

**Задание к практическому занятию**  
**«Исследование принципов функционирования скалярных и векторных процессоров»**  
**по курсу «Вычислительные системы»**  
**для студентов групп А-7,8-17**

**Цель:** исследовать процесс выполнения на скалярном и векторном процессоре с двухпортовыми векторными регистрами циклического фрагмента программы

*For I=1 to N DO*

$A(I)=F(X(I),Y(I),P(I),Q(I),W(I),Z(I),V(I),L(I),U(I)).$

Функция  $F$  берется в соответствии с таблицей 4 индивидуального задания, количество итераций  $N$  меняется в зависимости от пунктов расчетного задания.

В обоих процессорах использовано одинаковое АЛУ, времена выполнения операций в котором приведены в таблице 1. Память команд и операндов в процессорах разделена. Память команд имеет 1 модуль, а память операндов – 4 модуля.

В скалярном процессоре имеется конвейер команд, стадии и времена преобразований команд которого приведены в таблице 2 (подготовительный этап выполнения цикла DO – 2 машинных такта).

В векторном процессоре используется конвейер команд, стадии и времена преобразования векторных команд которого приведены в таблице 3.

Таблица 1

Времена выполнения операций в АЛУ

Операции	Машинные такты
Сложение	1
Вычитание	1
Умножение	3
Деление	4
Считывание, запись операндов	1
Приращение, проверка условия перехода	2

Таблица 2

Стадии и времена преобразования команд в скалярном процессоре

Стадии	Машинные такты
Выборка – $B$	1
Декодирование – $D$	1
Преобразование виртуальных адресов в физические – $A$	1
Считывание операндов – $F$	1
Выполнение операций – $E$	см. таблицу 1

При выполнении задания используются процессоры следующей организации.

1. **Суперскалярный процессор** имеет два конвейера команд, два АЛУ, каждое из которых состоит из автономно работающих сумматора, умножителя и делителя, и одно устройство обращения к памяти, позволяющее выполнять операции записи (считывания) параллельно с работой АЛУ. Имеется узел предсказания переходов, общий файл регистров. Используется память с расслоением, поэтому этапы  $F$ ,  $E$  (считывания/записи) идут с задержкой на 1 такт по отношению друг к другу.

Стадии и времена преобразования векторных команд

Стадии	Машинные такты
Выборка – $B$	1
Проверка на бесконфликтность – $H$	1
Выдача команд – $K$	1
Декодирование – $D$	1
Подготовка параметров – $G$	2
Настройка конвейера – $L$	2
Выполнение операции – $E$	см. таблицу 1

**2. Векторный процессор** имеет один конвейер команд, **8 двухпортовых** векторных регистров, следующий набор исполнительных устройств: **2 сумматора**, 1 умножитель, 1 делитель. Загрузка векторных регистров может осуществляться параллельно с выполнением арифметических операций, но не связанных с загружаемым регистром.

Вариант задания выбирается **из таблицы 4 по номеру студента в журнале.**

**Выполнение работы состоит из следующих пунктов.**

**1.** Для каждого типа процессора представить исследуемый фрагмент программы в виде последовательности команд этого процессора, показать, что полученный ассемблерный код является "оптимальным" для данного типа процессора.

**2. Для суперскалярного процессора**

а) построить временную диаграмму занятости устройств и регистров при выполнении программы при  $N=10$ ;

б) построить зависимость  $T_{\text{вып}} = f(N)$ , при  $N=1 \div 20$ ,

где  $T_{\text{вып}}$  – время выполнения программы;  $N$  – количество итераций.

