = 1  pcaddr = 00000004  instr = 01285025  v0 = 0  or     $10, $9, $8
15 pc = 1  pcaddr = 00000004  instr = 00094842  v0 = 0  srl    $9, $0, $9
16 pc = 1  pcaddr = 00000004  instr = 008a582b  v0 = 0  sltu   $11, $4, $10
17 pc = 1  pcaddr = 00000004  instr = 15600002  v0 = 0  bne    $11, $0, 3
18 pc = 1  pcaddr = 00000004  instr = 00084082  v0 = 0  srl    $8, $0, $8
19 pc = 1  pcaddr = 00000004  instr = 1500fff8  v0 = 0  bne    $8, $0, -7
20 pc = 1  pcaddr = 00000004  instr = 01285025  v0 = 0  or     $10, $9, $8
21 pc = 1  pcaddr = 00000004  instr = 00094842  v0 = 0  srl    $9, $0, $9
22 pc = 1  pcaddr = 00000004  instr = 008a582b  v0 = 0  sltu   $11, $4, $10
23 pc = 1  pcaddr = 00000004  instr = 15600002  v0 = 0  bne    $11, $0, 3
24 pc = 1  pcaddr = 00000004  instr = 00084082  v0 = 0  srl    $8, $0, $8
25 pc = 1  pcaddr = 00000004  instr = 1500fff8  v0 = 0  bne    $8, $0, -7
26 pc = 1  pcaddr = 00000004  instr = 01285025  v0 = 0  or     $10, $9, $8
27 pc = 1  pcaddr = 00000004  instr = 00094842  v0 = 0  srl    $9, $0, $9
28 pc = 1  pcaddr = 00000004  instr = 008a582b  v0 = 0  sltu   $11, $4, $10
29 pc = 1  pcaddr = 00000004  instr = 15600002  v0 = 0  bne    $11, $0, 3
30 pc = 1  pcaddr = 00000004  instr = 00084082  v0 = 0  srl    $8, $0, $8
31 pc = 1  pcaddr = 00000004  instr = 1500fff8  v0 = 0  bne    $8, $0, -7
32 pc = 1  pcaddr = 00000004  instr = 01285025  v0 = 0  or     $10, $9, $8
33 pc = 1  pcaddr = 00000004  instr = 00094842  v0 = 0  srl    $9, $0, $9
34 pc = 1  pcaddr = 00000004  instr = 008a582b  v0 = 0  sltu   $11, $4, $10
35 pc = 1  pcaddr = 00000004  instr = 15600002  v0 = 0  bne    $11, $0, 3
36 pc = 1  pcaddr = 00000004  instr = 00084082  v0 = 0  srl    $8, $0, $8
37 pc = 1  pcaddr = 00000004  instr = 1500fff8  v0 = 0  bne    $8, $0, -7
38 pc = 1  pcaddr = 00000004  instr = 01285025  v0 = 0  or     $10, $9, $8
39 pc = 1  pcaddr = 00000004  instr = 00094842  v0 = 0  srl    $9, $0, $9

```

40	pc = 1	pcaddr = 00000004	instr = 008a582b	v0 = 0	sltu	\$11, \$4, \$10
41	pc = 1	pcaddr = 00000004	instr = 15600002	v0 = 0	bne	\$11, \$0, 3
42	pc = 1	pcaddr = 00000004	instr = 00084082	v0 = 0	srl	\$8, \$0, \$8
43	pc = 1	pcaddr = 00000004	instr = 1500fff8	v0 = 0	bne	\$8, \$0, -7
44	pc = 1	pcaddr = 00000004	instr = 01285025	v0 = 0	or	\$10, \$9, \$8
45	pc = 1	pcaddr = 00000004	instr = 00094842	v0 = 0	srl	\$9, \$0, \$9
46	pc = 1	pcaddr = 00000004	instr = 008a582b	v0 = 0	sltu	\$11, \$4, \$10
47	pc = 1	pcaddr = 00000004	instr = 15600002	v0 = 0	bne	\$11, \$0, 3
48	pc = 1	pcaddr = 00000004	instr = 00084082	v0 = 0	srl	\$8, \$0, \$8
49	pc = 1	pcaddr = 00000004	instr = 1500fff8	v0 = 0	bne	\$8, \$0, -7
50	pc = 1	pcaddr = 00000004	instr = 01285025	v0 = 0	or	\$10, \$9, \$8
51	pc = 1	pcaddr = 00000004	instr = 00094842	v0 = 0	srl	\$9, \$0, \$9
52	pc = 1	pcaddr = 00000004	instr = 008a582b	v0 = 0	sltu	\$11, \$4, \$10
53	pc = 1	pcaddr = 00000004	instr = 15600002	v0 = 0	bne	\$11, \$0, 3
