Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет информационных технологий и управления
Кафедра интеллектуальных информационных технологий

ОТЧЕТ
по лабораторной работе 1
по дисциплине

МОДЕЛИ РЕШЕНИЯ ЗАДАЧ В ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ

Вариант 10

Студент гр. 221702 А. Н. Хлуд
Руководитель В. П. Ивашенко

Минск 2025

СОДЕРЖАНИЕ

1 ХОД ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Вариант 10

Тема: конвейерная архитектура.

Цель: ознакомиться и получить навыки реализации модели решения задачи на конвейерной архитектуре.

Задание: реализовать алгоритм вычисления произведения пары 6-разрядных чисел умножением со старших разрядов со сдвигом частичной суммы влево.

1.1 Описание

Задача заключается в написании алгоритма вычисления произведения пары 6-разрядных чисел умножением со старших разрядов со сдвигом частичной суммы влево. При запуске программы пользователь вводит некоторое кол-во чисел для вектора A и B, чьи элементы будут попарно перемножаться.

Для реализации программы использовался язык программирования Python. Также из структур данных был использован список.

1.2 Структура программы

Программа включает в себя следующие функции:

– **check_values(a,b)** - проверка, что входные числа нужной разрядности p=6;

– **sum(p:str, s:str)** - выполняет сумму двух бинарных чисел;

– **algorithm(list)** - выполняет алгоритм для одного такта;

– **print_tact(queue, steps, res, tact)** - выводит информацию каждого такта;

– **main()** - функция, запускающая программу.

1.3 Описание модели

Декларативное описание модели:

а) Пользователь вводит вектор A и B;

б) Если вектора имеют разный размер или числа превышают разряд, равный 6, модель останавливает работу;

в) Инициализируется список, в который записывают преобразованные в двоичную систему пары из вектора A и B;

г) Для каждого этапа модель инициализирует список, состоящий из нулевых множимого, множителя, частичного произведения и частичной суммы.

д) Для каждого нового такта, модель берет значения, если они имеются, из этапов для обработки. Если модель берет из первого этапа, в первый этап записывается пара из списка пар, если тот не пуст.

е) Если модель берет данные из последнего этапа, частичная сумма будет результатом умножения.

ж) Взятые данные из этапа, модель сдвигает частичную сумму и множитель влево. Если старший разряд равен 1, записывает множимое в частичное произведение. Складывает частичную сумму с частичным произведением.

з) Если конвейер пуст и список пар пуст, модель завершает работу. Иначе повторяет шаги д-е.

Описание входов и выходов программы:

Вход: вектора A и B 6-разрядных чисел.

Выход: протокол работы конвейера.

Исходные данные:

– Разрядность(p) попарно умножаемых чисел - 6.

– Разрядность результата умножения 2*p - 12.

– Количество этапов конвейера p - 6.

– Количество пар m(зависит от количества введенных пар).

Демонстрация результатов работы программы:

Пример отработки для перемножения чисел 1 и 2, 3 и 4:

```
введите вектор A:
1, 2
введите вектор B:
3, 4
такт 0
Входная очередь:
1 и 3
2 и 4
этап 1
-
этап 2
-
этап 3
-
этап 4
-
этап 5
-
этап 6
-
```

```
такт 1
Входная очередь:
2 и 4
этап 1
множимое:  000001
множитель:  000011
частичное произведение:  000000000000
частичная сумма:  000000000000
этап 2
-
этап 3
-
этап 4
-
этап 5
-
этап 6
-
```

```
такт 2
Входная очередь:
-
этап 1
множимое:  000010
множитель:  000100
частичное произведение:  000000000000
частичная сумма:  000000000000
этап 2
множимое:  000001
множитель:  000110
частичное произведение:  000000000000
частичная сумма:  000000000000
этап 3
-
```

```
такт 3
Входная очередь:
-
этап 1
-
этап 2
множимое:  000010
множитель:  001000
частичное произведение:  000000000000
частичная сумма:  000000000000
этап 3
множимое:  000001
множитель:  001100
частичное произведение:  000000000000
частичная сумма:  000000000000
```

```
такт 4
Входная очередь:
-
этап 1
-
этап 2
-
этап 3
множимое:  000010
множитель:  010000
частичное произведение:  000000000000
частичная сумма:  000000000000
этап 4
множимое:  000001
множитель:  011000
частичное произведение:  000000000000
частичная сумма:  000000000000
этап 5
-
этап 6
-
```

```
такт 5
Входная очередь:
-
этап 1
-
этап 2
-
этап 3
-
этап 4
множимое:  000010
множитель:  100000
частичное произведение:  000000000010
частичная сумма:  000000000010
этап 5
множимое:  000001
множитель:  110000
частичное произведение:  000000000001
частичная сумма:  000000000001
этап 6
-
```

```
такт 6
Входная очередь:
-
этап 1
-
этап 2
-
этап 3
-
этап 4
-
этап 5
множимое:  000010
множитель:  000000
частичное произведение:  000000000010
частичная сумма:  000000000100
этап 6
множимое:  000001
множитель:  100000
частичное произведение:  000000000001
частичная сумма:  000000000011
Результат:
000000000011 (3)
```

```
такт 7
Входная очередь:
-
этап 1
-
этап 2
-
этап 3
-
этап 4
-
этап 5
-
этап 6
множимое:  000010
множитель:  000000
частичное произведение:  000000000010
частичная сумма:  000000001000
Результат:
000000000011 (3)
000000001000 (8)
```

Таблица 1.2 – Пример работы конвейера для умножения чисел 1 и 3, 2 и 4

1.4 Графики

Обозначения:

n – количество этапов.

r – ранг задачи(количество пар).

$T_1(n, r) = r * n$ – время, затрачиваемое на вычисления в однопроцессорной вычислительной системе.

$T_n(n, r) = n + r - 1$, при условии, что r>0 – время, затрачиваемое на вычисления в параллельной вычислительной системе.

$K_y(T_1, T_n) = \frac{T_1}{T_n}$ – коэффициент ускорения.

$e(K_y, n) = \frac{K_y}{n}$ – эффективность.

