

Учереждение образования
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ»

Кафедра интеллектуальных информационных технологий

Отчет по лабораторной работе №1

по курсу «МРЗВИС»

на тему: «Реализация модели решения задачи на конвейерной архитектуре»

Выполнили:

Студенты группы 821702

Зайцев Н.А.

Шепко М.Т.

Проверил:

Крачковский Д.Я.

Минск 2020

Тема: "Реализация модели решения задачи на конвейерной архитектуре"

Цель: Реализовать и исследовать модель решения на конвейерной архитектуре задачи вычисления попарного произведения (деления (обращения)) компонентов двух векторов чисел.

Описание модели: краткое описание особенностей

Модель арифметического (сбалансированного) конвейера, реализующего операцию произведения пары 6-разрядных чисел умножением со старших разрядов со сдвигом множимого (частичного произведения) вправо.

Данный конвейер состоит из 3 этапов:

Такт	Этапы		
	1	2	3
1	1 разряд $a1*b1$		
2	2 разряд $a1*b1$		
3	1 разряд $a2*b2$	3 разряд $a1*b1$	
4	2 разряд $a2*b2$	4 разряд $a1*b1$	
5	1 разряд $a3*b3$	3 разряд $a2*b2$	5 разряд $a1*b1$
6	2 разряд $a3*b3$	4 разряд $a2*b2$	6 разряд $a1*b1$
7		3 разряд $a3*b3$	5 разряд $a2*b2$
8		4 разряд $a3*b3$	6 разряд $a2*b2$
9			5 разряд $a3*b3$
10			6 разряд $a3*b3$

Такт 1:

(I) Вычисляется умножение первых разрядов первой пары чисел.

Такт 2:

(I) Вычисляется умножение вторых разрядов первой пары чисел

Такт 3:

(I) Вычисляется умножение первых разрядов второй пары чисел.

(II) Вычисляется умножение третьих разрядов первой пары чисел.

Такт 4:

(I) Вычисляется умножение вторых разрядов второй пары чисел.

(II) Вычисляется умножение четвёртых разрядов первой пары чисел.

Такт 5:

(I) Вычисляется умножение первых разрядов третьей пары чисел

(II) Вычисляется умножение третьих разрядов второй пары чисел.

(III) Вычисляется умножение пятых разрядов первой пары чисел.

Такт 6:

(I) Вычисляется умножение вторых разрядов третьей пары чисел

(II) Вычисляется умножение четвёртых разрядов второй пары чисел.

(III) Вычисляется умножение шестых разрядов первой пары чисел.

Такт 7:

(II) Вычисляется умножение третьих разрядов третьей пары чисел.

(III) Вычисляется умножение пятых разрядов второй пары чисел.

Такт 8:

(II) Вычисляется умножение четвёртых разрядов третьей пары чисел.

(III) Вычисляется умножение шестых разрядов второй пары чисел.

Такт 9:

(III) Вычисляется умножение пятых разрядов третьей пары чисел.

Такт 10:

(III) Вычисляется умножение шестых разрядов третьей пары чисел.

$p = 6$ - разрядность умножаемых чисел

$2 * p = 12$ – разрядность частичного произведения и суммы частичных произведений

Количество этапов конвейера – 3 ($= n$)

Количество пар задается пользователем – m

Работа конвейера. Результаты счёта и времена их получения:

Пользователю предлагается самостоятельно выбрать не только количество пар чисел, над которыми будут производиться операции, но и сами числа.

Числа предлагаются выбрать от 0 до 63, так как число 64 и выше будет иметь 7-разрядное значение.

При вводе неправильного числа программа экстренно завершает работу

```

Количество пар чисел: 3

1 пара:
Введите первый множитель: 4
Введите второй множитель: 5

2 пара:
Введите первый множитель: 20
Введите второй множитель: 15

3 пара:
Введите первый множитель: 60
Введите второй множитель: 45

```

Пример ввода чисел

Числа, введенные в десятичной системе, переводятся в двоичную систему. Далее взаимодействие происходит именно с ними. В конце ответы отображаются как в двоичной, так и в десятичной системах счисления.

```

Результат перевода с десятичной в двоичную систему
1 пара:

Первый множитель: 4 = 000100
Второй множитель: 5 = 000101
*****

2 пара:

Первый множитель: 20 = 010100
Второй множитель: 15 = 001111
*****

3 пара:

Первый множитель: 60 = 111100
Второй множитель: 45 = 101101
*****

