

Министерство Образования и Науки Украины
Севастопольский Национальный Технический Университет

Методические указания
к выполнению курсового проекта
по дисциплине «Программирование»
для студентов дневной формы обучения направления
0915- компьютерная инженерия
Варианты заданий

Севастополь
2002

Цель методических указаний - обеспечить эффективную работу студентов при выполнении курсового проекта по дисциплине «Программирование».

Методические указания утверждены на заседании кафедры кибернетики и вычислительной техники «____» _____ 2002 г. Протокол N ____

Рецензент: доцент Кожаев Е.А.

Методические указания составили: доцент Васильченко А.К., доцент Сергеев Г.Г.

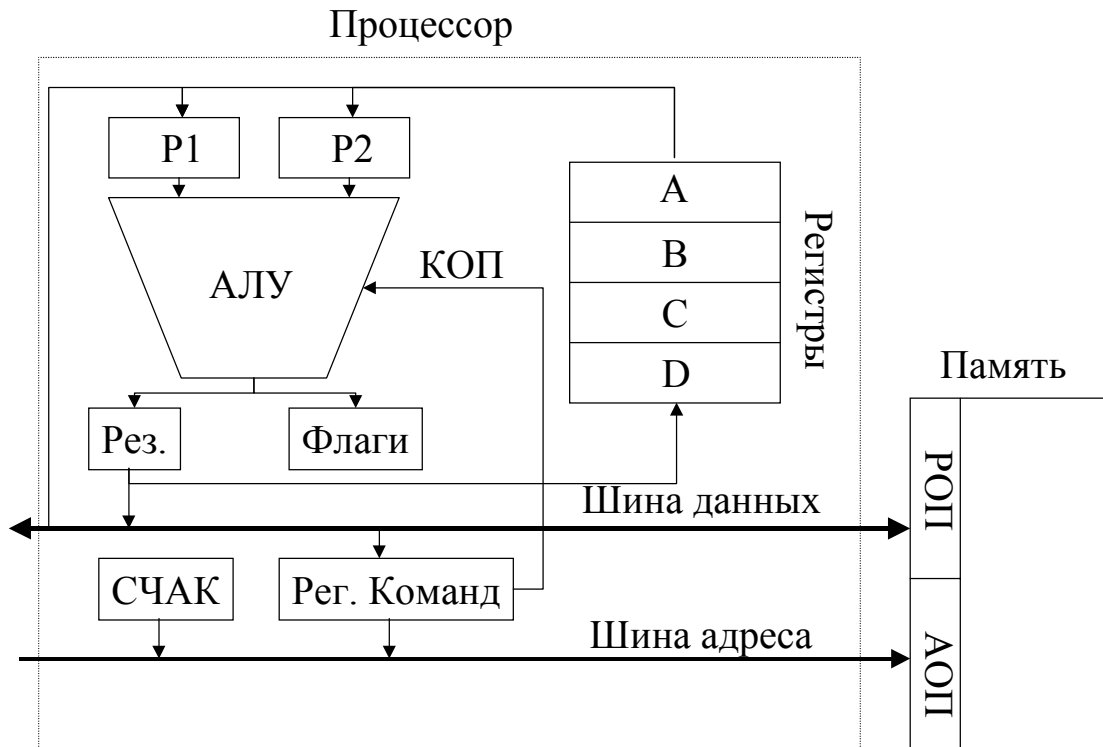
СОДЕРЖАНИЕ

ВАРИАНТЫ ЗАДАНИЙ	4
4.1 ОПИСАНИЕ СТРУКТУРЫ ПРОЦЕССОРА, СИСТЕМЫ КОМАНД И СПОСОБОВ КОДИРОВАНИЯ	4
<i>Структура 1.</i>	4
<i>Структура 2.</i>	5
<i>Структура 3.</i>	5
<i>Структура 4.</i>	7
<i>Структура 5.</i>	10
<i>Структура 6.</i>	10
<i>Структура 7.</i>	12
<i>Структура 1. Описание и система команд</i>	4
<i>Структура 2. Описание и система команд.</i>	5
<i>Структура 3. Описание и система команд.</i>	7
<i>Структура 4. Описание и система команд.</i>	8
<i>Структура 5. Описание и система команд</i>	10
<i>Структура 6. Описание и система команд.</i>	12
<i>Структура 7. Описание и система команд.</i>	13
4.2 ВАРИАНТЫ ЗАДАНИЙ	14
СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ	15

Варианты заданий

4.1 Описание структуры процессора, системы команд и способов кодирования

Структура 1.



Структура 1. Описание и система команд

Объем оперативной памяти – 256 байт.