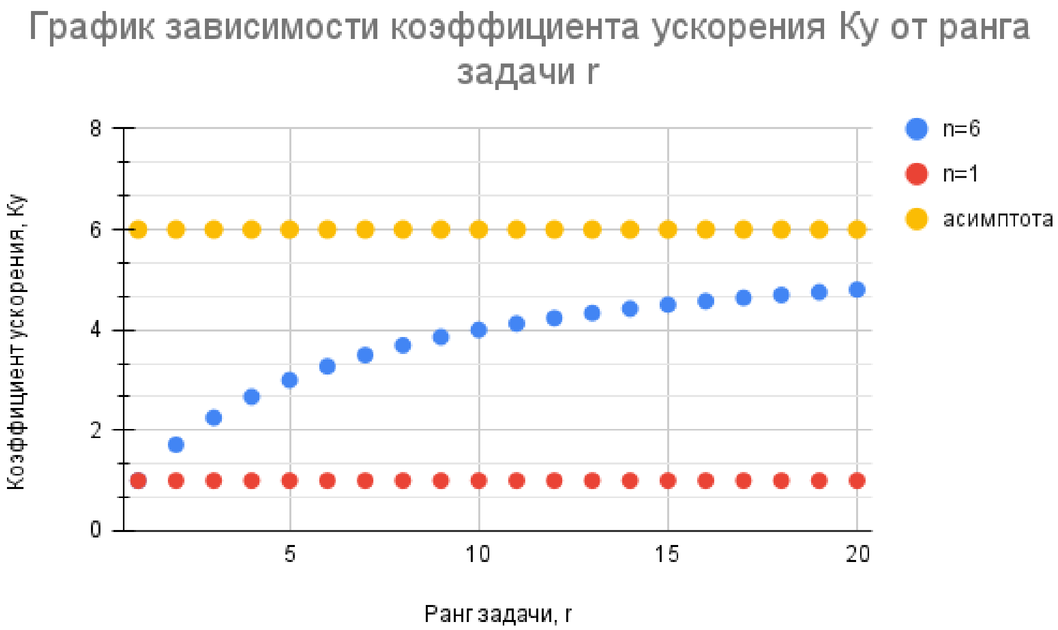

Рисунок 1.1 – График зависимости коэффициента ускорения K_y от ранга r

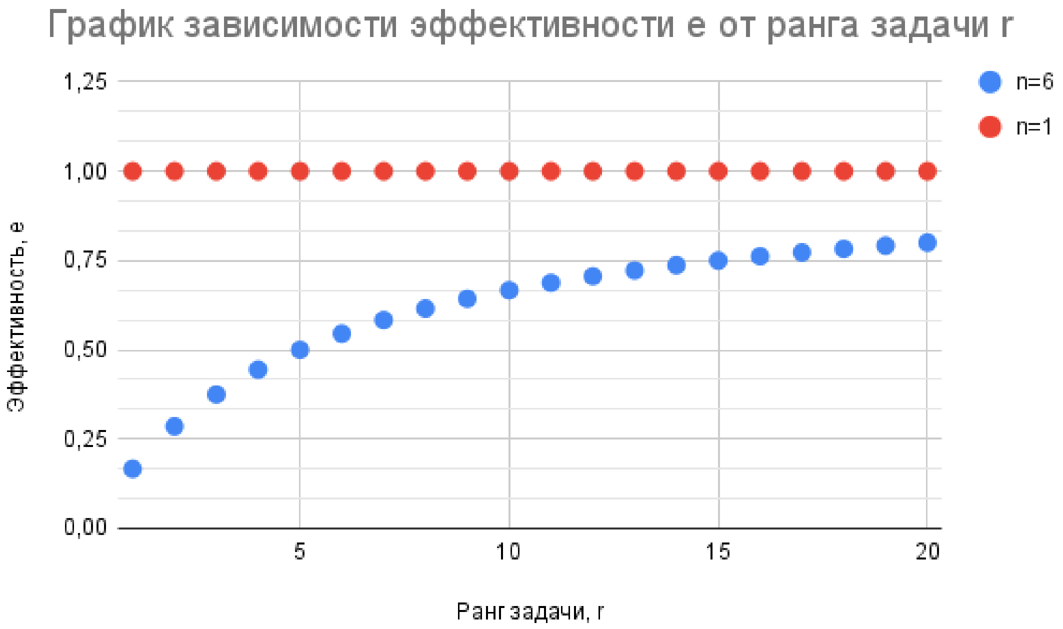

Рисунок 1.2 – График зависимости эффективности е от ранга r

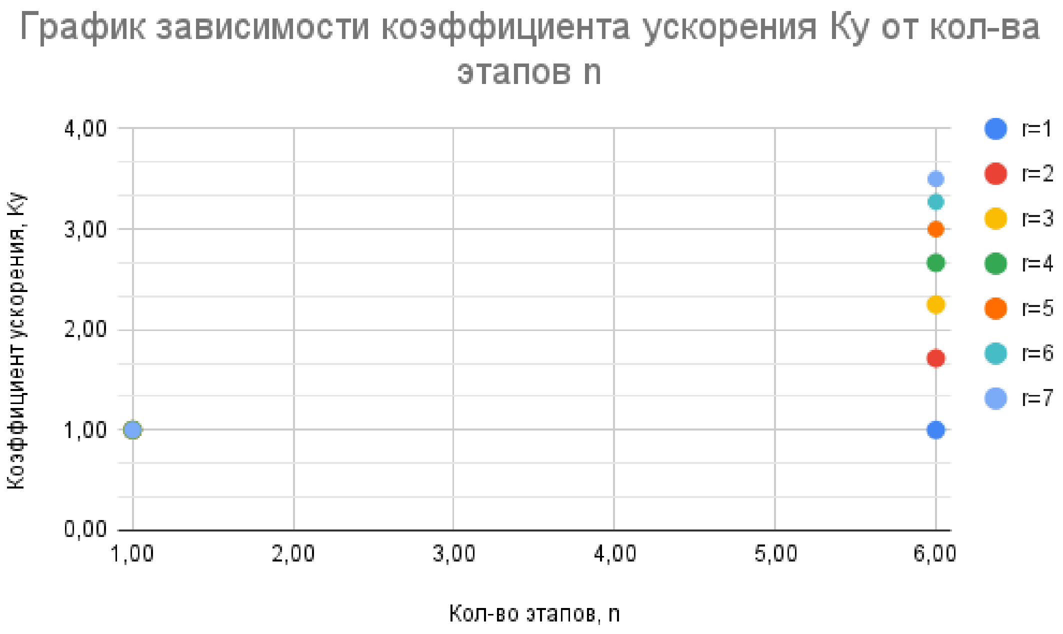

Рисунок 1.3 – График зависимости коэффициента ускорения K_y от количества этапов n

