

Министерство образования Республики Беларусь  
Учреждение образования  
“Белорусский государственный университет информатики и  
радиоэлектроники”  
Факультет информационных технологий и управления  
Кафедра интеллектуальных информационных технологий

Отчёт по лабораторной работе №1 по курсу «МРЗВИС»  
на тему «Реализация модели решения задачи на  
конвейерной архитектуре»

Выполнил  
студент группы  
821701

Грицель И.А.

Проверили

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 8-разрядных чисел умножением со старших разрядов со сдвигом множимого (частичного произведения) вправо.

Данный конвейер содержит 3 этапа, представленных тремя видами операций: вычисление множимого суммы, сдвиг множимого суммы вправо (он выполняется в любом случае даже если значение разряда равно 0, так как на последующих этапах используется сдвинутая сумма, поэтому мы всегда сдвигаем на 1 разряд) и вычисление промежуточного результата.

**Алгоритм:**

Умножение со старших разрядов 0110.0110*1001.1001= 0011.1100.1111.0110 (0) (1) (2) (3) (4) (5) (6) (7)		
Обозначим множимое 0110 за М; номер разряда за i, а его значение за X		
№ (номер разряда в числе)	Арифметические действия	Пояснение
8	1- 0110.0110.0000.0000 2- 0011.0011.0000.0000 3- 0011.0011.0000.0000	1 - Вычисление множимого-1 (i=0): Xi*M=1*0110.0110 2 - Сдвиг множимого-1 вправо на 1 разряд 3 - Прибавление множимого-1 к промежуточному результату
7	1- 0000.0000.0000.0000 2- 0000.0000.0000.0000 3- 0011.0011.0000.0000	1 - Вычисление множимого-2 (i=1): Xi*M=0000.0000 2 - Сдвиг множимого-2 вправо на 1 разряд 3 - Прибавление множимого-2 к промежуточному результату
6	1- 0000.0000.0000.0000 2- 0000.0000.0000.0000 3- 0011.0011.0000.0000	1 - Вычисление множимого-3 (i=2): Xi*M=0000.0000 2 - Сдвиг множимого-3 вправо на 1 разрядов 3 - Прибавление множимого-3 к промежуточному результату

5	1- 0110.0110.0000.0000 2- 0000.0110.0110.0000 3- 0011.1001.0110.0000	1 - Вычисление множимого-4 (i=3): $X_i * M = 0110.0110$ 2 - Сдвиг множимого-4 вправо на 1 разряд 3 - Прибавление множимого-4 к промежуточному результату
4	1-0110.0110.0000.0000 2-0000.0011.0011.0000 3-0011.1100.1001.0000	1- Вычисление множимого-5 (i=4): $X_i * M = 0110.0110$ 2- Сдвиг множимого-5 вправо на 1 разряд 3- Прибавление множимого-5 к промежуточному результату
3	1-0000.0000.0000.0000 2-0000.0000.0000.0000 3-0011.1100.1001.0000	1- Вычисление множимого-6 (i=5): $X_i * M = 0000.0000$ 2- Сдвиг множимого-6 вправо на 1 разряд 3- Прибавление множимого-6 к промежуточному результату
2	1-0000.0000.0000.0000 2-0000.0000.0000.0000 3-0011.1100.1001.0000	1- Вычисление множимого-7 (i=6): $X_i * M = 0000.0000$ 2- Сдвиг множимого-7 вправо на 1 разряд 3- Прибавление множимого-7 к промежуточному результату
1	1-0110.0110.0000.0000 2-0000.0000.0110.0110 3-0011.1100.1111.0110	1- Вычисление множимого-8 (i=7): $X_i * M = 0110.0110$ 2- Сдвиг множимого-8 вправо на 1 разряд 3- Прибавление множимого-8 к промежуточному результату

## Исходные данные:

$p = 8$  - разрядность умножаемых чисел

$2 * p = 16$  – разрядность частичного произведения и суммы частичных произведений

Количество этапов конвейера – 26 ( $= n$ ) Количество пар - 3

## Работа конвейера. Результаты счёта и времена их получения:

Количества пар равно 3, числа генерируются случайно, однако не составит труда добавить возможность вводить их самому пользователю при необходимости.

```

par 0 :41 107
0010.1001 0110.1011
par 1 :214 235
1101.0110 1110.1011
par 2 :44 169
0010.1100 1010.1001

```

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.