Длина команд – один или два байта. В первом байте биты 0..3 являются кодом операции, а следующие интерпретируются в зависимости от принадлежности команд к той или иной группе:

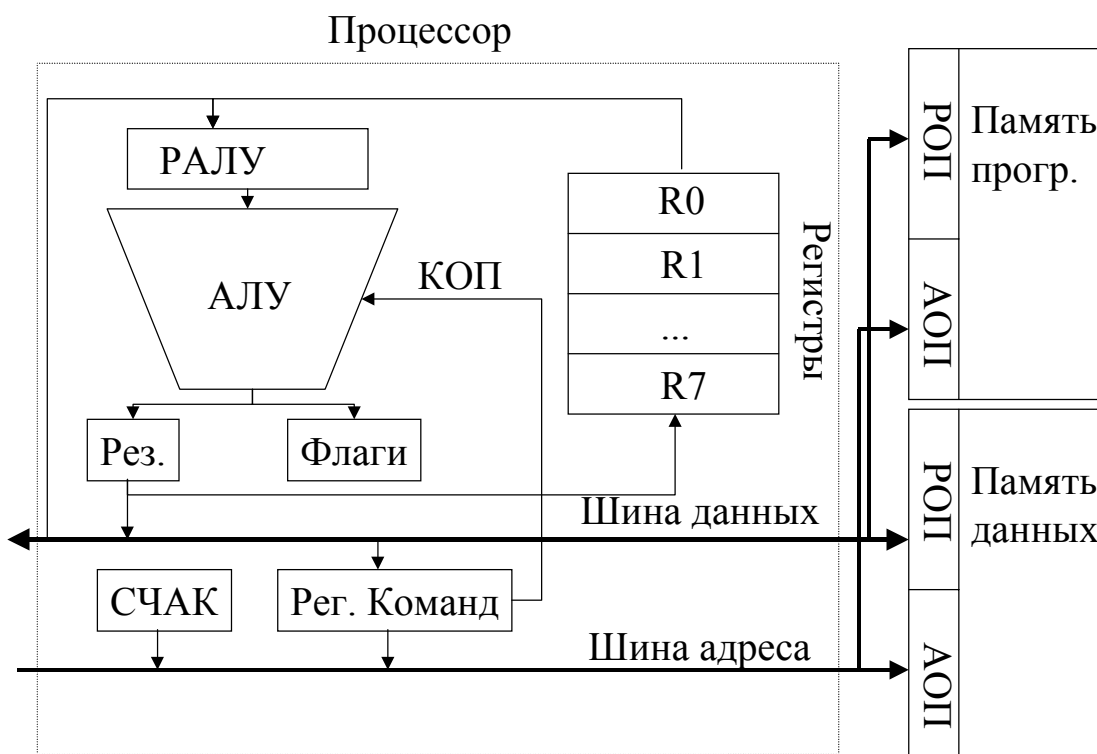
- Для арифметических и логических команд – содержат номера РОИ (0-А, 1-В, 2-С, 3-Д), над которыми производятся операции;
- Для команд пересылки информации – номер регистра, с которым осуществляется операция;
- Не используются для команд передачи управления и управления процессом.

Второй байт используется для задания адреса памяти, где находится операнд, адреса перехода для команд передачи управления или константы.

Код	Мнемокод и операнды	Описание
0000	ADD регистр1,регистр2	регистр1=регистр1+регистр2
0001	ADD регистр, память	регистр=регистр+память
0010	AND регистр1,регистр2	регистр1=регистр1&регистр2
0011	OR регистр1,регистр2	регистр1=регистр1 регистр2
0100	NOT регистр	инверсия битов регистра
0101	MOV регистр,константа	занести константу в регистр

0110	LOAD регистр, адрес	загрузка данных в регистр из памяти
0111	STORE регистр, адрес	запись данных из регистра в память
1000	JMP адрес	безусловный переход
1001	JZ адрес	переход, если 0
1010	JO адрес	переход, если переполнение
1011	JL адрес	переход, если меньше
1100	JNZC регистр, адрес	переход, если не ноль, декремент регистра
1101	CLF	установка регистра флагов в 0
1110	CALL адрес	переход к подпрограмме, адрес возврата в регистре D
1111	RET	возврат из подпрограммы

Структура 2.



Структура 2. Описание и система команд.

Объем оперативной памяти данных –256 байтов, объем оперативной памяти программ –256 байтов. Данные и программы хранятся отдельно.

Команды имеют длину один или два байта. Первый байт определяет код команды в соответствии с таблицей команд (бит 0-4), а также номер регистра-операнда (биты 5-7), указывающего на один из РОНов процессора.

