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 дисциплине

МОДЕЛИ РЕШЕНИЯ ЗАДАЧ В ИНТЕЛЛЕКУТАЛЬНЫХ СИСТЕМАХ

Студент гр. 121701 Р. В. Липский
Руководитель В. П. Ивашенко

Минск 2024

СОДЕРЖАНИЕ

1 ХОД ВЫПОЛНЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ

Вариант 16

Тема: конвейерная архитектура.

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

Задание Реализовать алгоритм вычисления произведения пары 8-разрядных чисел умножением со старших разрядов со сдвигом частичной суммы влево

1.1 Описание

Задача заключается в написании алгоритма вычисления произведения пары 8-разрядных чисел умножением со старших разрядов со сдвигом частичноц суммы влево. Входом программы является файл, содержащий набор пар 8-разрядных чисел, выходом программы должно являтся графическое изображение, представляющее процесс решения данной задачи при помощи конвейерной архитектуры.

Для реализация программы использовался язык программирования Kotlin.

Были использованы следующие структуры данных:

– Список

– Кортеж

Структура приложения выглядит следующим образом:

– src – каталог, содержащий исходный код программы:

– src/main/kotlin/by/bsuir/ArtimeticPipeline.kt, BinaryNumber.kt – файлы, содержащие описание предметной области программы;

– src/main/kotlin/Main.kt – точка входа программы, содержащая алгоритм решения поставленной задачи.

1.2 Описание модели

1.2.1 Декларативное описание модели

а) Чтение входных пар из файла;

б) Инициализация конвейера, состоящего из 8 этапов;

в) Создается список троёк, где первое число соответствует множимому, второе число соответствует множителю, третье число соотвествует частичной сумме;

г) Инициализация пустого списка, в который будут помещаться результаты вычислений;

д) Если длина списка с результатами вычислений равна изначальной длине списка с входными парами, переходим к шагу (к);

е) Перемещаем четвёрки произведения каждого шага произведения на один шаг дальше (движение конвейера).

ж) Если список входных троек не пустой - перемещаем верхнюю тройку из списка на первый шаг конвейера.

з) Для каждого шара конвейера:

1) Записываем в протокол изначальное состояние этапа конвейера.

2) Сдвигаем частичную сумму влево на 1 бит;

3) Если первый бит множителя = 1, прибавляем к частичной сумме множимое;

4) Сдвигаем множитель влево на 1 бит;

5) Записываем в протокол конечно состояние этапа конвейера.

и) Переходим к шагу (е).

к) Выводим результат работы программы: протокол выполнения.

1.2.2 Описание входов и выходов модели

Вход: список пар, состоящих из 8-мираздядных чисел.
Выход: протокол работы конвейера

1.2.3 Исходные данные

– Разрядность (p) попарно умножаемых чисел равняется 8.
– Разрядность произведения равна 16.
– Количество этапов конвейера (n) равняется разрядности (p) – 8.
– Количество пар равно m и зависит от входного файла.

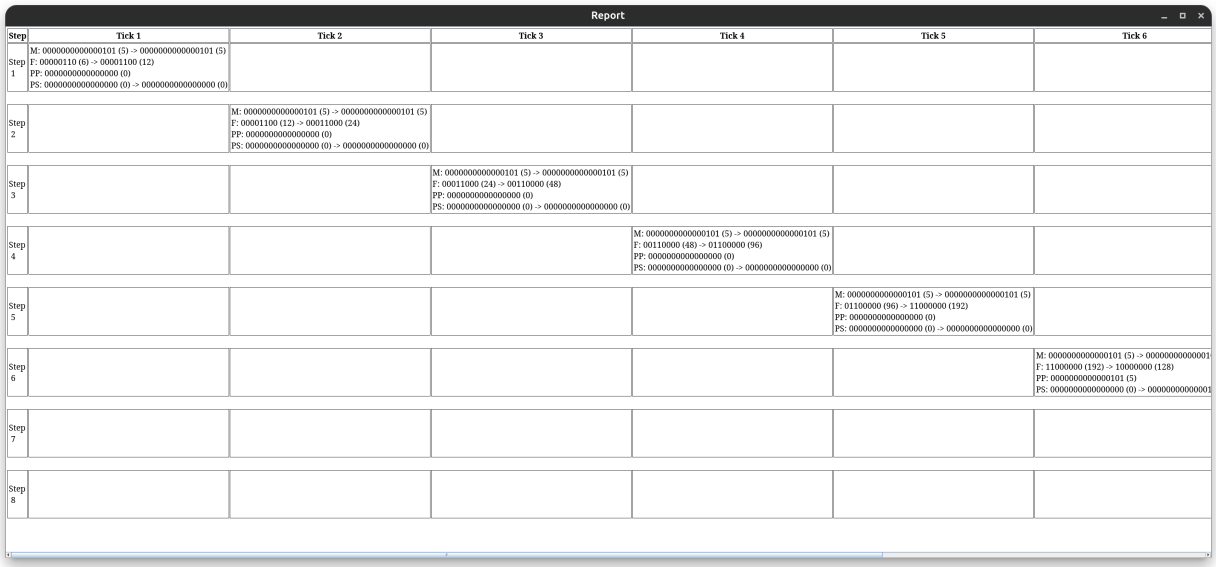
1.3 Демонстрация результатов работы программы