Ниже изображено то, что видит пользователь в качестве ответа- результат в двоичной системе

```
rez0: 0001.0100.0101.0111
rez1: 1101.0101.0010.1010
rez2: 0010.1011.1101.0100
```

Выводится таблица, в которой выделены «множимое», «сдвиг» и «промежуточный результат».

```
1: par 1: par 2: par 3: opred mnozetela 0010.1100
2: par 1: par 2: opred mnozetela 1101.0110 par 3: sdvig mnozetela 0001.0110.0
3: par 1: opred mnozetela 0000.0000 par 2: sdvig mnozetela 0110.1011.0 par 3: pribavlenie k rezultatu 0001.0110.0000.0000
4: par 1: sdvig mnozetela 0000.0000 par 2: pribavlenie k rezultatu 0110.1011.0000.0000 par 3: opred mnozetela 0000.0000
5: par 1: pribavlenie k rezultatu 0000.0000.0000.0000 par 2: opred mnozetela 1101.0110 par 3: sdvig mnozetela 0000.1011.00
6: par 1: opred mnozetela 0010.1001 par 2: sdvig mnozetela 0011.0101.10 par 3: pribavlenie k rezultatu 0010.0001.0000.0000
7: par 1: sdvig mnozetela 0000.1010.01 par 2: pribavlenie k rezultatu 1010.0000.1000.0000 par 3: opred mnozetela 0010.1100
8: par 1: pribavlenie k rezultatu 0000.1010.0100.0000 par 2: opred mnozetela 1101.0110 par 3: sdvig mnozetela 0000.0101.100
9: par 1: opred mnozetela 0010.1001 par 2: sdvig mnozetela 0001.1010.110 par 3: pribavlenie k rezultatu 0010.0110.1000.0000
10: par 1: sdvig mnozetela 0000.0101.001 par 2: pribavlenie k rezultatu 1011.1011.0100.0000 par 3: opred mnozetela 0000.0000
11: par 1: pribavlenie k rezultatu 0000.1111.0110.0000 par 2: opred mnozetela 0000.0000 par 3: sdvig mnozetela 0000.0010.1100
12: par 1: opred mnozetela 0000.0000 par 2: sdvig mnozetela 0000.1101.0110 par 3: pribavlenie k rezultatu 0010.1001.0100.0000
13: par 1: sdvig mnozetela 0000.0010.1001 par 2: pribavlenie k rezultatu 1100.1000.1010.0000 par 3: opred mnozetela 0010.1100
14: par 1: pribavlenie k rezultatu 0001.0001.1111.0000 par 2: opred mnozetela 1101.0110 par 3: sdvig mnozetela 0000.0001.0110.0
15: par 1: opred mnozetela 0010.1001 par 2: sdvig mnozetela 0000.0110.1011.0 par 3: pribavlenie k rezultatu 0010.1010.1010.0000
16: par 1: sdvig mnozetela 0000.0001.0100.1 par 2: pribavlenie k rezultatu 1100.1111.0101.0000 par 3: opred mnozetela 0000.0000
17: par 1: pribavlenie k rezultatu 0001.0011.0011.1000 par 2: opred mnozetela 0000.0000 par 3: sdvig mnozetela 0000.0000.1011.00
18: par 1: opred mnozetela 0000.0000 par 2: sdvig mnozetela 0000.0011.0101.10 par 3: pribavlenie k rezultatu 0010.1011.0101.0000
19: par 1: sdvig mnozetela 0000.0000.1010.01 par 2: pribavlenie k rezultatu 1101.0010.1010.1000 par 3: opred mnozetela 0000.0000
20: par 1: pribavlenie k rezultatu 0001.0011.1101.1100 par 2: opred mnozetela 1101.0110 par 3: sdvig mnozetela 0000.0000.0101.100
21: par 1: opred mnozetela 0010.1001 par 2: sdvig mnozetela 0000.0001.1010.110 par 3: pribavlenie k rezultatu 0010.1011.1010.1000
22: par 1: sdvig mnozetela 0000.0000.0101.001 par 2: pribavlenie k rezultatu 1101.0100.0101.0100 par 3: opred mnozetela 0010.1100
23: par 1: pribavlenie k rezultatu 0001.0100.0010.1110 par 2: opred mnozetela 1101.0110 par 3: sdvig mnozetela 0000.0000.0010.1100
24: par 1: opred mnozetela 0010.1001 par 2: sdvig mnozetela 0000.0000.1101.0110 par 3: pribavlenie k rezultatu 0010.1011.1101.0100
25: par 1: sdvig mnozetela 0000.0000.0010.1001 par 2: pribavlenie k rezultatu 1101.0101.0010.1010 par 3:
26: par 1: pribavlenie k rezultatu 0001.0100.0101.0111 par 2: par 3:
```