```

Перевод с десятичной системы в двоичную

Далее пользователю предлагаются три варианта:

- 1-Посмотреть работу по тактам
- 2-Посмотреть итоговый результат
- 3-Выйти из программы

При нажатии клавиш 2,3 программа перейдет непосредственно к итоговому результату(без просмотра тактов) и выхода из программы соответственно.

```

Консоль отладки Microsoft Visual Studio

Результат перевода с десятичной в двоичную систему
1 пара:

Первый множитель: 4 = 000100
Второй множитель: 5 = 000101
*****

2 пара:

Первый множитель: 6 = 000110
Второй множитель: 20 = 010100
*****

Что вы хотите сделать?
1-Посмотреть работу по тактам
2-Посмотреть итоговый результат
3-Выйти из программы

Результаты вычислений:
1 пара: 4 * 5 = 20 = 0000-0001-0100
2 пара: 6 * 20 = 24 = 0000-0001-1000

```

Пример “2” варианта развития события

```

Консоль отладки Microsoft Visual Studio

Результат перевода с десятичной в двоичную систему
1 пара:

Первый множитель: 4 = 000100
Второй множитель: 6 = 000110
*****

Что вы хотите сделать?
1-Посмотреть работу по тактам
2-Посмотреть итоговый результат
3-Выйти из программы

```

Пример “3” варианта развития событий

При введении первого варианта будет выведена таблица потактового выполнения конвейера. Но пользователю остается возможность пропустить потактовое выполнение и сразу вывести ответ (скрин приведен ниже).

Сама таблица представляет собой 4 столбца.

- 1-столбец “№ Такта”
- 2-столбец “1 этап”
- 3-столбец “2 этап”
- 4-столбец “3 этап”

При этом программа предусматривает то, что пользователь введет одну, две или три пары чисел. В зависимости от этого будет выведена таблица с разным количеством тактов:

- для одной пары – 6 тактов
- для двух пар – 8 тактов
- для трех пар – 10 тактов
- и т.д.

При введении более трех пар элементов, то будет выведен результат всех операций в двух системах счисления и количество тактов, которые понадобились для их вычисления (скрин приведен ниже).

```

Что вы хотите сделать?
1-Посмотреть работу по тактам
2-Посмотреть итоговый результат
3-Выйти из программы

Кол-во пар: 6   Кол-во тактов: 16

      Результаты вычислений:
1 пара: 12 * 20 = 240 = 0000-1111-0000
2 пара: 25 * 60 = 1500 = 0101-1101-1100
3 пара: 63 * 31 = 1953 = 0111-1010-0001
4 пара: 15 * 48 = 720 = 0010-1101-0000
5 пара: 20 * 22 = 440 = 0001-1011-1000
6 пара: 14 * 12 = 168 = 0000-1010-1000

```

Вывод количества тактов и результат вычислений при введении более 3 пар чисел

№ Такта	1 этап	2 этап	3 этап
такт 1	-----	-----	-----
такт 2	-----	-----	-----
такт 3	-----	-----	-----
такт 4	-----	-----	-----
такт 5	-----	-----	-----
такт 6	-----	0000-0110-0100	0011-0011-0100
такт 7	-----	-----	-----
такт 8	-----	-----	-----

Следующий такт - 1
Все такты - 2

№ Такта	1 этап	2 этап	3 этап
такт 1	-----	-----	-----
такт 2	-----	-----	-----
такт 3	-----	-----	-----
такт 4	-----	-----	-----
такт 5	-----	-----	-----
такт 6	-----	-----	-----
такт 7	-----	-----	-----
такт 8	-----	-----	0001-1111-0100

Кол-во пар: 2 Кол-во тактов: 8

Результаты вычислений:

1 пара: 41 * 20 = 820 = 0011-0011-0100
2 пара: 25 * 20 = 500 = 0001-1111-0100

Результат выполнения работы

Графики (всего четыре семейства):

Обозначения:

$K_y(n,r) = T_1/T_n$;

$e(n,r) = K_y(n,r)/n$;