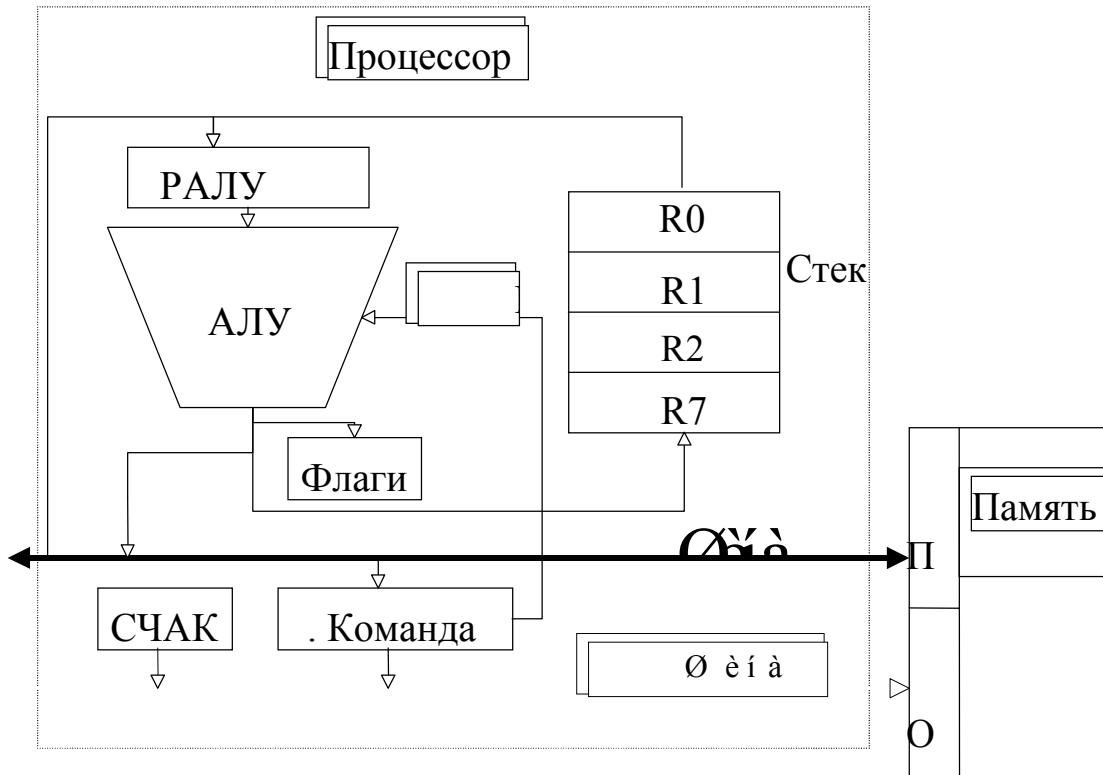
Второй байт определяет адрес операнда в памяти, или задает константу.

Арифметические и логические операции выполняются над операндом, находящемся в памяти или регистре, и результатом предыдущей операции, который нахо-

дится в регистре результатов. Флаг переполнения учитывается как единица младшего разряда, если используется в арифметических командах.

Код	Мнемокод и операнды	Описание
00000	ADD PОН	рез=PОН+рез
00001	ADD адрес	рез=память[адрес]+рез
00010	ADC PОН	рез=PОН+рез+флаг переполнения
00011	SUB PОН	рез= рез-PОН
00100	SUB адрес	рез=рез-память[адрес]
00101	SBB PОН	рез=рез-PОН-флаг переполнения
00110	AND PОН	рез=PОН&рез
00111	AND адрес	рез=память[адрес]&рез
01000	OR PОН	рез=PОН рез.
01001	OR адрес	рез=память[адрес] рез
01010	XOR PОН	рез=PОН⊗рез.
01011	XOR адрес	рез=память[адрес]⊗рез
01100	MUL PОН	рез=рез*PОН
01101	DIV PОН	рез=рез/PОН (деление нацело, остаток – в PОН+1)
01110	LDR PОН	рез=PОН
01111	SVR PОН	PОН=рез
10000	CALL PОН,адрес	переход к подпрограмме; адрес возврата сохраняется в PОНе;
10001	RET PОН	возврат из подпрограммы; адрес возврата находится в PОНе
10010	INC PОН	увеличение PОНа на 1
10011	DEC PОН	уменьшение PОНа на 1
10100	MOV PОН,константа	занести в PОН константу
10101	MOV PОН,адрес	PОН= память[адрес]
10110	MOV память,PОН	память[адрес]=PОН
10111	LOAD PОН	рез=память[адрес=PОН]
11000	SAVE PОН	память[адрес=PОН]=рез
11001	JMP адрес	безусловный переход
11010	JZ адрес	переход, если 0
11011	JNZ адрес	переход, если не 0
11110	HALT	останов
11111	CLR	установка регистра результата в 0

Структура 3.



Структура 3. Описание и система команд.

Объем оперативной памяти – 4096 байт.