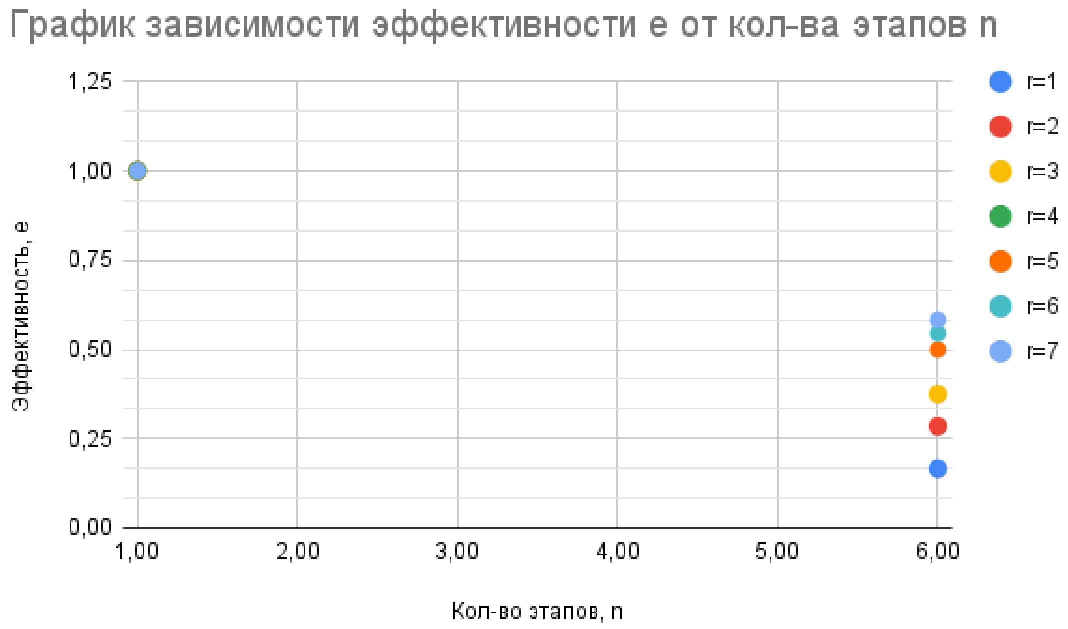

Рисунок 1.4 – График зависимости эффективности e от количества этапов n

1.5 Ответы на вопросы

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

Доказательство корректной работы модели представлена в пункте "Демонстрация результатов работы программы"в разделе 1.2.