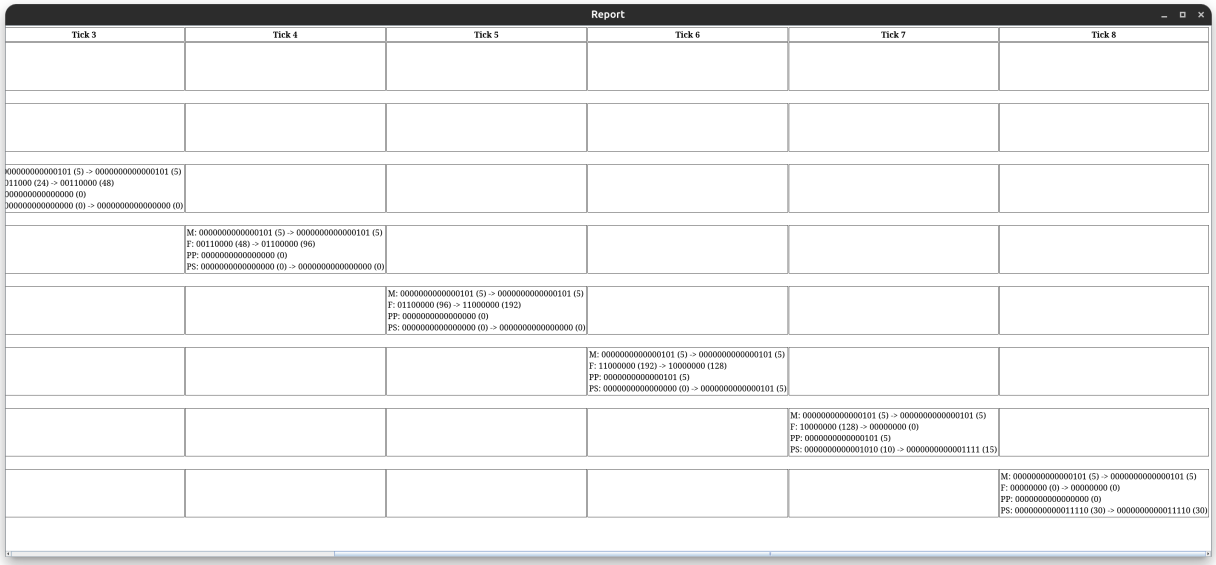
1.3.1 Тест 1

Входной файл input1.json:

```
[
    [5, 6]
]
```

Результат работы программы для файла input1.json:



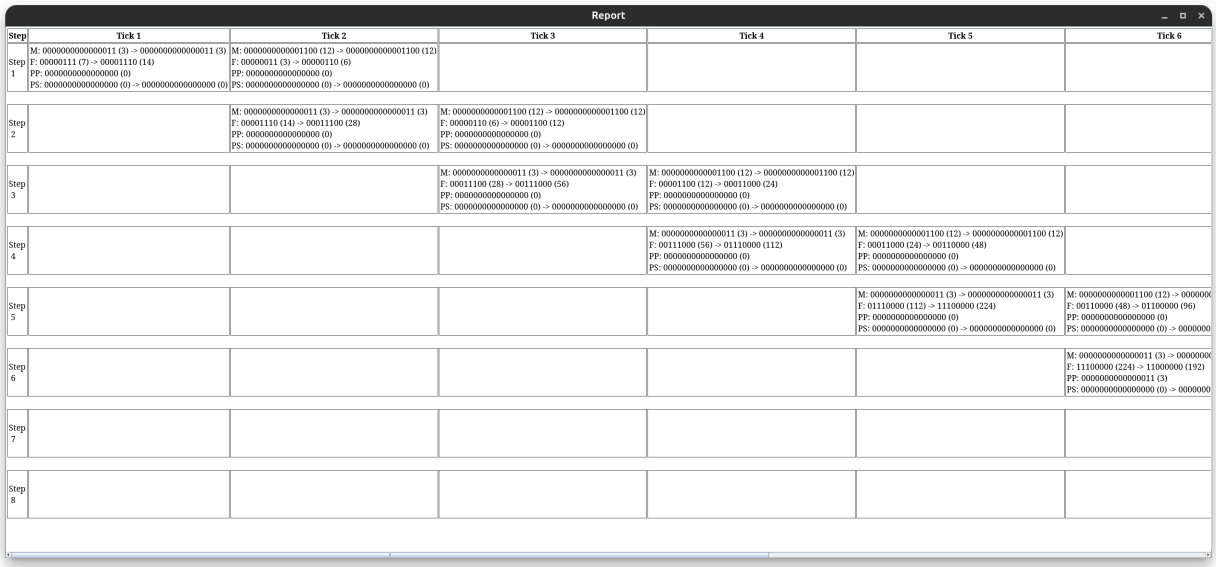


1.3.2 Тест 2

Входной файл input2.json:

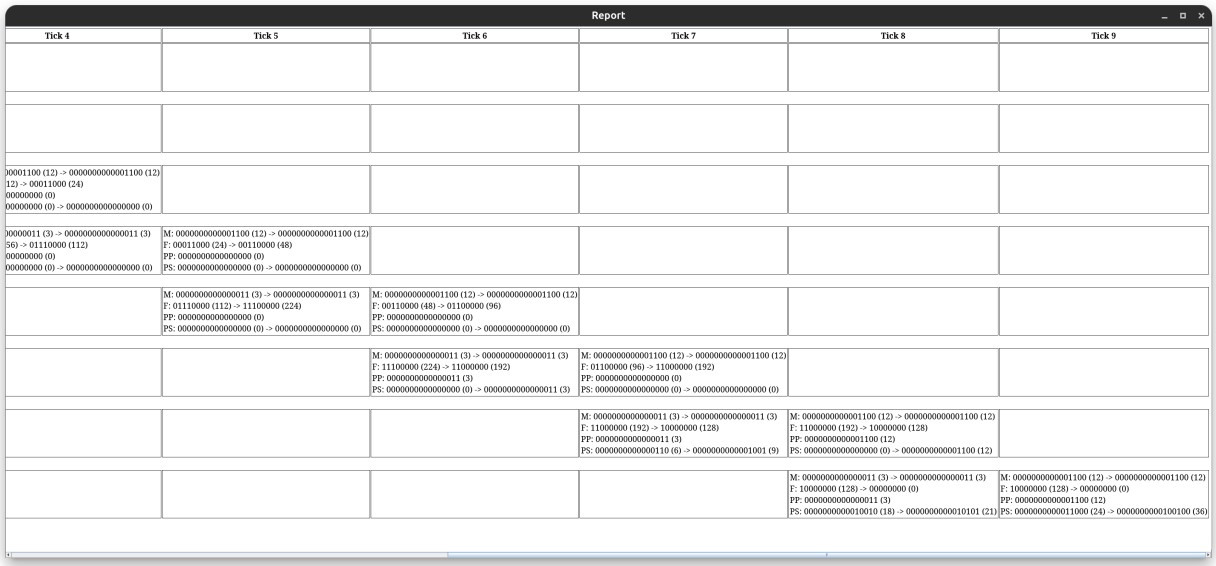
```
[
    [12, 3],
    [3, 7]
]
```

Результат работы программы для файла input2.json:



Report

Step	Tick 1	Tick 2	Tick 3	Tick 4	Tick 5	Tick 6
Step 1	M: 0000000000000011 (3) -> 0000000000000011 (3) F: 00000111 (7) -> 00001110 (14) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)	M: 0000000000001100 (12) -> 0000000000001100 (12) F: 00000011 (3) -> 00000110 (6) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)				
Step 2		M: 0000000000000011 (3) -> 0000000000000011 (3) F: 00001110 (14) -> 00011100 (28) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)	M: 0000000000001100 (12) -> 0000000000001100 (12) F: 00000110 (6) -> 00001100 (12) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)			
Step 3			M: 0000000000000011 (3) -> 0000000000000011 (3) F: 00011100 (28) -> 00111000 (56) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)	M: 0000000000001100 (12) -> 0000000000001100 (12) F: 00001100 (12) -> 00011000 (24) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)		
Step 4				M: 0000000000000011 (3) -> 0000000000000011 (3) F: 00111000 (56) -> 01110000 (112) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)	M: 0000000000001100 (12) -> 0000000000001100 (12) F: 00011000 (24) -> 00110000 (48) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)	
Step 5					M: 0000000000000011 (3) -> 0000000000000011 (3) F: 01110000 (112) -> 11100000 (224) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)	M: 0000000000001100 (12) -> 0000000 F: 00110000 (48) -> 01100000 (96) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000
Step 6						M: 0000000000000011 (3) -> 0000000 F: 11100000 (224) -> 11000000 (192) PP: 0000000000000011 (3) PS: 0000000000000000 (0) -> 0000000
Step 7						
Step 8						