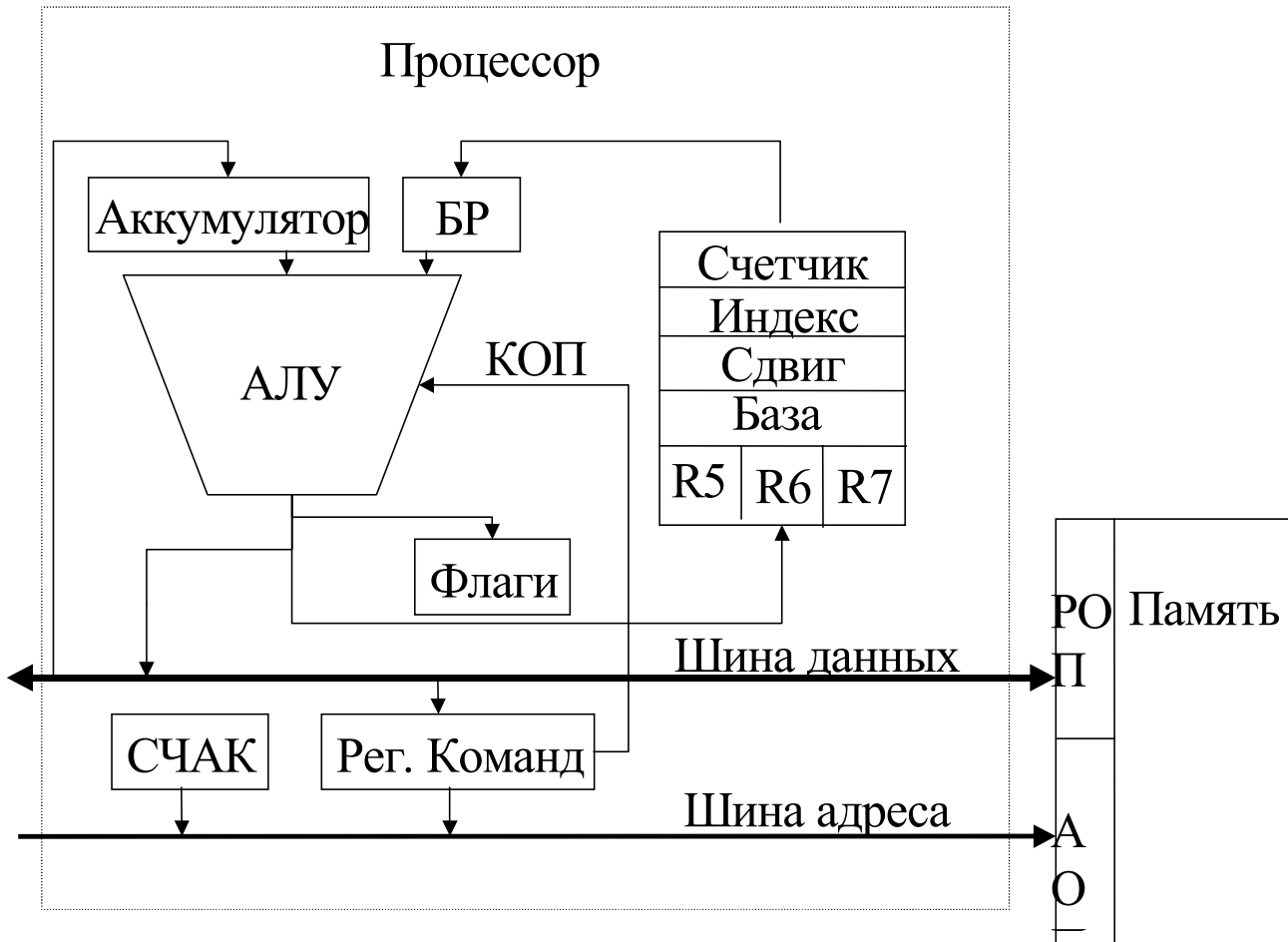
Команды имеют длину один или два байта. Биты 0-3 первого байта содержат код команды в соответствии с таблицей команд. Биты 4-7 и второй байт используются для задания адреса памяти, где находится операнд или константа.

Арифметические и логические операции выполняются над операндом, находящемся в памяти, и верхним элементом стека. Результат выполнения операций в память.

Код	Мnemonic	Описание
0000	ADD адрес	память[адрес]=стек[0]+ память[адрес]
0001	ADC адрес	память[адрес]=стек[0]+ память[адрес]+ флаг переполнения
0010	SUB адрес	память[адрес]=стек[0]-память[адрес]
0011	SUB адрес	память[адрес]=стек[0]- память[адрес]- флаг переполнения
0100	AND адрес	память[адрес]=стек[0]& память[адрес]
0101	OR адрес	память[адрес]=стек[0] память[адрес]
0110	NOT	инверсия бит стек[0]
0111	PUSH адрес	сдвиг стека; стек[0]=память[адрес]
1000	POP адрес	сдвиг стека; память[адрес]= стек[0]
1001	INC регистр	стек[0]=стек[0]+1
1010	DEC регистр	стек[0]=стек[0]-1
1011	JMP адрес	безусловный переход

Код	Мнемокод	Описание
1101	JNZ адрес	переход, если не 0
1110	CALL адрес	переход к подпрограмме; адрес возврата сохраняется в регистрах стек[0], стек[1];
1111	RET	возврат из подпрограммы; адрес возврата находится в регистрах стек[0], стек[1];

Структура 4.