**Графики (всего четыре семейства):**

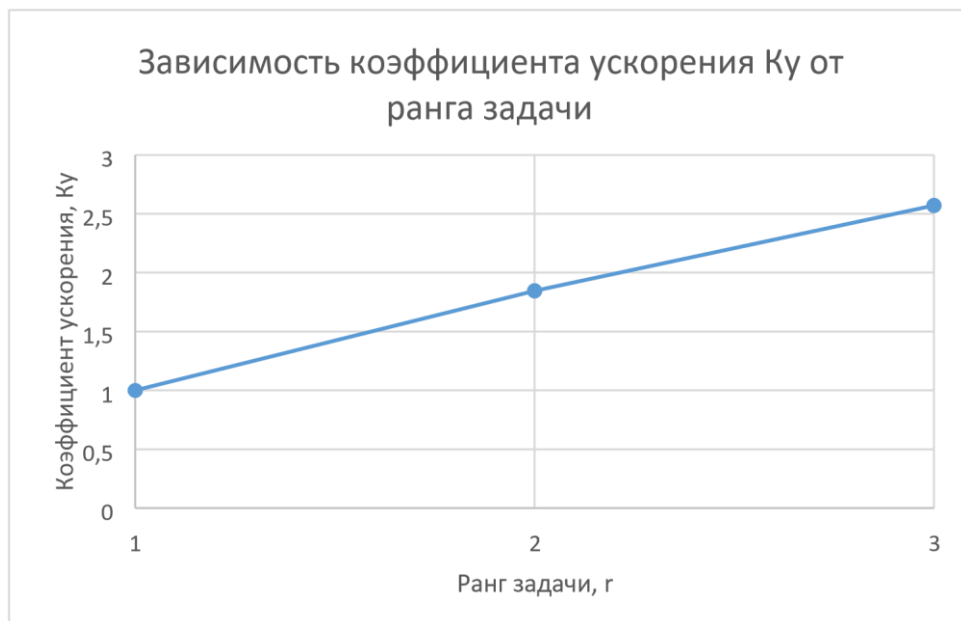
Обозначения:

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

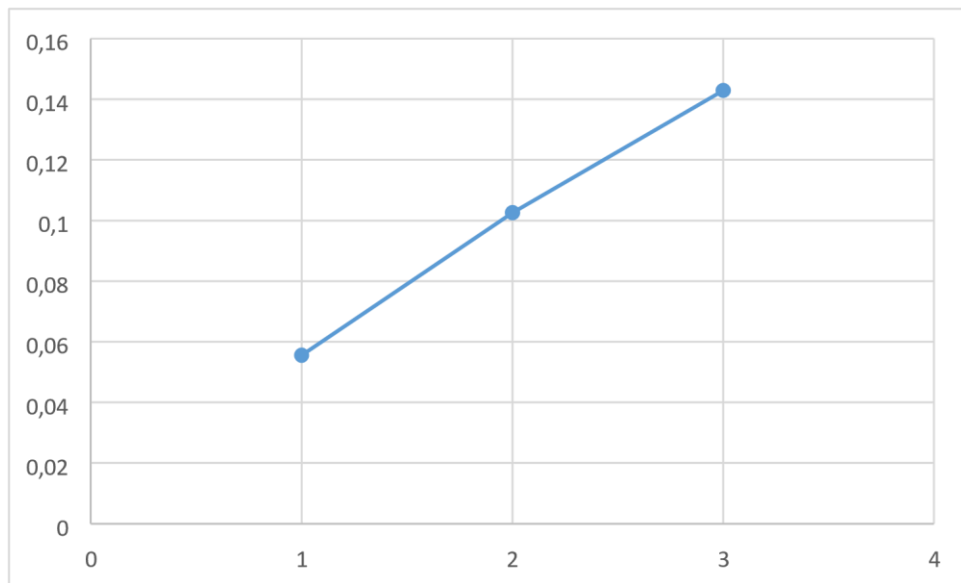
$= K_y(n,r)/n$ ; где

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

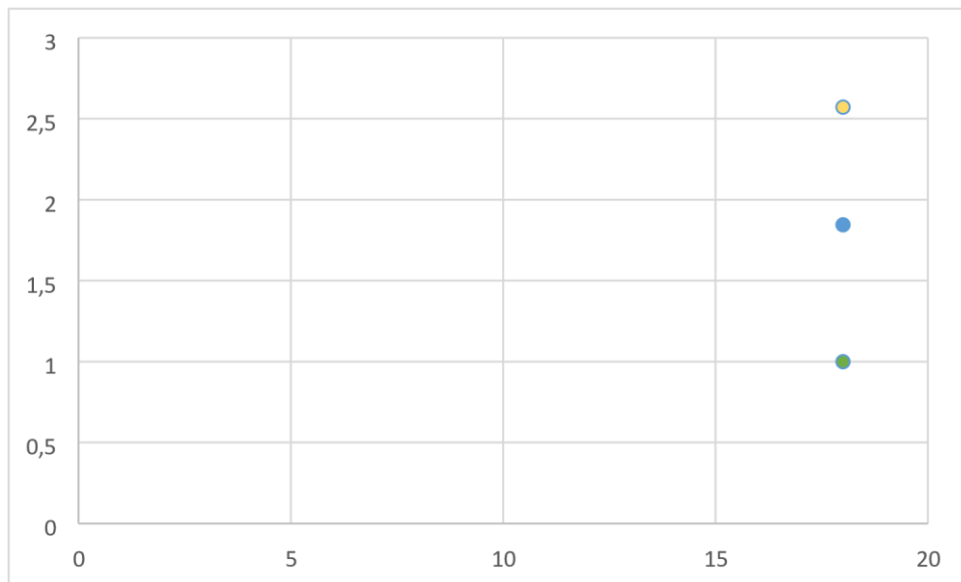
**График 1. График зависимости коэффициента ускорения  $K_u$  от ранга задачи  $r$**



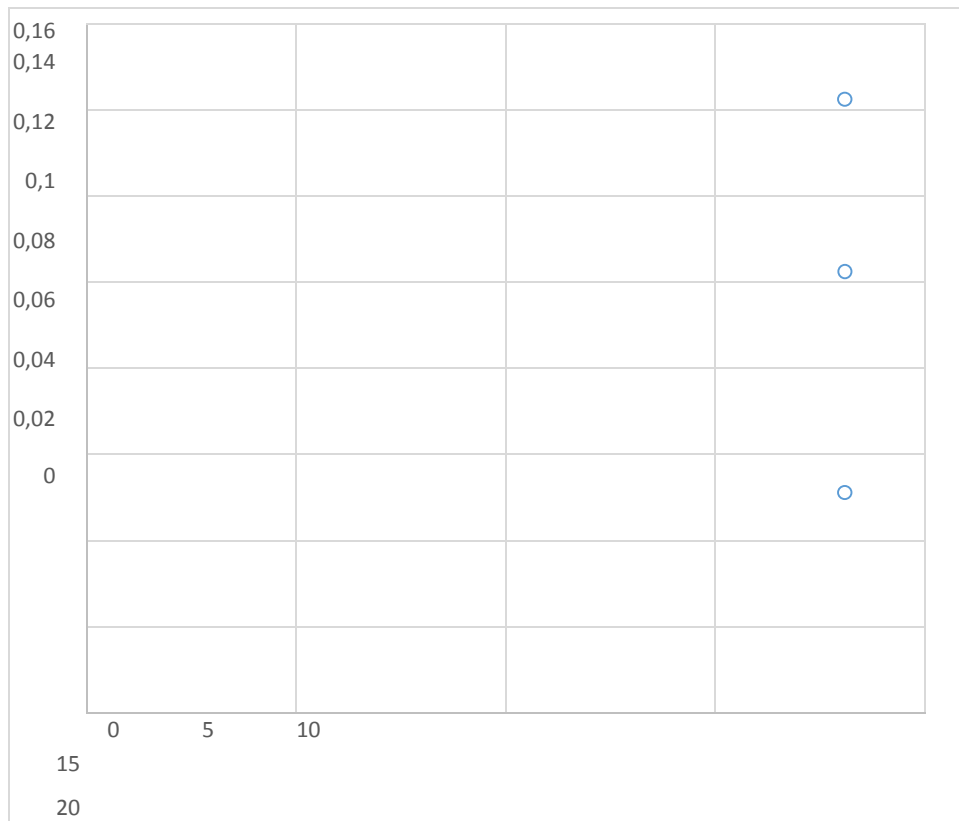
**График 2. График зависимости эффективности  $e$  от ранга задачи  $r$**