Report

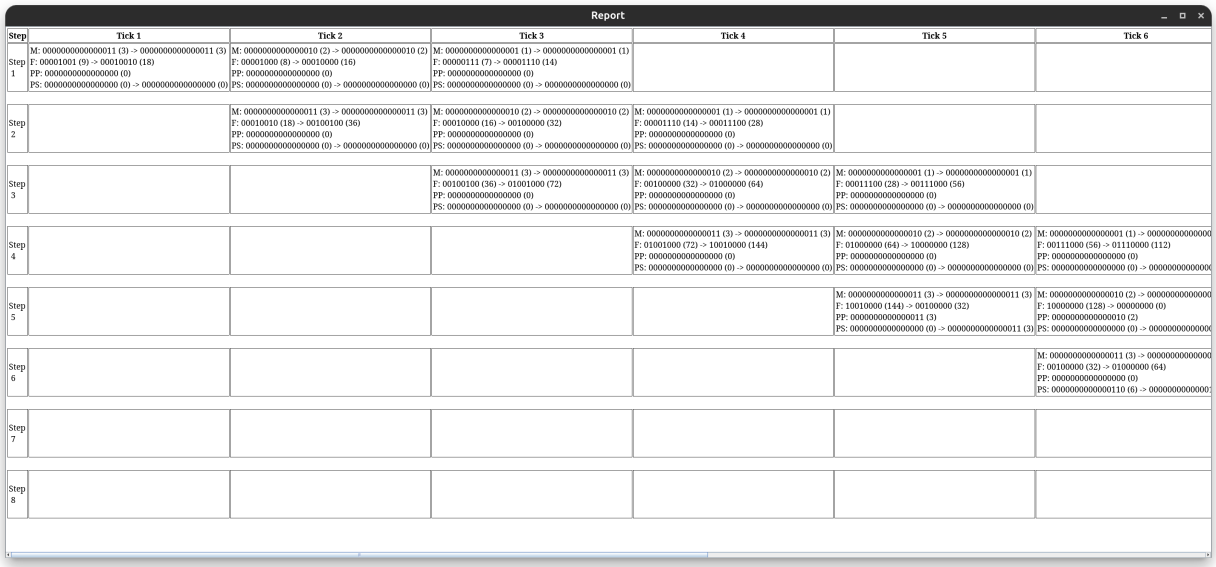
Tick 4	Tick 5	Tick 6	Tick 7	Tick 8	Tick 9
00001100 (12) -> 0000000000001100 (12) 12) -> 00011000 (24) 00000000 (0) 00000000 (0) -> 0000000000000000 (0)					
0000011 (3) -> 0000000000000011 (3) 56) -> 01110000 (112) 00000000 (0) 00000000 (0) -> 0000000000000000 (0)	M: 0000000000001100 (12) -> 0000000000001100 (12) F: 00011000 (24) -> 00110000 (48) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)				
	M: 0000000000000011 (3) -> 0000000000000011 (3) F: 01110000 (112) -> 11100000 (224) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)	M: 0000000000001100 (12) -> 0000000000001100 (12) F: 00110000 (48) -> 01100000 (96) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)			
		M: 0000000000000011 (3) -> 0000000000000011 (3) F: 11100000 (224) -> 11000000 (192) PP: 0000000000000011 (3) PS: 0000000000000000 (0) -> 0000000000000011 (3)	M: 0000000000001100 (12) -> 0000000000001100 (12) F: 01100000 (96) -> 11000000 (192) PP: 0000000000000000 (0) PS: 0000000000000000 (0) -> 0000000000000000 (0)		
			M: 0000000000000011 (3) -> 0000000000000011 (3) F: 11000000 (192) -> 10000000 (128) PP: 0000000000000011 (3) PS: 0000000000000110 (6) -> 0000000000001001 (9)	M: 0000000000001100 (12) -> 0000000000001100 (12) F: 11000000 (192) -> 10000000 (128) PP: 0000000000001100 (12) PS: 0000000000000000 (0) -> 0000000000001100 (12)	
				M: 0000000000000011 (3) -> 0000000000000011 (3) F: 10000000 (128) -> 00000000 (0) PP: 0000000000000011 (3) PS: 0000000000010010 (18) -> 0000000000010101 (21)	M: 0000000000001100 (12) -> 0000000000001100 (12) F: 10000000 (128) -> 00000000 (0) PP: 0000000000001100 (12) PS: 0000000000011000 (24) -> 0000000000100100 (36)