Структура 4. Описание и система команд.

Объем оперативной памяти – 256 байт. Микропроцессор имеет 8 регистров, 5 из них – специального назначения:

РОН0 (AC)– аккумулятор (в него заносятся результаты выполнения всех операций);

РОН 1 (CNT) – счетчик (для операций увеличения (уменьшения) на 1);

РОН 2 (SH) – сдвиговой регистр (для выполнения команд сдвига);

РОН 3 (BASE) – регистр базы (содержит базовый адрес операнда в ОП);

РОН 4 (IND) – регистр индекса (для формирования смещения операнда в ОП);

Коды команд:

Команды имеют длину один или два байта. Биты 0-3 первого байта определяют код команды в соответствии с таблицей команд. Номер регистра хранится в битах 4-6 первого байта. Бит 7 указывает направление перемещения данных: 0 означает перемещение данных в оперативную память, в остальных случаях он равен 1.

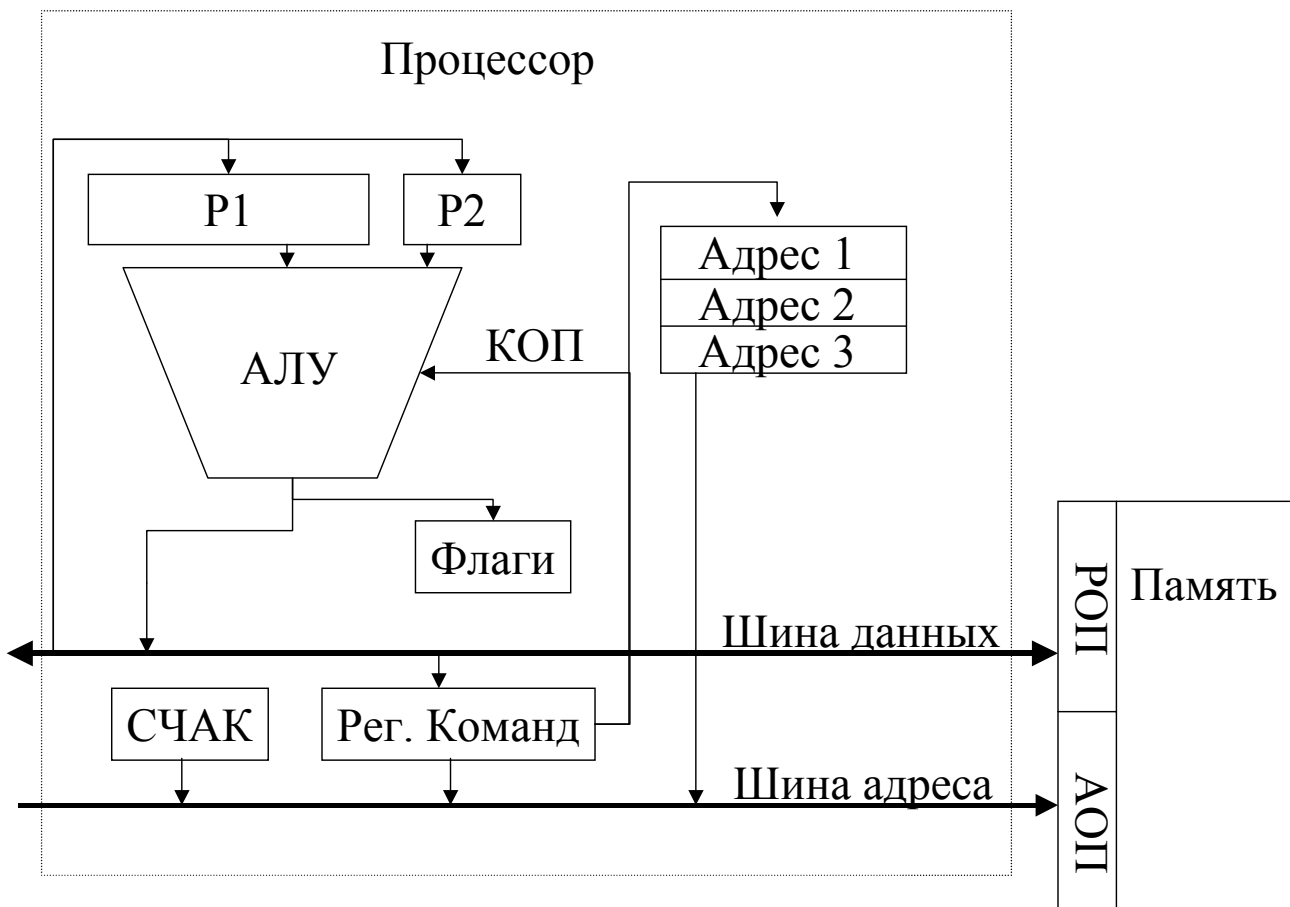
Второй байт определяет константу или адреса памяти.