1.5.2 Объясните на графиках точки перегиба и асимптоты

Для графиков коэффициента ускорения рассмотрим 2 случая:

Поскольку $K_y = \frac{r*n}{n+r-1}$, то, найдя для него предел при $r \to \infty$: $\lim_{r\to\infty} \frac{r*n}{n+r-1} = n$, мы поймем, что в данном случае коэффициент ускорения ограничен числом n(количество этапов конвейера). Ранг задачи принимает максимально возможное значение, вычисления выполняются на ограниченном количестве этапов.

При $n \to \infty$: $\lim_{n\to\infty} \frac{r*n}{n+r-1} = r$, таким образом при большом количестве вычислительных блоков, коэффициент ускорения ограничивается рангом задачи, поскольку в таком случае некоторое количество вычислительных блоков будет простаивать.

Для объяснения асимптот графика эффективности действуем аналогично:

Поскольку $e = \frac{r}{n+r-1}$, то при $r \to \infty$ предел принимает значение $\lim_{r\to\infty} \frac{r}{n+r-1} = 1$. Так как эффективность показывает долю работы одного процессорного элемента, то при ограниченном количестве блоков каждый блок будет задействован в вычислениях. Следовательно, эффективность будет максимальной(ни один вычислительный блок не будет простаивать).

При $n \to \infty$ предел принимает значение $\lim_{n\to\infty} \frac{r}{n+r-1} = 0$. В этом случае ситуация обратная: из-за того, что количество вычислительных блоков больше возможного ранга задачи, некоторые блоки будут простаивать. Следовательно, эффективность будет минимальной.

1.5.3 Спрогнозируйте, как изменится вид графиков при изменении параметров модели

$K_y(r)$ и $K_y(n)$ растут при увеличении r и n. При увеличении r значение эффективности e(r) растет. При увеличении n значение эффективности e(n) снижается. Причины данного поведения объяснены в ответе на предыдущий вопрос.

1.5.4 Какого соотношение между параметрами n, r, m, p

Ранг задачи(r) – максимальное количество экземпляров данных (одного типа) с необходимостью подлежащей обработке, которые могут быть обработаны (вне зависимости от варианта реализации) при решении этой задачи одновременно. Экземпляры данных одного типа – количество пар чисел(m). Следовательно, r = m.

Разрядность (p) – разрядность умножаемых попарно чисел (для данного варианта p = 6).

Количество (n) – количество процессорных элементов, n = p = 6.

1.5.5 Каким будет соотношение между r_1 и r_2 характеристики h, если для нее выполняется $h(n_1, r_1) = h(n_2, r_2), n_1 > n_2$?

Допустим, что имеется некоторая характеристика h (эффективность e или ускорение K_y) и для нее выполняется: $h(n_1, r_1) = h(n_2, r_2), n_1 > n_2$. Каким будет соотношение между r_1 и r_2?

В качестве характеристики h возьмем эффективность e. Тогда подставим формулу эффективности e:

$$e = \frac{K_y}{n} = \frac{T_1}{T_n * n} = \frac{r}{n+r-1}; n \in N; r \in N$$

в заданное выражение:

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

$$\begin{cases} \frac{r_1}{n_1+r_1-1} = \frac{r_2}{n_2+r_2-1} \\ r_1 \neq r_2 \\ n_1 > 1, n_2 \geq 1 \\ r_1 \geq 1, r_2 \geq 1 \\ n_1 > n_2 \end{cases} \iff$$

$$\iff \begin{cases} r_1 n_2 + r_1 r_2 - r_1 = r_2 n_1 + r_1 r_2 - r_2 \\ r_1 \neq r_2 \\ n_1 > 1, n_2 \geq 1 \\ r_1 \geq 1, r_2 \geq 1 \\ n_1 > n_2 \end{cases} \iff$$