**3. Для векторного процессора с двухпортовыми регистрами**

а) построить временную диаграмму занятости устройств и регистров при выполнении заданного фрагмента программы для  $N=10$ ;

б) анализируя построенную временную диаграмму, определить значения  $N$  ( $N=1 \div 20$ ), при которых меняется характер зависимости  $T_{\text{вып}}(N)$ . Минимальное и максимальное значения  $N$ , при которых меняется характер зависимости, назовем *предельными* значениями  $N$ ;

в) построить временные диаграммы занятости устройств и регистров при выполнении заданного фрагмента программы **для двух предельных** значений  $N$ ; показать правильность определения предельных значений  $N$ ;

г) построить зависимость  $T_{\text{вып}} = f(N)$ , при  $N=1 \div 20$ ;

Все зависимости  $T_{\text{вып}} = f(N)$  показать в виде формул и графиков. Для сравнения различных организаций процессоров зависимости  $T_{\text{вып}} = f(N)$  построить на одних графиках.

## Варианты расчетных заданий для студентов группы А-07-17

№	Текст задания	Подпись
1	$A(i)=L(i)/(Z(i)*W(i)-X(i)) + (Y(i)-V(i))*Q(i) + P(i)/U(i)$	
2	$A(i) = (Z(i)-X(i))/Y(i) + V(i)*Q(i) - W(i)/(P(i)+Z(i)*U(i))$	
3	$A(i)= Q(i)/(X(i) - Y(i)) + V(i)*U(i) - W(i)/(L(i) + P(i)*Z(i))$	
4	$A(i) = (Q(i)+X(i))*Y(i) - (P(i)*U(i)-W(i))/V(i)+L(i)/Z(i)$	
5	$A(i) = (L(i)-X(i))*Y(i) + V(i)/Q(i) + W(i)*(P(i)-Z(i)/U(i))$	
6	$A(i) = P(i)/(L(i)+V(i)) - (X(i)*Z(i)+Y(i))/Q(i) + U(i)*W(i)$	
7	$A(i)=U(i)*(X(i)+Y(i)) - (V(i)/W(i)+Q(i))/L(i) - P(i)*Z(i)$	
8	$A(i)=Z(i)/(U(i)+V(i)*Y(i)) - X(i)/(Q(i)+L(i)) - P(i)*W(i)$	
9	$A(i) = (Q(i)+X(i))*Y(i) - (V(i)*W(i)-L(i))/P(i)+Z(i)/U(i)$	
10	$A(i)=Q(i)/(Z(i)+W(i)) - (X(i)*U(i)+Y(i))/V(i) + P(i)*L(i)$	
11	$A(i) = (Y(i)+X(i)*Q(i))/(V(i) - W(i)*(U(i)+L(i)) - P(i)/Z(i)$	
12	$A(i)=V(i)*Y(i) - X(i)/W(i) + (P(i)+Q(i))*Z(i) - U(i)/L(i)$	
13	$A(i)=P(i)*U(i) + (X(i)-Y(i))/V(i) + Q(i)/(L(i)+W(i)*Z(i))$	
14	$A(i) = Y(i)/U(i) + (X(i)-W(i))*V(i) - Q(i)*(L(i)/Z(i) + P(i))$	
15	$A(i) = X(i)/(L(i)+Y(i)) - V(i)*Q(i) + W(i)/(U(i))+P(i)*Z(i))$	
16	$A(i)=(P(i)+X(i)*U(i))/Y(i) - V(i)*(Q(i)-L(i))+W(i)/Z(i)$	
17	$A(i)=(Z(i)-X(i))/Y(i) + V(i)/(U(i)+Q(i)*L(i)) - P(i)*W(i)$	
18	$A(i) = U(i)/(X(i)+Y(i)) - V(i)*Q(i) + L(i)*(P(i) + Z(i)/W(i))$	
19	$A(i) = Y(i)/U(i) + X(i)*(W(i)-V(i)*Q(i)) + L(i)/(P(i)+Z(i))$	
20	$A(i) = L(i)/(X(i)-Y(i)) + V(i)*Q(i) + W(i)/(U(i)+P(i)*Z(i))$	
21		