Арифметические и логические операции выполняются над операндами, находящимися в аккумуляторе и (или) регистре.

Код	Мнемокод	Описание
0000	ADD регистр	аккумулятор= аккумулятор + регистр
0001	ADC регистр	аккумулятор= аккумулятор + регистр + флаг переполнения
0010	SUB регистр	аккумулятор= аккумулятор - регистр
0011	SBB регистр	аккумулятор= аккумулятор - регистр - флаг переполнения
0100	AND регистр	аккумулятор=аккумулятор & регистр
0101	OR регистр	аккумулятор=аккумулятор регистр
0110	NOT	аккумулятор=не аккумулятор
0111	MOV регистр, константа	регистр=константа
1000	STOS	память[BASE+IND]= аккумулятор IND=IND+1
1000	LODS*	Аккумулятор =память[BASE+IND], IND=IND+1
1001	OUTR регистр, адрес	память[адрес]= регистр
1001	INR* регистр, адрес	регистр= память[адрес]
1010	LOOP адрес	переход по адресу, если CNT>0, CNT=CNT-1
1011	JNZ адрес	переход, если не 0
1100	JB адрес	переход, если результат меньше 0
1101	SWAP регистр	Обмен данными между аккумулятором и регистром
1110	SHR	Сдвиг вправо регистра SH на число разрядов в регистре CNT
1111	SHL	Сдвиг влево регистра SH на число разрядов в регистре CNT

* направление перемещения данных определяется последним битом первого байта

Структура 5.



Структура 5. Описание и система команд

Объем оперативной памяти – 2048 байт (8 страниц по 256 байт). Регистр СЧАК имеет 12 битов, но старший бит не используется. Микропроцессор имеет три адресных регистра. Регистры АДРЕС1 (А1) и АДРЕС2 (А2) содержат адреса операндов команд, регистр АДРЕС3 (А3) – адрес результата операции. Эти регистры имеют по 8 битов и определяют смещение в диапазоне 0-255 внутри страницы. Номер страницы задан в битах 5-7 регистра команд.

Команды имеют длину один или два байта. Биты 0-4 первого байта определяют код команды в соответствии с таблицей команд. Биты 5-7 первого байта задают номер страницы ОП, в которой располагаются данные. Второй байт определяет константу.

Код	Мнемокод	Описание
00000	MOVA1 константа	A1=константа
00001	MOVA2 константа	A2=константа
00010	MOVA3 константа	A3=константа
00100	ADD страница*	Память[A3]= Память[A1]+ Память[A2]
00101	ADC страница	Память[A3]= Память[A1]+ Память[A2]+флаг переп.