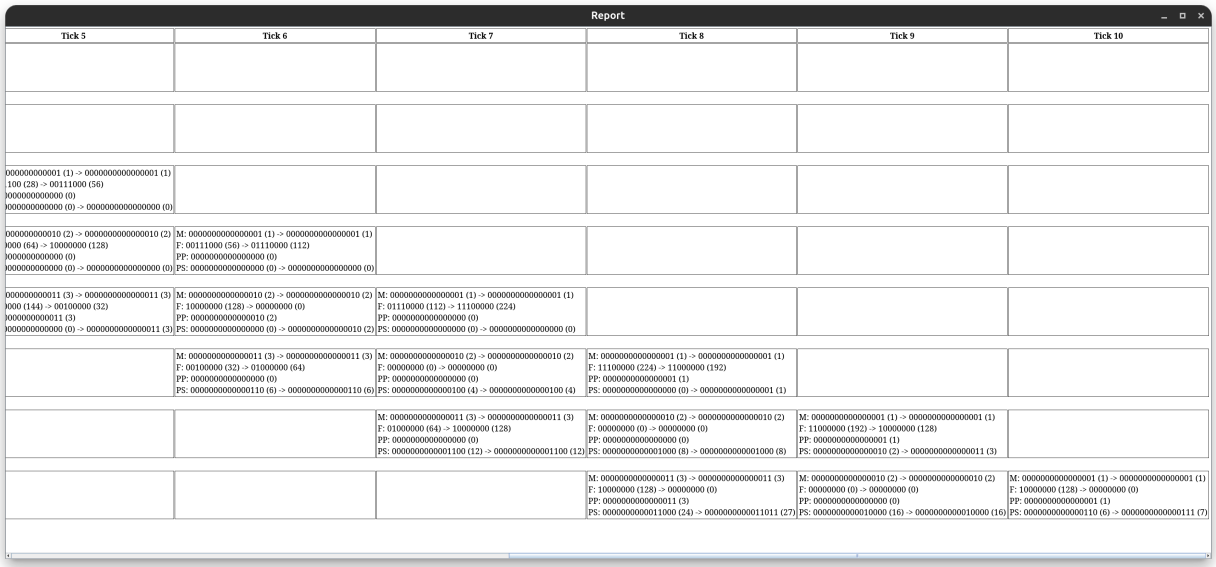
1.3.3 Тест 3

Входной файл input3.json:

```
[
    [1, 7],
    [2, 8],
    [3, 9]
]
```

Результат работы программы для файла input3.json:





1.4 Графики

Обозначения:

n – количество этапов конвейера.

r – ранг задачи (количество обрабатываемых 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.

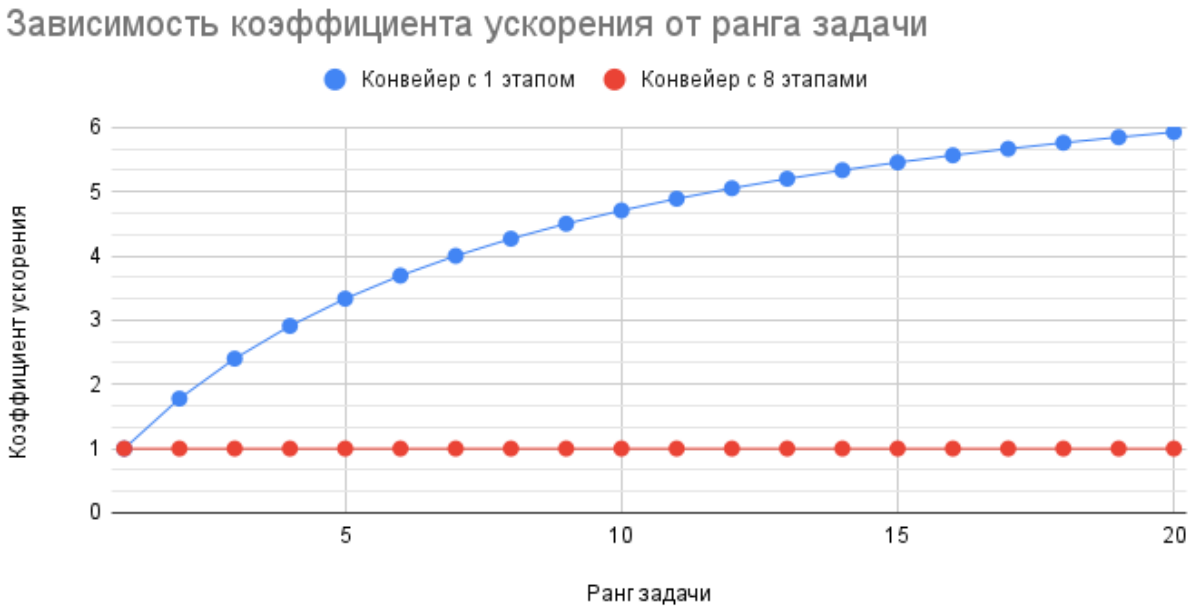


Рисунок 1.1 – График зависимости коэффициента ускорения от ранга задачи для конвейеров с различным количеством этапов