где $K_y(n,r)$ – коэффициент ускорения;

$e(n,r)$ – эффективность;
 n – количество процессорных элементов в системе;
 k – количество пар, поступающих на вход;
 r – ранг;

Данный график построен на основании таблицы:

m	n	r	T1	Tn	Ky	e
1	3	1	6	6	1	0,333333333
2	3	2	12	8	1,5	0,5
3	3	3	18	10	1,8	0,6
4	3	4	24	12	2	0,666666667
5	3	5	30	14	2,142857143	0,714285714
6	3	6	36	16	2,25	0,75
7	3	7	42	18	2,333333333	0,777777778
8	3	8	48	20	2,4	0,8
9	3	9	54	22	2,454545455	0,818181818
10	3	10	60	24	2,5	0,833333333
11	3	11	66	26	2,538461538	0,846153846
12	3	12	72	28	2,571428571	0,857142857
13	3	13	78	30	2,6	0,866666667
14	3	14	84	32	2,625	0,875
15	3	15	90	34	2,647058824	0,882352941
16	3	16	96	36	2,666666667	0,888888889
17	3	17	102	38	2,684210526	0,894736842
18	3	18	108	40	2,7	0,9
19	3	19	114	42	2,714285714	0,904761905
20	3	20	120	44	2,727272727	0,909090909
21	3	21	126	46	2,739130435	0,913043478
22	3	22	132	48	2,75	0,916666667
23	3	23	138	50	2,76	0,92

График 1. График зависимости коэффициента ускорения K_u от ранга задачи r

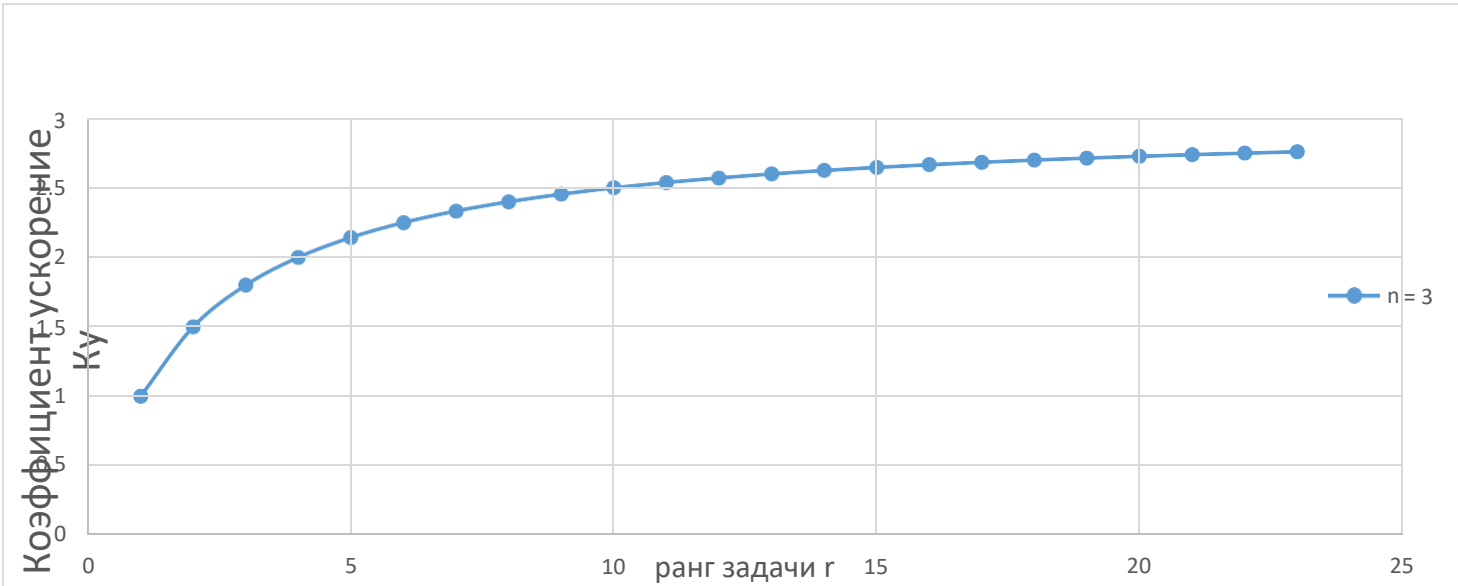


График 2. График зависимости коэффициента ускорения K_u от количества процессорных элементов n



График 3. График зависимости эффективности e от ранга задачи r

На следующем графике можно увидеть, как график растет к этому значению, но сможет достичь этого значения только при $r \rightarrow \infty$.