$$\iff \begin{cases} r_1 n_2 - r_1 = r_2 n_1 - r_2 \\ r_1 \neq r_2 \\ n_1 > 1, n_2 \geq 1 \\ r_1 \geq 1, r_2 \geq 1 \\ n_1 > n_2 \end{cases} \iff$$

$$\iff \begin{cases} r_1(n_2 - 1) = r_2(n_1 - 1) \\ r_1 \neq r_2 \\ n_1 > 1, n_2 \geq 1 \\ r_1 \geq 1, r_2 \geq 1 \\ n_1 > n_2 \end{cases} \iff$$

$$\iff \begin{cases} \frac{r_2}{r_1} = \frac{n_2-1}{n_1-1} \\ r_1 \neq r_2 \\ n_1 > 1, n_2 \geq 1 \\ r_1 \geq 1, r_2 \geq 1 \\ n_1 > n_2 \end{cases} \iff$$

$$\iff \begin{cases} r_1 > r_2 \\ n_1 > 1, n_2 \geq 1 \\ r_1 \geq 1, r_2 \geq 1 \\ n_1 > n_2 \end{cases}$$

Ответ: $r_1 > r_2$.

1.6 Задачи

Дано: несбалансированный конвейер (заданы конкретные значения: n, t_i - времена выполнения обработки на этапах конвейера, e_0 - некоторое фиксированное значение эффективности.

1.6.1 Определить значение r_0, при котором выполняется $e(n, r_0) > e_0$. (Получить формулу и подставить в нее значение параметров).

При $r_0 > 0; t_i > 0; n > 0; t_{max} > t_i$ верно:
Время выполнения в однопроцессорной системе:

$$T_1 = r_0 \sum_{i=1}^{n} t_i$$

Время выполнения в конвейерной системе:

$$T_n = \sum_{i=1}^{n} t_i + (r_0 - 1) * t_{max}$$

Коэффициент ускорения:

$$K_y(n, r_0) = \frac{T_1}{T_n} = \frac{r_0 \sum_{i=1}^{n} t_i}{\sum_{i=1}^{n} t_i + (r_0 - 1) * t_{max}}$$

Эффективность:

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

Выразим r_0 :
Подставим выражение для эффективности:

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

Умножим обе части на знаменатель(положительный по условию $t_i > 0$) :

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

Раскроем скобки и сгруппируем слагаемые с r_0 :

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

Перенесем все члены с r_0 :

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

Для любого конвеера $\sum_{i=1}^{n} t_i - t_{max} > 0$

При $\sum_{i=1}^{n} t_i - e_0 n t_{max} > 0$:

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

удовлетворяет условиям $r_0 > 0; t_i > 0; n > 0; t_{max} \geq t_i, \sum_{i=1}^{n} t_i - t_{max} > 0$

При $\sum_{i=1}^{n} t_i - e_0 n t_{max} < 0$:

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

противоречит условию $\mathrm{r}_0 > 0, \sum_{i=1}^{n} t_i - e_0 n t_{max} < 0$ знак станет противоположным.

При $\sum_{i=1}^{n} t_i - e_0 n t_{max} = 0$

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

$$0 > e_0 n(\sum_{i=1}^{n} t_i - t_{max})$$

противоречит условиям $n > 0; \sum_{i=1}^{n} t_i - t_{max} > 0$

Отсюда:

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

Ответ: минимальное значение r_0, при котором эффективность превышает e_0:

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

при $r_0 > 0, t_i > 0, n > 0, t_{max} \geq t_i$

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

$$e(n,r) = \frac{K_y(n,r)}{n} = \frac{r * \sum_{i=1}^{n} t_i}{n * (\sum_{i=1}^{n} t_i + (r-1) * \max_{i=1}^{n}(t_i))}$$

Следовательно:

$$\lim_{r\to\infty} e(n,r) = \lim_{r\to\infty} \frac{r * \sum_{i=1}^{n} t_i}{n * (\sum_{i=1}^{n} t_i + (r-1) * \max_{i=1}^{n}(t_i))}$$