**График 3. График зависимости коэффициента ускорения  $K_u$  от количества этапов  $n$**



**График 4. График зависимости эффективности  $e$  от количества этапов  $n$**



## Вопросы и ответы на них:

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

Имеются исходные векторы четырехразрядных чисел: А

$$= \langle 11, 13, 11 \rangle$$

$$В = \langle 9, 15, 13 \rangle$$

Входные пары:

Первая умножаемая пара -  $\langle 11, 9 \rangle$

Вторая умножаемая пара -  $\langle 13, 15 \rangle$  Третья

умножаемая пара -  $\langle 11, 13 \rangle$  Проверка

результатов:

$$\circ 11 * 9 = 99 \circ 13$$

$$* 15 = 195 \circ 11*$$

$$13 = 143$$

Результаты верны. Скриншоты, подтверждающие корректную работу программы, приведены выше.

### 2. объяснить на графиках точки перегиба и асимптоты

Для объяснения точек перегиба и асимптот обратимся к формулам:

$$Ky = \frac{T_1}{T_n}; Ky = \frac{r * n * t_i}{n * t_i + (r - 1) * t_i} = \frac{r * n}{n + r - 1}$$

$n \rightarrow \infty$  и  $r \rightarrow \infty$ :

$$\lim_{n \rightarrow \infty} Ky = \lim_{n \rightarrow \infty} \frac{r * n}{n + r - 1} = r; \lim_{r \rightarrow \infty} Ky = \lim_{r \rightarrow \infty} \frac{r * n}{n + r - 1} = n$$

Значит асимптотой для  $Ky$  будет являться прямая  $Ky = r$  при  $n = const$   
 $Ky = n$  при  $r = const$ .  
 Возьмём предел при

, и прямая

Для  
 эффективности  
 проделаем  
 аналогичную  
 работу:

$$e = \frac{Ky}{n} = \frac{r}{n + r - 1}; \lim_{n \rightarrow \infty} e = \lim_{n \rightarrow \infty} \frac{r}{n + r - 1} = 0; \lim_{r \rightarrow \infty} Ky = \lim_{r \rightarrow \infty} \frac{r}{n + r - 1} = 1$$

Значит асимптотой для  $e$  будет являться прямая  $e = 1$  при  $n = const$   
 $e = 0$  при  $r = const$ .

, и прямая

**3. спрогнозировать как измениться вид графиков при изменении параметров модели** ○ параметр  $r$  □ график  $Ky$ : при увеличении растет значение коэффициента ускорения остается

неизменным

□ график  $e$ :

при увеличении растет значение ускорения остается неизменным

○ параметр  $k$  □ график  $Ky$ :

при увеличении уменьшается значение коэффициента ускорения □

график  $e$ :

при увеличении падает значение ускорения

**4. каково соотношение между параметрами  $n$ ,  $r$ ,  $m$ ,  $p$  модели сбалансированного конвейера**  $m = 3$   $r = 3$   $p = 4$   
 $n = 18$

**5. допустим: имеется некоторая характеристика  $h$  (эффективность  $e$  или ускорение  $Ky$ ) и для нее выполняется:** ○  $h(n_1, r_1) = h(n_2, r_2)$  ○  $n_1 > n_2$



$$e(n_1, r_1) = e(n_1, r_1); \quad ;$$

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

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

Т.к.