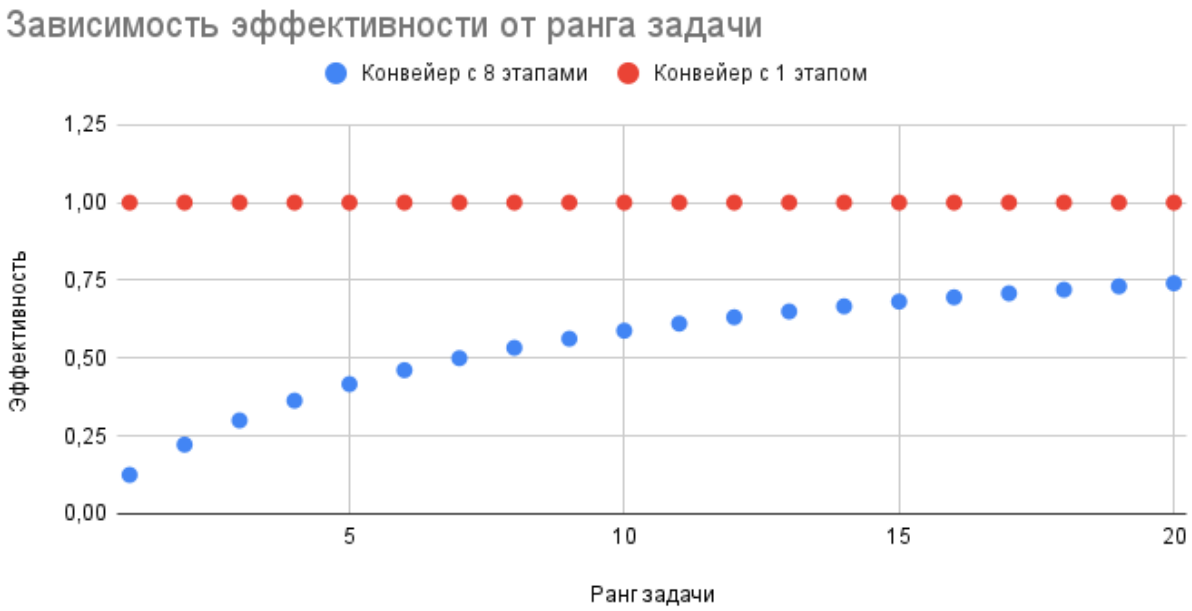


Рисунок 1.2 – График зависимости эффективности от ранга задачи для конвейеров с различным количеством этапов

1.5 Ответы на контрольные вопросы

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

Доказательство корректной работы модели находится в разделе 1.3.

1.5.2 Объясните на графиках точки перегиба и асимптоты

Для графиков коэффициента ускорения рассмотрим два случая.

Поскольку $K_y = \frac{r \times n}{n+r-1}$, то, найдя для него предел при $r \to \infty$: $\lim\limits_{r\to\infty} \frac{r\times n}{n+r-1} = n$, мы поймем, что в данном случае коэффициент ускорения ограничен числом n (количеством этапов конвейера). Это объясняется тем, что при решении задачи выского ранга, мы ограничены количеством вычислительных блоков.

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

Для объяснения асимптот **графика эффективности** действуем аналогично:

Поскольку $e = \frac{r}{n+r-1}$, то при $r \to \infty$ предел принимает значение $\lim\limits_{r\to\infty} \frac{r}{n+r-1} = 1$. 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 $\lim\limits_{n\to\infty} \frac{r}{n+r-1} = 0$. В этом случае ситуация обратная: из-за того, что количество вычислительных блоков гораздо 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) растёт. При увеличении n значение эффективности e(n) снижается. Причины данного поведения объяснены в ответе на предыдущий вопрос.

1.5.4 Каково соотношение между параметрами n, r, m, p модели сбалансированного конвейера?

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

p - разрядность умножаемых попарно чисел (для данного варианта $p = 4$)

n - количество процессорных элементов, $n = p = 4$;

1.5.5 Каким будет соотношение между $\mathbf{r}_1$ и $\mathbf{r}_2$ характеристики h, если для неё выполняется $\mathbf{h(n_1, r_1) = h(n_2, r_2)n_1 > n_2}$?

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

В качестве характеристики h возьмём 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:

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

$$\begin{cases} \frac{r_1}{n_1 + r_1 - 1} = \frac{r_2}{n_2 + r_2 - 1} \\ n_1 > n_2 \end{cases} \Leftrightarrow$$

$$\Leftrightarrow \begin{cases} r_1 n_2 + r_1 r_2 - r_1 = r_2 n_1 + r_1 r_2 - r_2 \\ n_1 > n_2 \end{cases} \Leftrightarrow$$

$$\Leftrightarrow \begin{cases} r_1(n_2 - 1) = r_2(n_1 - 1) \\ n_1 > n_2 \end{cases} \Leftrightarrow$$

$$\Leftrightarrow \begin{cases} \frac{r_2}{r_1} = \frac{n_2 - 1}{n_1 - 1} \\ n_1 > n_2 \end{cases} \Leftrightarrow$$

$$\Leftrightarrow \begin{cases} r_1 > r_2 \\ n_1 > n_2 \end{cases}$$

Ответ: $\mathrm{r}_1 > r_2$.

