

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ  
Государственное образовательное учреждение высшего профессионального образования  
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ  
ПРЕПОДАВАТЕЛЬ

Ассистент

А.Н.Долидзе

\_\_\_\_\_  
должность, уч.степень,звание

\_\_\_\_\_  
подпись, дата

\_\_\_\_\_  
инициалы,фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №4

**«Проект архитектуры набора команд и микроархитектуры  
простейшего 8-битного процессора»**

по курсу: Организация ЭВМ и систем.

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР.

\_\_\_\_\_  
4143

\_\_\_\_\_  
подпись, дата

\_\_\_\_\_  
Е.Д.Тегай  
инициалы,фамилия

## Текст задания

### 1) Исходный алгоритм

Вариант исходного алгоритма №4 продемонстрирован на рисунке 1.

4	Умножение целых чисел со знаком в	С коррекцией результата
---	-----------------------------------	-------------------------



5	дополнительном коде со сдвигом суммы частичных произведений вправо,	С предварительным изменением знака
6	неподвижным множимым и анализом множителя, начиная с младших разрядов.	С преобразованием множителя

Рисунок 1 – Исходный алгоритм

### 2) Вид архитектуры

Гарвардская (команды и данные в разных ЗУ):

$$19 \bmod 2 + 1 = 2$$

### 3) Тип и формат данных

Знаковый (дополнительный код)

### 4) Операции АЛУ

1. Операция из ЛР1: умножение

2. Логическая операция 1: LSR

$$19 \bmod 12 + 1 = 8$$

3. Логическая операция 2: NOT

$$(19 + 5) \bmod 12 + 1 = 1$$

4. Логическая операция 2: AND

$$(19 + 7) \bmod 12 + 1 = 2$$

**5) Число регистров общего назначения**

$$19 \bmod 13 + 4 = 10$$

**6) Пересылка**

Mov (пересылка данных: память-память, память-регистр, регистр-регистр, регистр-память, а также запись константы).

$$19 \bmod 3 + 1 = 2$$

**7) Переход по условию, в соответствии с операцией сравнения**

<

$$19 \bmod 6 + 1 = 2$$

**8) Способы адресации**

Первый: прямая регистровая

Второй: адресация по смещению

$$19 \bmod 7 + 1 = 6$$

Третий: адресация с автоувеличением (простая косвенная с автоувеличением)

$$(19 \bmod 7 + 5) \bmod 7 + 1 = 4$$

**Система команд**

Составим список команд, присвоим им код операции, определим количество операндов в каждой команде и формируемые командой флаги. Искомый набор команд представлен в таблице 1.

Таблица 1

Код	Команда	Свойство	Кол. Оп.	Флаги
0000	-	-	-	-
0001	MUL	Умножение	2	-
0010	NOT	Лог.отрицание	1	Z N V
0011	AND	Лог.умножение	2	Z N
0100	LSR	Лог. сдвиг влево	2	Z C N
0101	CMP	Сравнение	2	Z N C V
0110	MOV	Пересылка данных	2	-
0111	BNE	Переход, если не равно	1	Z
1000	B	Безусловный переход	1	-
1001	BEQ	Переход, если равно	1	Z
1010	JMP	Безусловный переход по адресу	1	-
1011	#	Число	1	-

Рассмотрим таблицу подробнее. Кодировка команд производилась на основании обычного счёта от 0 до некоторого числа в формате четырёх бит.

Кодировка режимов адресации проводилась по той же аналогии, отчего

получаем, что:

00 – прямая регистровая

01 – адресация с автоувеличением

10 – адресация по смещению

Согласно варианту используется 10 регистров. Их кодировка приведена ниже (здесь уже используется восьмибитная запись).

00 – R0

01 – R1

02 – R2

03 – R3

04 – R4

05 – R5

06 – R6

07 – R7

08 – R8

09 – R9

0A – R10

Приведём примеры кодировки режимов адресации. Результаты приведены в таблице 2.