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

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

6. дано:  $\frac{r_2}{r_1} = \frac{n_2-1}{n_1-1};$  ;

•  $n_1 > n_2 > 1, \text{ то } r_1 > r_2$

несбалансированный конвейер  
(заданы конкретные значения:

$n, \{t_i\}$  – времена выполнения обработки на этапах конвейера); •  $e_0$  – 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.)

$$e = \frac{Ky}{n} = \frac{T_1}{T_n * n}; \quad n \in N$$

$$T_n = \sum_{i=1}^n t_i + (r-1)t_{\max}$$

$$T_1 = r \sum_{i=1}^n t_i$$

$$e(n, r) = \frac{r \sum_{i=1}^n t_i}{n(\sum_{i=1}^n t_i + (r-1)t_{\max})} \Rightarrow \frac{r_0 \sum_{i=1}^n t_i}{n(\sum_{i=1}^n t_i + (r_0-1)t_{\max})} > e_0$$

$$r_0 \sum_{i=1}^n t_i > e_0 n \left( \sum_{i=1}^n t_i + (r_0 - 1)t_{\max} \right)$$

$$r_0 \sum_{i=1}^n t_i > e_0 n \sum_{i=1}^n t_i + e_0 n r_0 t_{\max} - e_0 n t_{\max}$$

$$r_0 \sum_{i=1}^n t_i - e_0 n r_0 t_{\max} > e_0 n \sum_{i=1}^n t_i - e_0 n t_{\max}$$

$$r_0 \left( \sum_{i=1}^n t_i - e_0 n t_{\max} \right) > e_0 n \left( \sum_{i=1}^n t_i - t_{\max} \right)$$

Необходимо определить знаки выражений:

$$\sum_{i=1}^n t_i - t_{\max} \geq 0$$

Если  $\sum_{i=1}^n t_i - e_0 n t_{\max} > 0$ , то

$$r_0 > \frac{e_0 n (\sum_{i=1}^n t_i - t_{\max})}{\sum_{i=1}^n t_i - e_0 n t_{\max}}$$

если  $\sum_{i=1}^n t_i - e_0 n t_{\max} < 0$ , то

$$r_0 < \frac{e_0 n (\sum_{i=1}^n t_i - t_{\max})}{\sum_{i=1}^n t_i - e_0 n t_{\max}}$$

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

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

$$\frac{\sum_{i=1}^n t_i}{n t_{\max}}$$

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

Т.к.  $e$  функция от двух переменных, и  $r_0$  задано, то необходимо найти при каком  $n$  будет выполняться заданное условие.

$$e(n, r) = \frac{r_0 \sum_{i=1}^n t_i}{n(\sum_{i=1}^n t_i + (r_0-1)t_{\max})} > e_0$$

$$< \frac{r_0 \sum_{i=1}^n t_i}{e_0(\sum_{i=1}^n t_i + (r_0-1)t_{\max})}.$$

$n$

Необходимо объединять этапы конвейера таким образом, чтобы выполнялось неравенство 
$$1 \leq n < \frac{r_0 \sum_{i=1}^n t_i}{e_0(\sum_{i=1}^n t_i + (r_0-1)t_{\max})}$$

Таким образом, конвейер необходимо перестроить с целью уменьшения  $n$  если оно выходит за указанный выше предел. Это можно сделать объединив некоторые этапы конвейера.

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? Получить для него формулы  $K_y(n, r)$ ,  $e(n, r)$ ?

Для того, чтобы получить максимально быстрый конвейер, нужно перестроить так, чтобы он стал сбалансированным, и каждый этап выполнялся за минимальное время  $t_0$ . Необходимо разделить его на столько этапов, чтобы время каждого этапа было равно  $t_0$ .

Следовательно:  $t_0 = t_i = t_{\max}$

$$K_y(n, r) = \frac{r \sum_{i=1}^n t_0}{\sum_{i=1}^n t_0 + (r-1)t_0} = \frac{rn}{n+(r-1)}.$$

Аналогично с эффективностью:

$$e(n, r) = \frac{r \sum_{i=1}^n t_0}{n(\sum_{i=1}^n t_0 + (r-1)t_0)} = \frac{r}{n+(r-1)}.$$

То есть необходимо разделить этапы конвейера, которые длятся дольше, чем  $t_0$ , на более мелкие этапы.

### **Вывод:**

В результате выполнения лабораторной работы была реализована модель сбалансированного конвейера для вычисления произведения пар чисел умножением со 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.