54	pc = 1	pcaddr = 00000004	instr = 00084082	v0 = 0	srl	\$8, \$0, \$8
55	pc = 1	pcaddr = 00000004	instr = 1500fff8	v0 = 0	bne	\$8, \$0, -7
56	pc = 1	pcaddr = 00000004	instr = 01285025	v0 = 0	or	\$10, \$9, \$8
57	pc = 1	pcaddr = 00000004	instr = 00094842	v0 = 0	srl	\$9, \$0, \$9
58	pc = 1	pcaddr = 00000004	instr = 008a582b	v0 = 0	sltu	\$11, \$4, \$10
59	pc = 1	pcaddr = 00000004	instr = 15600002	v0 = 0	bne	\$11, \$0, 3
60	pc = 1	pcaddr = 00000004	instr = 00084082	v0 = 0	srl	\$8, \$0, \$8
61	pc = 1	pcaddr = 00000004	instr = 1500fff8	v0 = 0	bne	\$8, \$0, -7
62	pc = 1	pcaddr = 00000004	instr = 01285025	v0 = 0	or	\$10, \$9, \$8
63	pc = 1	pcaddr = 00000004	instr = 00094842	v0 = 0	srl	\$9, \$0, \$9
64	pc = 1	pcaddr = 00000004	instr = 008a582b	v0 = 0	sltu	\$11, \$4, \$10
65	pc = 1	pcaddr = 00000004	instr = 15600002	v0 = 0	bne	\$11, \$0, 3
66	pc = 1	pcaddr = 00000004	instr = 00084082	v0 = 0	srl	\$8, \$0, \$8
67	pc = 1	pcaddr = 00000004	instr = 1500fff8	v0 = 0	bne	\$8, \$0, -7
68	pc = 1	pcaddr = 00000004	instr = 01285025	v0 = 0	or	\$10, \$9, \$8
69	pc = 1	pcaddr = 00000004	instr = 00094842	v0 = 0	srl	\$9, \$0, \$9
70	pc = 1	pcaddr = 00000004	instr = 008a582b	v0 = 0	sltu	\$11, \$4, \$10
71	pc = 1	pcaddr = 00000004	instr = 15600002	v0 = 0	bne	\$11, \$0, 3
72	pc = 1	pcaddr = 00000004	instr = 00084082	v0 = 0	srl	\$8, \$0, \$8
73	pc = 1	pcaddr = 00000004	instr = 1500fff8	v0 = 0	bne	\$8, \$0, -7
74	pc = 1	pcaddr = 00000004	instr = 01285025	v0 = 0	or	\$10, \$9, \$8
75	pc = 1	pcaddr = 00000004	instr = 00094842	v0 = 0	srl	\$9, \$0, \$9
76	pc = 1	pcaddr = 00000004	instr = 008a582b	v0 = 0	sltu	\$11, \$4, \$10
77	pc = 1	pcaddr = 00000004	instr = 15600002	v0 = 0	bne	\$11, \$0, 3
78	pc = 1	pcaddr = 00000004	instr = 00084082	v0 = 0	srl	\$8, \$0, \$8
79	pc = 1	pcaddr = 00000004	instr = 1500fff8	v0 = 0	bne	\$8, \$0, -7
80	pc = 1	pcaddr = 00000004	instr = 01285025	v0 = 0	or	\$10, \$9, \$8
81	pc = 1	pcaddr = 00000004	instr = 00094842	v0 = 0	srl	\$9, \$0, \$9
82	pc = 1	pcaddr = 00000004	instr = 008a582b	v0 = 0	sltu	\$11, \$4, \$10
83	pc = 0	pcaddr = 00000000	instr = 15600002	v0 = 0	bne	\$11, \$0, 3
84	pc = 0	pcaddr = 00000000	instr = 008a2023	v0 = 0	subu	\$4, \$4, \$10
85	pc = 0	pcaddr = 00000000	instr = 01284825	v0 = 0	or	\$9, \$9, \$8
86	pc = 0	pcaddr = 00000000	instr = 00084082	v0 = 0	srl	\$8, \$0, \$8
87	pc = 0	pcaddr = 00000000	instr = 1500fff8	v0 = 0	bne	\$8, \$0, -7

ные способы моделирования. В процессе был получен опыт работы с платой DE10-Lite, на которой проверялась работоспособность полученного устройства.

Список использованных источников

1. Thomas D., Moorby P. The Verilog Hardware Description Language. – Springer Science & Business Media, 2008.
2. Khor W. Y. et al. Evaluation of FPGA Based QSPI Flash Access Using Partial Reconfiguration //2019 7th International Conference on Smart Computing & Communications (ICSCC). – IEEE, 2019. – С. 1-5