## Варианты расчетных заданий для студентов группы А-08-17

№	Текст задания	Подпись
1	$A(i) = (L(i) + X(i) * Z(i)) / Y(i) - Q(i) * (U(i) - W(i)) + P(i) / V(i)$	
2	$A(i) = (P(i) - X(i)) * (V(i) + Q(i)) - U(i) / W(i) + Z(i) / (Y(i) - L(i))$	
3	$A(i) = X(i) / (L(i) * V(i) - U(i)) + (Y(i) - Q(i)) * P(i) + W(i) / Z(i)$	
4	$A(i) = X(i) * Z(i) + (V(i) - Y(i)) / Q(i) + P(i) / (W(i) + U(i) * L(i))$	
5	$A(i) = (Z(i) + X(i) * Y(i)) / (Q(i) - U(i) * (V(i) + W(i))) - P(i) / L(i)$	
6	$A(i) = Z(i) / (X(i) - V(i)) + Q(i) * U(i) + W(i) / (Y(i) + P(i) * L(i))$	
7	$A(i) = U(i) * Z(i) + X(i) / (V(i) - Q(i)) + Y(i) / (W(i) - P(i) * L(i))$	
8	$A(i) = (Y(i) - X(i)) * (Z(i) + V(i)) - Q(i) / L(i) + U(i) / (P(i) - W(i))$	
9	$A(i) = Y(i) / (Z(i) - X(i) * P(i)) + Q(i) / (U(i) - W(i)) + P(i) * L(i)$	
10	$A(i) = (V(i) + X(i)) / U(i) - Z(i) * (P(i) - Y(i) / (W(i)) + Q(i) * L(i)$	
11	$A(i) = P(i) / Z(i) + (L(i) - V(i)) * X(i) - Y(i) / (Q(i) - U(i) * W(i))$	
12	$A(i) = (Q(i) - Y(i)) / (X(i) * W(i)) + P(i) / (U(i) - Z(i)) - V(i) * L(i)$	
13	$A(i) = X(i) / (Z(i) + W(i)) - (P(i) * U(i) + Y(i)) / V(i) + Q(i) * L(i)$	
14	$A(i) = (Y(i) * Z(i) - X(i)) / V(i) + (Q(i) - U(i)) / W(i) + P(i) * L(i)$	
15	$A(i) = Y(i) / U(i) + (Z(i) - W(i)) * X(i) - P(i) / (V(i) - Q(i) * L(i))$	
16	$A(i) = P(i) * (X(i) + Y(i)) - (Q(i) / Z(i) + U(i)) / W(i) - V(i) * L(i)$	
17	$A(i) = (W(i) - X(i)) / Y(i) + Q(i) / (Z(i) + U(i) * L(i)) - P(i) * V(i)$	
18	$A(i) = Z(i) / (X(i) - Y(i)) + Q(i) * V(i) - U(i) / (W(i) + P(i) * L(i))$	
19	$A(i) = (Z(i) + X(i)) / (W(i) * P(i)) - Q(i) / Y(i) + U(i) * (V(i) + L(i))$	
20	$A(i) = V(i) / Z(i) + (X(i) - Y(i)) * Q(i) - U(i) * (W(i) / P(i) + L(i))$	
21	$A(i) = W(i) / Z(i) + X(i) * (Y(i) - Q(i) * U(i)) + V(i) / (P(i) + L(i))$	
22	$A(i) = (Z(i) - X(i)) / Y(i) + Q(i) * U(i) - W(i) / (P(i) + L(i) * V(i))$	
23	$A(i) = W(i) * X(i) - (V(i) + P(i)) / Z(i) - Y(i) / U(i) - Q(i) * L(i)$	
24	$A(i) = U(i) / (X(i) + Y(i)) - Q(i) * V(i) + W(i) * (P(i) + L(i) / Z(i))$	
25	$A(i) = L(i) * (X(i) + Q(i)) - (P(i) + W(i)) / U(i) + V(i) / (Y(i) - Z(i))$	