Таблица 2 – Примеры кодировки режимов адресации

Мnemonic код	Машинный код	Комментарий
MUL R3, R4	01 03 04 0000 0001 (КОП) 0000 0011 (1 оп) 0000 0100 (2 оп)	Режим адресации кодируется в двух старших битах операнда. Прямая адресация
MOV (R10)+, R5	06 4A 05 0000 0110 (КОП) 0100 1010 (1 оп) 0000 0101 (2 оп)	Режим адресации кодируется в двух старших битах операнда. Адресация с автоувеличением
AND R1, [R2, #08]	03 01 82 08 0000 0011 (КОП) 0000 0001 (1 оп) 1000 0010 (2 оп) 0000 1000 (смещ)	Режим адресации кодируется в двух старших битах операнда. Адресация по смещению

Далее необходимо определить функции АЛУ. АЛУ выполняет все арифметические и логические операции над данными, включая операции сдвига. Результат определения функций АЛУ продемонстрирован в таблице 3.

Таблица 3 – Определение функций АЛУ

Команда	Функция АЛУ	Код функции	Комментарий
MUL	Умножение	0001	
NOT	Лог. отрицание	0010	
AND	Лог. Умножение	0011	
LSR	Лог. Сдвиг влево	0100	Функция УУ
CMP	Сравнение	0101	
JMP	Безусловный переход по адресу	1010	Функция УУ
BNE	Переход, если не равно	0111	Функция УУ
B	Безусловный переход	1000	Функция УУ
BEQ	Переход, если равно	1001	Функция УУ

Размер регистровой памяти для разрабатываемого 8-битного микропроцессора составляет 8 бит, что соответствует 1 байту. Эти регистры используются для хранения данных, адресов оперативной памяти, промежуточных результатов вычислений и другой важной информации во время выполнения программы.

Оперативная память в данном контексте может быть описана как адресуемая по байту, где каждая ячейка памяти имеет размер 8 бит. Размер оперативной памяти определяется количеством доступных адресов, и для 8-битного микропроцессора это может быть ограничено до  $2^8 = 256$  байт (или 64 Кб).

# Схема Гарвардской микроархитектуры

Искомая схема продемонстрирована на рисунке 2.