1.6 Задачи

Дано: несбалансированный конвейер (заданы конкретные значения: n , $\{t_i\}$ - времена выполнения обработки на этапах конвейера, e0 - 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 неё значение параметров)

$$\begin{cases} e(n,r_0) = \frac{K_y(n,r_0)}{n}(1) \\ K_y(n,r_0) = \frac{T_1}{T_n} \\ T_n = \sum\limits_{i=1}^{n} t_i + (r_0 - 1)t_{max} \\ T_1 = r_0 \sum\limits_{i=1}^{n} \\ t_i > 0 \\ r_0, n \in N \end{cases} \Longrightarrow$$

$$\Longrightarrow \begin{cases} K_y(n,r_0) = \dfrac{r_0 \sum\limits_{i=1}^{n}}{\sum\limits_{i=1}^{n} t_i + (r_0 - 1)t_{max}} \\ t_i > 0 \\ r_0, n \in N \end{cases} \quad (1.1)$$

Подставим полученную формулу коэффициента ускорения в (1.1):

$$\begin{cases} e(n,r_0) = \frac{K_y(n,r_0}{n} = \dfrac{r_0 \sum\limits_{i=1}^{n}}{n\left(\sum\limits_{i=1}^{n} t_i + (r_0 - 1)t_{max}\right)}(2) \\ t_i > 0 \\ r_0, n \in N \end{cases} \quad (1.2)$$

Подставим полученную формулу эффективности в исходное неравенство:

$$\begin{cases} \frac{K_y(n,r_0}{n} = \dfrac{r_0 \sum\limits_{i=1}^{n}}{n\left(\sum\limits_{i=1}^{n} t_i + (r_0 - 1)t_{max}\right)} > e_0(3) \\ t_i > 0 \\ r_0, n \in N \end{cases} \quad (1.3)$$

Выразим r0:

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

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

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

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

Получим:

$$\begin{cases} r_0 > \dfrac{e_0 n \left(\sum\limits_{i=1}^{n} t_i - t_{max} \right)}{\sum\limits_{i=1}^{n} t_i - e_0 n t_{max}} t_i > 0 \\ r_0, n \in N \end{cases}$$

Так как для любого конвейера: $\sum_{i=1}^{n} t_i - t_{max} \geq 0$

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

$$\begin{cases} r_0 > \dfrac{e_0 n \left(\sum\limits_{i=1}^{n} t_i - t_{max} \right)}{\sum\limits_{i=1}^{n} t_i - e_0 n t_{max}} \\ r_0 \geq 1, t_i \geq 1, n \geq 1, \\ t_{max} \geq t_i, e_0 > 0 \end{cases}$$

все условия выполняются.

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

$$\begin{cases} r_0 < \dfrac{e_0 n\left(\sum\limits_{i=1}^{n} t_i - t_{max}\right)}{\sum\limits_{i=1}^{n} t_i - e_0 n t_{max}} \\ r_0 \geq 1, t_i \geq 1, n \geq 1, \\ t_{max} \geq t_i, e_0 > 0 \end{cases}$$

так как в числителе стоит положительное число, в знаменателе отрицательное, результат выражения принимает отрицательные значения. Но по условию r0 может принимать только значения $r_0 \geq 1$. Следовательно, данный случай не войдёт в итоговое решение.

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

$$\begin{cases} r_0 < \dfrac{e_0 n\left(\sum\limits_{i=1}^{n} t_i - t_{max}\right)}{\sum\limits_{i=1}^{n} t_i - e_0 n t_{max}} \\ r_0 \geq 1, t_i \geq 1, n \geq 1, \\ t_{max} \geq t_i, e_0 > 0 \end{cases}$$

знаменатель выражения будет равен нулю. А в арифметике деление на ноль не имеет смысла. Данный случай также не войдёт в итоговое решение.