Для вычисления данного предела может быть использовано правило Лопиталя:

$$\lim_{x\to x_0} \frac{f(x)}{g(x)} = \lim_{x\to x_0} \frac{f'(x)}{g'(x)}$$

Тогда выражение примет вид:

$$\lim_{r\to\infty} e(n,r) = \lim_{r\to\infty} \frac{r * \sum_{i=1}^{n} t_i}{n * (\sum_{i=1}^{n} t_i + (r-1) * \max_{i=1}^{n}(t_i))} = \frac{\sum_{i=1}^{n} t_i}{n * \max_{i=1}^{n}(t_i)}$$

Ответ: $\lim_{r\to\infty} e(n,r) = \frac{\sum_{i=1}^{n} t_i}{n * \max_{i=1}^{n}(t_i)}$

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

$$e(n,r_0) > e_0$$

Для несбалансированное конвейера:

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

Следовательно, по условию дано:

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

При n>0:

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

Случай с $n \leq 0$ не имеет смысла, так как конвейер с отрицательным или нулевым количеством процессорных элементов не имеет смысла.

При $e_0 > 0$:

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

Случай с $e_0 \leq 0$ не имеет смысла, так как конвейер с отрицательной или нулевой эффективностью не имеет смысла.

Таким образом, конвейер можно перестроить путем изменения количества его процессорных элементов(или этапов) так, чтобы для n было верно $1 \leq n < \frac{r_0 \sum_{i=1}^{n_0} t_i}{e_0 * (\sum_{i=1}^{n} t_i + (r_0 - 1) * t_{max})}$ Добиться этого можно путем объединения этапов конвейера.

Приведем пример решения данной задачи на конкретных данных. Пусть $r_0 = 4$, $n_0 = 5$, $t_i = \{2, 3, 4, 6, 15\}$. Исходная эффективность $e_0 = 0.32$. Объединим 1-й этап со 2-ым, 3-й с 4-ым. Тогда получим n=3 и $t_i = \{5, 10, 15\}$ Тогда e=0.53, следовательно $e > e_0$.

Ответ: перестроить путем изменения количества его процессорных элементов(или этапов) так, чтобы для n было верно $1 \leq n < \frac{r_0 \sum_{i=1}^{n_0} t_i}{e_0 * (\sum_{i=1}^{n} t_i + (r_0 - 1) * t_{max})}$

1.6.4 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 Ky(n,r), e(n,r)

Данный конвейер необходимо перестроить так, чтобы он был сбалансированным и каждый этап выполнялся за минимальный квант времени t0, т.е. разделить этапы конвейера, которые длятся дольше, чем t0, на более мелкие этапы, которые будут длиться t0. Количество этапов в этом сбалансированном конвейере будет равняться:

$$n = \frac{\sum_{i=1}^{n} t_i}{t_0}$$

Тогда:

$$K_y = \frac{r * \frac{\sum_{i=1}^{n} t_i}{t_0}}{\frac{\sum_{i=1}^{n} t_i}{t_0} + r - 1}$$

$$e = \frac{r}{\frac{\sum_{i=1}^{n} t_i}{t_0} + r - 1}$$

1.7 Вывод

В ходе выполнения лабораторной работы приобретены навыки реализации моделей решения задач на конвейерной архитектуре. В рамках данной лабораторной работы была исследована и реализована модель сбалансированного конвейера для вычисления частного пар 6-разрядных чисел делением без восстановления частичного остатка. Были изучены различные числовые характеристики конвейерной архитектуры, а именно коэффициент ускорения и эффективность .

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Ивашенко, В. П. Модели решения задач в интеллектуальных системах : учеб.-метод. пособие: в 2 ч. Ч. 1 : Формальные модели обработки информации и параллельные модели решения задач / В. П. Ивашенко. — Минск : БГУИР, 2020.