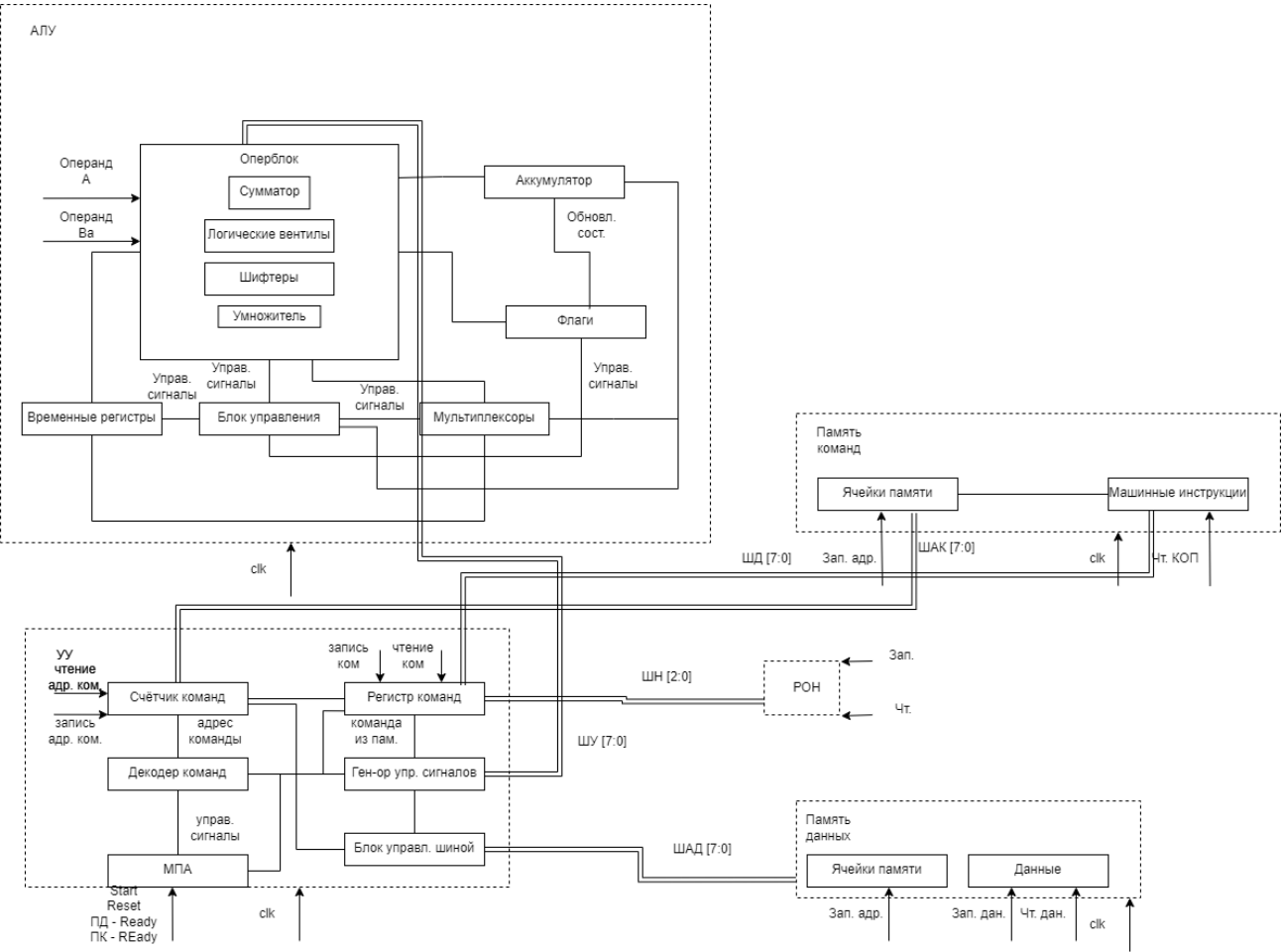


Рисунок 2 – Схема Гарвардской микроархитектуры

Выпишем в таблицу 4 все входные/выходные сигналы.

Таблица 4 – Сигналы (считывание команды из памяти)

Сигнал	Тип	Описание
УУ		
Start	I	Управ. сигнал, иниц. начало выполнения операции или процесса
Reset	I	Управ. сигнал, иниц. сброс сост. у-ва до нач. или опред. Сост.
ПД - Ready	I	Команда в ПД выполнена



ПК - ready	I	Команда в ПК выполнена
Чтение адр. ком.	I	Управ.сигнал, указ. на то, что МП готов проч. адрес след. команды из пам.
Запись адр. ком.	I	Управ.сигнал, кот. указ. на необход. зап. адр. след. ком. в пам.
Чтение ком.	I	Управ. сигнал, кот. указ. на готовн. мп счит. ком. из пам.
Запись ком.	I	Управ. сигнал, кот. указ. на необход. зап. новой ком. в пам. у-ва
Управ. сигналы	I/O	Сигналы, кот. управ. работой у-ва или сист. использ для синхрон. операций
clk	I	Управ. сигнал, кот. опред. такт. част. работы у-ва
ШАК	I/O	Шина адреса и команд. Управ. шина, кот. использ. для передачи адр. и ком. между компонентами сист.
ШД	I/O	Шина данных. Управ. шина, кот. использ. для передачи данных между различн. компонентами сист.
ШАД	I/O	Шина адреса данных. Шина, кот. использ. для передачи адрю пам. и самих данных между компонентами сист.
ШУ	I/O	Шина управления. Шина, кот. использ. для передачи управл. сигн. между компонентами сист.
ШН	I	Шина номера (регистрового). Шина, кот. использ. для передачи номера рег. или адр. рег. внутри процессора
АЛУ		
Операнд А	I	Сигнал, представ. значение операнда А
Операнд В	I	Сигнал, представ. значение операнда В
Запись адр. ком.	I	Управ.сигнал, кот. указ. на необход. зап. адр. след. ком. в пам.
Управ. сигналы	I/O	Сигналы, кот. управ. работой у-ва или сист. использ для синхрон. операций