Ответ: $r_0 > \dfrac{e_0 n\left(\sum\limits_{i=1}^{n} t_i - t_{max}\right)}{\sum\limits_{i=1}^{n} t_i - e_0 n t_{max}}$ при $\sum\limits_{i=1}^{n} t_i - e_0 n t_{max} > 0$

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

Воспользуемся неравенством (1.3) и преобразуем его в следующее неравенство:

$$\frac{r_0 \sum_{i=1}^{n} t_i}{n\left(\sum_{i=1}^{n} t_i + (r_0 - 1)t_{max}\right)} > \frac{r_0 \sum_{i=1}^{n_0} t_i}{n_0\left(\sum_{i=1}^{n} t_i + (r_0 - 1)t_{max}\right)}$$

Учитывая, что

$$\begin{cases} n_0\left(\sum_{i=1}^{n} t_i + (r_0 - 1)t_{max}\right) > 0 \\ \sum_{i=1}^{n} t_i > 0 \\ \sum_{i=1}^{n} t_i = \sum_{i=1}^{n_0} t_i \\ r_0, n \in N \end{cases}$$

получаем $\frac{1}{n} > \frac{1}{n_0}, n < n_0$. Это означает, что для выполнения заданного условия необходимо, чтобы n в перестроенном конвейере было меньше, чем n_0 в конвейере до перестроения. При этом необходимо, чтобы при объединение этапов максимальное время этапа на перестроенном конвейере было равно максимальному времени этапа на конвейере до перестроения.

Необходимо объединять этапы конвейера таким образом, чтобы выполнялось неравенство

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

Следует учесть, что объединять мы можем только соседние этапы. Приведём примеры решения такой задачи на конкретных данных:

Пример 1. Пусть:

$$r_0 = 3, n_0 = 8, t_i = \{1, 1, 2, 2, 3, 3, 4, 4\}$$

Тогда $e_0 = 0,27$ (по формуле (1.2)). Перестроим конвейер следующим образом:

$$r_0 = 3, n_0 = 4, t_i = \{2, 4, 6, 8\}$$

(объединены: 1-й со 2-м, 3-й с 4-м, 5-й с 6-м, 7-й с 8-м), тогда $e = 0,42. e > e_0$, что соответствует условию задачи.

Пример 2. Пусть:

$$r_0 = 5, n_0 = 6, t_i = \{6, 1, 4, 3, 2, 5\}$$

Вычислим эффективность получившегося конвейера:

$$e_0 = \frac{5 * 21}{6 * (21 + (5 - 1) * 6} = 0,389$$

Перестроим конвейер:

$$r_0 = 5, n_0 = 4, t_i = \{6, 5, 5, 5\}$$

Пересчитаем эффективность: $e(n, r) = \frac{521}{4(21+(51)6} = 0,583$. Неравенство e>$e_0$ также выполняется.

Ответ: $1 \leq n < \dfrac{r_0 \sum_{i=1}^{n} t_i}{e_0 \left(\sum_{i=1}^{n} t_i + (r_0 - 1) t_{max} \right)}$

1.6.3 Каким образом можно перестроить несбалансированный конвейер, чтобы добиться максимальной скорости работы? 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 $K_y(n, r), e(n, r)$?

анный конвейер необходимо перестроить так, чтобы он был сбалансированным и каждый этап выполнялся за минимальный квант времени t_0, т.е. разделить этапы конвейера, которые длятся дольше, чем t_0, на более мелкие этапы, которые будут длиться t_0. Количество этапов в этом сбалансированном конвейере будет равняться:

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

Тогда:

$$K_y(n,r) = \frac{T_1}{T_n} = \frac{\frac{\sum_{i=1}^{n} t_i}{T_0} rt}{\left(\frac{\sum_{i=1}^{n} t_i}{t_0} + r - 1 \right)} t_0 = \frac{\frac{\sum_{i=1}^{n} t_i}{t_0} r}{\frac{\sum_{i=1}^{n} t_i}{t_0} + r - 1}$$

$$e(n,r) = \frac{K_y}{n} = \frac{\frac{\sum_{i=1}^{n} t_i}{t_0} r}{\left(\frac{\sum_{i=1}^{n} t_i}{t_0} + r - 1 \right) \frac{\sum_{i=1}^{n} t_i}{t_0}} = \frac{r}{\frac{\sum_{i=1}^{n} t_i}{t_0} + r - 1}$$

Ответ: $K_y(n,r) = \frac{\frac{\sum_{i=1}^{n} t_i}{t_0} r}{\frac{\sum_{i=1}^{n} t_i}{t_0} + r - 1}, e(n,r) = \frac{r}{\frac{\sum_{i=1}^{n} t_i}{t_0} + r - 1}$

2 ВЫВОД

В процессе выполнения лабораторной работы, были получены навыки реализации моделей решения задач на конвейерной архитектуре. Была исследована и реализована модель сбалансированного конвейера для вычисления произведений пар 8-разрядных чисел умножением со старших разрядов со сдвигом частичной суммы влево. Были изучены различные числовые характеристики конвейерной архитектуры, а именно коэффициент ускорения K_y и эффективность e.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

[1] Ивашенко., В. П. Модели решения задач в интеллектуальных системах. В 2 ч. Ч. 1 : М74 Формальные модели обработки информации и параллельные модели решения задач : учеб.-метод. пособие / В. П. Ивашенко. — 2020.