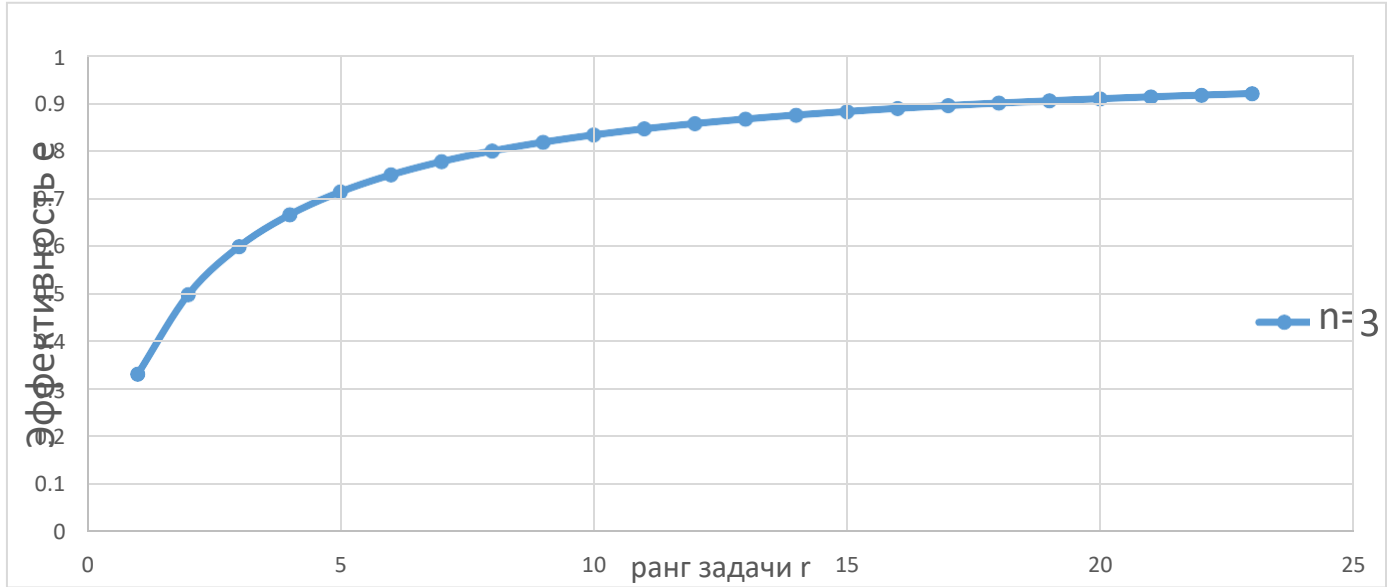


График 4. График зависимости эффективности e от количества процессорных элементов n



Вопросы и ответы на них:

Ответы на вопросы непосредственно по реализованной модели:

1. Проверить, что модель создана верно: программа работает правильно (на всех этапах конвейера)

Имеются исходные векторы шестиразрядных чисел:

A = <25, 16, 61>

B = <20, 48, 55>

Входные пары:

Первая умножаемая пара - <25, 20>

Вторая умножаемая пара - <16, 48>

Третья умножаемая пара - <61, 55>

Проверка результатов:

○ $25 * 20 = 500$

○ $16 * 48 = 768$

○ $61 * 55 = 3355$

Результаты верны. Скриншот, подтверждающие корректную работу программы.

```
Первый множитель: 25 = 011001
Второй множитель: 20 = 010100
*****
2 пара:
Первый множитель: 16 = 010000
Второй множитель: 48 = 110000
*****
3 пара:
Первый множитель: 61 = 111101
Второй множитель: 55 = 110111
*****
Что вы хотите сделать?
1-Посмотреть работу по тактам
2-Посмотреть итоговый результат
3-Выйти из программы

Результаты вычислений:
1 пара: 25 * 20 = 500 = 0001-1111-0100
2 пара: 16 * 48 = 768 = 0011-0000-0000
3 пара: 61 * 55 = 3355 = 1101-0001-1011
```

2. Объяснить на графиках точки перегиба и асимптоты

Асимптоты означают, что рост производительности конвейера ограничен и зависит от количества процессорных элементов и объектов.