Обнов. Сост.	I	Сигн, указ. на необход. измен. сост. у-ва или его компонентов
ШУ	I/O	Шина управления. Шина, кот. использ. для передачи управл. сигн. между компонентами сист.
clk	I	Управ. сигнал, кот. опред. такт. част. работы у-ва
Память команд		
Запись адр. ком.	I	Управ.сигнал, кот. указ. на необход. зап. адр. след. ком. в пам.
Чтение КОП	I	Управ. сигнал, кот. указ. на готовн. мп счит. ком. из пам.
clk	I	Управ. сигнал, кот. опред. такт. част. работы у-ва
ШД	I/O	Шина данных. Управ. шина, кот. использ. для передачи данных между различн. компонентами сист.
ШАК	I/O	Шина адреса и команд. Управ. шина, кот. использ. для передачи адр. и ком. между компонентами сист.
РОН		
Зап.	I	Управ. сигнал, который задаёт разреш. на запись
Чт.	I	Управ. сигнал, который задаёт разреш. на чтение
ШН	I	Шина номера (регистрового). Шина, кот. использ. для передачи номера рег. или адр. рег. внутри процессора
Память данных		
clk	I	Управ. сигнал, кот. опред. такт. част. работы у-ва
Запись адр.	I	Управ.сигнал, кот. указ. на необход. зап. адр. дан. в пам.
Запись дан.	I	Управ.сигнал, кот. указ. на необход. зап. дан. в пам.
Чт. дан.	I	Управ.сигнал, кот. указ. на необход. прочит. дан. из пам.

ШАД	I/O	Шина адреса данных. Шина, кот. использ. для передачи адрю пам. и самих данных между компонентами сист.
-----	-----	--

### **Детальное описание функционирования спроектированной архитектуры**

В самом начале работы подаётся стартовый сигнал Start. По этому сигналу УУ начинает свою работу. Он вычисляет адрес команды и подаёт этот адрес на шину адреса команд. УУ генерирует этот сигнал, используя текущий адрес счётчика команд (который потом увеличится). Полученная команда загружается в регистр команд для дальнейшей декодировки. Декодер команд анализирует загруженную команду, определяет тип операции и необходимые операнды. Генератор управляющих сигналов формирует необходимые управляющие сигналы для выполнения команды в оперблоке.

Этот процесс можно разделить на несколько этапов. Сначала идёт получение операндов, затем (по необходимости) могут быть использованы временные регистры в АЛУ для хранения операндов до начала операции. После этого идёт настройка мультиплексоров. Они настраиваются для выбора соответствующих операндов для текущей операции. А после мультиплексоры направляют выбранные операнды на входы оперблока. Генератор управляющих сигналов посылает необходимые сигналы в оперблок для выполнения требуемой операции.

Если команда требует выполнения арифметической операции (сложение, вычитание – например, при сравнении), операнды передаются в сумматор. Для логических операций операнды передаются в логические вентилы, для операций сдвига операнд передаётся в шифтер, а для умножения операнды передаются в умножитель.

После этого идёт обработка результата. Он формируется на выходе АЛУ. Результат может быть записать обратно в РОН. Если используется аккумулятор,

то результат может быть сохранён в аккумуляторе для использования в последующих операциях. Далее идёт обновление флагов состояния в зависимости от результата операции. После завершения операции оперблок посылает сигнал завершения обратно в УУ. А УУ обновляет счётчик команд, чтобы указать на следующую команду в памяти команд.

Следует помнить, что операции могут быть описаны различными способами адресации. Например, если идёт прямая регистровая адресация, то операнды находятся в регистрах и адресуются непосредственно.

В этом случае сначала идёт передача команды в АЛУ. После декодирования УУ определяет, что операнды находятся в регистрах. Затем определяются регистры (их номер). Далее УУ инициирует чтение значений из РОН, указанных в команде. Значения потом передаются в оперблок. Далее идут уже рассмотренные действия. Затем УУ инициирует запись результата обратно в целевой регистр.

Процесс выполнения команды с использованием адресации по смещению немного сложнее, чем прямая регистровая, так как она включает в себя вычисление адреса перед доступом к операндам. После декодирования команды идёт определение базового регистра и смещения. Затем вычисляется адрес. УУ инициирует чтение значения базового регистра. АЛУ вычисляет адрес, складывая значение базового регистра и смещение. Полученный адрес передаётся на адресную шину для доступа к памяти данных и так далее.

Если операция использует адресацию с автоувеличением, то это означает, что операнд находится по адресу, хранящемуся в регистре, и после чтения или записи значения указатель в этом регистре автоматически увеличивается. После декодирования команды идёт определение регистра с указателем (адрес операнда). УУ инициирует чтение значения регистра, который содержит адрес операнда. Значение адреса передаётся на адресную шину для доступа к памяти

данных. По указанному адресу происходит доступ к памяти данных. Значение передаётся в АЛУ как операнд. Значение регистра-указателя автоматически увеличивается на величину, равную размеру операнда и так далее.