* номер страницы – целое число от 0 до 7

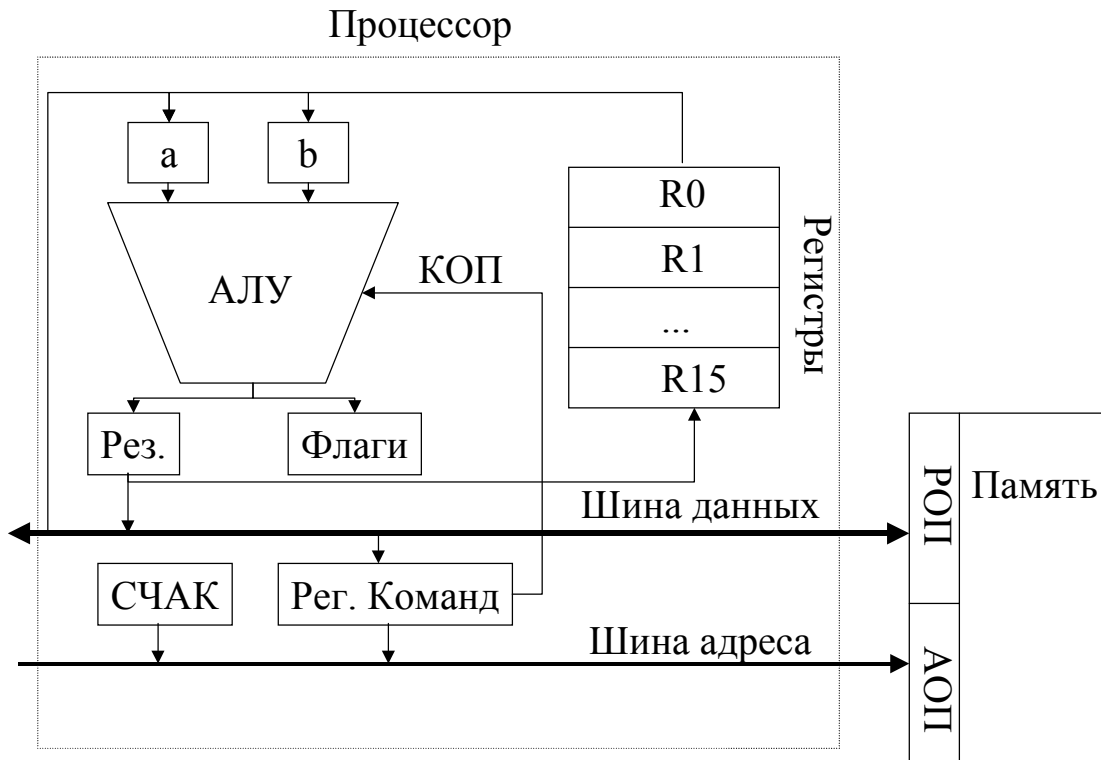
Код	Мнемокод	Описание
00110	SUB страница	Память[A3]= Память[A1]- Память[A2]
00111	SBB страница	Память[A3]= Память[A1]- Память[AC2] -флаг переп.
01000	MUL страница	Память[A3]= Память[A1]* Память[A2]
01001	DIV страница	Память[A3]= Память[A1]/ Память[A2]
01010	MOD страница	Память[A3]= Память[A1]% Память[A2]
01011	ABS страница	Память[A3]= Память[A1] ^{**}
01100	AND страница	Память[A3]=Память[A1]& Память[A2]
01101	OR страница	Память[A3]= Память[A1] Память[A2]
01110	XOR страница	Память[A3]= Память[A1] ^ Память[A2]
01111	NOT страница	Память[A3]=not Память[A1]
10000	JMP страница	Переход по адресу A3
10001	JB страница	Переход по адресу A3, если результат меньше 0
10010	JNZ страница	Переход по адресу A3, если результат не равен 0
10101	MOVS	Память[A3]=Память[A1], A3+=1, A1+=1
10101	CMPS	Сравнение данных Память[A1] и Память[A3] ^{***}

** отрицательные числа представляются в дополнительном коде. Для нахождения модуля необходимо выполнить перевод в прямой код и обнулить старший бит

*** выполняется аналогично операции вычитания, но результат операции не заносится в память

Структура 6

Описание и система команд.



Объем оперативной памяти – 256 байт. Буферные регистры *a* и *b* предназначены для хранения операндов операций. Рабочий регистр *Рез* используется как регистр результата, полученного при обработке данных в АЛУ.

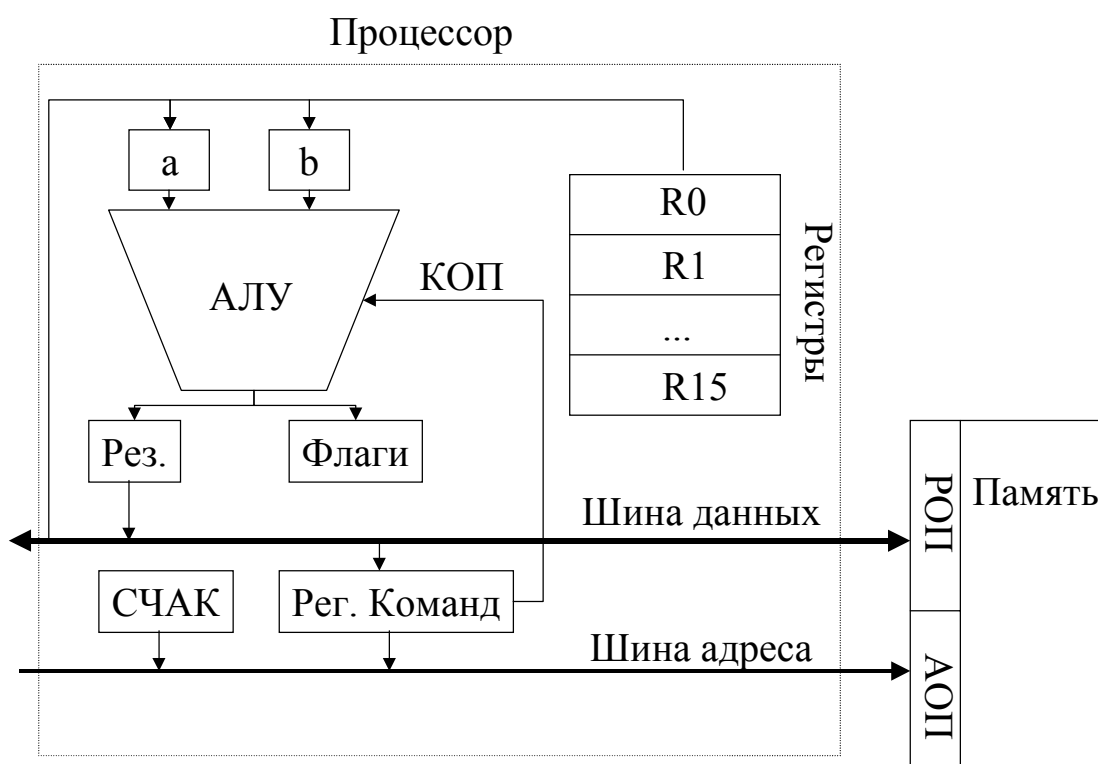
Коды команд:

Команды имеют длину один или два байта. Первый байт определяет код команды в соответствии с таблицей команд (биты 0-3). Следующие 4 бита предназначены для указания номера регистра, коммутируемого к буферному регистру *a*.

Второй байт используется для задания константы или адреса памяти, где находится операнд. Если в качестве второго операнда используется регистр, то его номер хранится в младших 4-х битах второго байта команды, а старшие 4 бита не используются.

Код	Мнемокод	Описание
0000	ADNAB регистр,регистр	$\bar{a} + b$
0001	ADANB регистр,регистр	$a + \bar{b}$
0010	ADAB регистр,регистр	$a + b$
0011	ANBONAB регистр,регистр	$ab \vee \bar{a}\bar{b}$
0100	ABONANB регистр,регистр	$ab \vee \bar{a}\bar{b}$
0101	NAB регистр,регистр	$\bar{a}\bar{b}$
0110	ANB регистр,регистр	$a\bar{b}$
0111	JZ адрес	Переход, если результат равен 0

Код	Мнемокод	Описание
1001	JL адрес	Переход, если результат меньше 0
1010	JMP адрес	Безусловный переход
1100	MOVC регистр, константа	Регистр:=Константа
1101	MOVR регистр, адрес	Регистр:= Память[адрес]
1110	MOVМ регистр, адрес	Память[адрес]:=регистр
1111	СМР регистр, регистр	Сравнение двух регистров. Установка флагов.



Структура 7.

Структура 7. Описание и система команд.

Объем оперативной памяти – 256 байт. Буферные регистры а и b предназначены для хранения операндов операций. Рабочий регистр Рез используется как регистр результата, полученного при обработке данных в АЛУ.

Коды команд:

Команды имеют длину один или два байта. Первый байт определяет код команды в соответствии с таблицей команд (биты 0-3). Следующие 4 бита предназначены для указания номера регистра, коммутируемого к буферному регистру а.

Второй байт используется для задания адреса памяти, где находится второй операнд. Если в качестве второго операнда используется регистр, то его номер хранится в младших 4-х битах второго байта команды, а старшие 4 бита не используются.

Код	Мnemonic	Описание
0000	ILR регистр	$R_n := R_n + \text{флаг переполнения}$
0001	NOTR регистр	$R_n := \overline{R_n} + \text{флаг переполнения}$
0010	ADR регистр, память	$R_n := R_n + \text{Память[Адрес]} + \text{флаг переполнения}$
0011	CLR регистр	$R_n := 0$
0100	ONR регистр, адрес	$R_n := \overline{R_n} \vee \text{Память[Адрес]}$
0101	ANR регистр, адрес	$R_n := \overline{R_n} \wedge \text{Память[Адрес]}$
0110	XNR регистр, адрес	$R_n := \overline{R_n} \oplus \text{Память[Адрес]}$
0111	JZ адрес	Переход, если результат равен 0
1001	JL адрес	Переход, если результат меньше 0
1010	JMP адрес	Безусловный переход
1011	CLF	Регистр флагов := 0
1100	DOP регистр	Перевод содержимого R_n в дополнительный код
1101	LOAD регистр, адрес	$R_n := \text{Память[адрес]}$
1110	SAVE регистр, адрес	$\text{Память[адрес]} := R_n$
1111	CMR регистр, адрес	Сравнение содержимого R_n и Память[адрес] . Установка флагов.

4.2 Варианты заданий

Вариант структуры вычисляется по номеру N зачетной книжки по формуле:

$$\text{Номер структуры} = ((N \bmod 100) \bmod 7) + 1.$$

Используя систему команд процессора, следует составить программу для придуманной задачи невысокой сложности, и отладить ее, используя созданный эмулятор. Текст программы, обоснование структуры программы и результаты выполнения программы должны быть представлены в пояснительной записке. В программе должны присутствовать ветвление и цикл. Условие задачи следует согласовать с преподавателем – руководителем курсового проектирования.

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. Бек Л. Системное программное обеспечение. – М.:Мир,1990.-425 с.