3. Спрогнозировать как измениться вид графиков при изменении параметров модели

Если увеличивается ранг задачи r , то коэффициент ускорения и эффективность увеличиваются, что видно из вышеперечисленных графиков.

4. Каково соотношение между параметрами n, r, m, p модели сбалансированного конвейера

m – задается пользователем

$$r = 3$$

$$p = 6$$

$$n = 3$$

5. Допустим: имеется некоторая характеристика h (эффективность e или ускорение K_y) и для нее выполняется:

$$\circ h(n_1, r_1) = h(n_2, r_2)$$

$$\circ n_1 > n_2$$

$$e(n_1, r_1) = e(n_2, r_2);$$

$$e = \frac{K_y}{n} = \frac{r}{n+r-1};$$

$$\frac{r_1}{n_1+r_1-1} = \frac{r_2}{n_2+r_2-1};$$

$$r_1 * n_2 + r_1 * r_2 - r_1 = r_2 * n_1 + r_2 * r_1 - r_2;$$

$$r_1 * (n_2 - 1) = r_2 * (n_1 - 1);$$

$$\frac{r_2}{r_1} = \frac{n_2-1}{n_1-1};$$

Т.к. $n_1 > n_2 > 1$, то $r_1 > r_2$

6. Дано:

1. несбалансированный конвейер (заданы конкретные значения: $n, \{t_i\}$ – времена выполнения обработки на этапах конвейера);

2. e_0 – некоторое фиксированное значение эффективности. Определить значение r_0 , при котором выполняется $e(n, r_0) > e_0$? (Получить формулу, затем подставить в неё значения параметров.)

Так как в результате построения графика получилась гипербола, большему значению x соответствует меньшее значение y . Следовательно, для того, чтобы значение e было больше e_0 , величина r должна находиться в интервале $r \in (0; r_0)$.

7. Для несбалансированного конвейера (использовать исходные данные предыдущего вопроса) определить: $\lim(e(n, r))$ при $r \rightarrow \infty$.

Так как $e(n, r) = \frac{r \sum_{i=1}^n t_i}{n(\sum_{i=1}^n t_i + (r-1)t_{\max})}$, то предел находим по правилу Лопиталя

$$\lim_{r \rightarrow \infty} e(n, r) = \lim_{r \rightarrow \infty} \left(\frac{r \sum_{i=1}^n t_i}{n(\sum_{i=1}^n t_i + (r-1)t_{\max})} \right) = \lim_{r \rightarrow \infty} \left(\frac{\sum_{i=1}^n t_i}{n \left(\frac{\sum_{i=1}^n t_i}{r} + (r-1)t_{\max}/r \right)} \right) = \frac{\sum_{i=1}^n t_i}{n t_{\max}}$$

Предел эффективности при $r \rightarrow \infty$ равен 0.

8. Дан несбалансированный конвейер (использовать исходные данные предыдущего вопроса). Каким образом можно перестроить данный конвейер, чтобы для заданного r_0 выполнялось $e(n, r_0) > e_0$?

Изменить структуру конвейера так, чтобы число r принадлежало интервалу $r \in (0; r_0)$.

9. Дан несбалансированный конвейер (использовать исходные данные предыдущего вопроса) и значение минимального кванта времени t_0 (условной временной единицы). Каким образом нужно перестроить данный конвейер, чтобы получить максимально быстрый конвейер?

Необходимо разделить его на столько этапов, чтобы время каждого этапа было равно

Вывод:

В результате выполнения лабораторной работы была реализована модель конвейера для вычисления произведения пар чисел умножением с старших разрядов со сдвигом множимого вправо. Реализованная модель была проверена на работоспособность и правильность получаемых результатов. Данная модель позволяет ускорить процесс вычисления результата для векторов значений (нескольких пар).

Были исследованы числовые характеристики конвейерной архитектуры: коэффициент ускорения и эффективность при решении поставленной задачи.