

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
**Федеральное государственное автономное
образовательное учреждение высшего образования**
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Невинномысский технологический институт (филиал)

Методические указания по выполнению лабораторных работ
по дисциплине
«Микропроцессорные системы управления»

15.03.04 «Автоматизация технологических процессов»

ЧАСТЬ I

Невинномысск, 2020

Методические указания предназначены для студентов очной и заочной форм обучения специальностей 15.03.04 «Автоматизация технологических процессов». Они содержат основы теории, описание опытных установок, порядок проведения лабораторных работ и обработки экспериментальных данных, перечень контрольных вопросов для самоподготовки и список рекомендуемой литературы. Объем и последовательность выполнения работ определяются преподавателем в зависимости от количества часов, предусмотренных учебным планом дисциплин как для очной, так и для заочной форм обучения.

Методические указания разработаны в соответствии с требованиями Государственного образовательного стандарта в части содержания и уровня подготовки бакалавров по направлению 15.03.04 «Автоматизация технологических процессов».

Составители: канд. техн. наук Карабак Ю.В.,
Евдокимов А.А.,
Щербина С.Ю.,
Ковалев М.В.,
Топчиев А.В.

Ответственный редактор: канд. техн. наук Болдырев Д.В.

Содержание

Введение	5
1. Основные правила составления программ на ассемблере.....	7
2. Общие сведения об элементах МП систем	20
3. Учебный микропроцессорный комплект.....	34
Лабораторная работа №1	37
Лабораторная работа №2.....	51
Лабораторная работа №3	55
Приложение	60

Условные обозначения, используемые в методических указаниях:

ОЗУ – оперативное запоминающее устройство;

АЦП – аналого-цифровой преобразователь;

ЦАП – цифро-аналоговый преобразователь;

ПК – программатор кодов;

ЗУ – запоминающее устройство;

БУК – блок управления комплектом;

МЗР – младший значимый разряд;

СЗР – старший значимый разряд;

УС – устройство сменное;

ИС – интегральная микросхема.

ПСИ – программатор серии импульсов;

ПК – программатор кодов;

ГПИ – генератор прямоугольных импульсов;

ОУ – операционный усилитель;

ЦОС – цепь отрицательной обратной связи;

ИОН – источник опорного напряжения.

ВВЕДЕНИЕ

Применение микропроцессоров (МП) в электроприводах и устройствах автоматики переносит центр тяжести процесса проектирования на разработку прикладных программ, реализующих алгоритм функционирования аппаратуры с МП.

В настоящее время программирование МП ведется на языке ассемблера или на языке высокого уровня. Ассемблирование может выполняться вручную или с привлечением соответствующей программы. Ручное ассемблирование, т.е. перевод команд (операторов) в машинные коды, рекомендуется лишь на этапе начального обучения проектированию прикладных программ, так как процесс этот трудоемок, а вероятность ошибок велика.

Язык ассемблера относится к машинно-ориентированным языкам низкого уровня. Каждой команде языка ассемблера соответствует, как правило, машинная команда. Каждый тип МП имеет свою систему команд и, следовательно, свой язык ассемблера. Команды, адреса и данные записываются в виде буквенно-цифровых символов. Мнемонические обозначения команд обычно легко ассоциируются с реально выполняемыми действиями (в англоязычном написании). Практика показывает, что адаптация к языку ассемблера происходит быстро. После составления нескольких простых программ мнемонические обозначения наиболее употребительных команд запоминаются. Написанная на ассемблере программа переводится на машинный язык с помощью транслирующей программы, называемой также ассемблер. При этом проводится проверка синтаксиса языка с указанием ошибок, которые необходимо исправить. Программирование на ассемблере позволяет создавать максимально эффективные с точки зрения быстродействия и потребного объема памяти программы.

Существенное повышение производительности труда при проектировании прикладных программ дает использование языков высокого уровня, специально приспособленных для программирования МП. Являясь

машинно-независимыми, они позволяют весь процесс получения программы в машинных кодах свести к трансляции (переводу) с одного представления в другое. С помощью специальной компилирующей программы производится трансляция с языка высокого уровня на язык ассемблера и далее с помощью ассемблера – в машинные коды. Каждому оператору языка высокого уровня соответствует 5-10 и более команд Ассемблера, что, очевидно, облегчает программирование. Поскольку трансляторы создаются для широкого круга программ, то, естественно, осуществляемый ими перевод не позволяет достичь той же эффективности, как при программировании на ассемблере. Окончательная отладка программы ведется обычно на ассемблере. Из языков высокого уровня последнее время широко используется язык СИ.

Лабораторный практикум организован на базе «Учебных микропроцессорных комплектов» (УМК) и «Блока управления комплектом К32» (БУК К32).

1. ОСНОВНЫЕ ПРАВИЛА СОСТАВЛЕНИЯ ПРОГРАММ НА АССЕМБЛЕРЕ

1.1. Коды команд

Микропроцессор работает только с двоичными кодами. Среди совокупности этих кодов имеется определенная группа кодов, каждый из которых может «заставить» микропроцессор выполнить определенные действия (операции). Такой код часто называется кодом команды и, как правило, определяет одну команду из системы команд микропроцессора. Код любой команды представляет в запоминающем устройстве двоичным восьмиразрядным числом (байтом). Всего с помощью байта можно формировать $2^8 = 256$ различных кодовых комбинаций. Почти столько же команд (244) имеет микропроцессор КР580ВМ80. Запомнить 244 кода довольно трудно, и поэтому каждому коду ставится в соответствие мнемоническое название (мнемоника) команды, которое является сокращением от английских слов, описывающих ее действие. Мнемонический код команд позволяет легче запомнить их функции и значительно упрощает написание программ. Такой язык написания программ называется ассемблером. После того как программа написана на ассемблере, ее необходимо снова перевести на язык, понятный микропроцессору, т.е. перевести в последовательность двоичных восьмиразрядных чисел. Перевод в последовательность двоичных цифр может происходить автоматически с помощью специальных программ – трансляторов (такие программы носят название «Кросс – ассемблер» или «ассемблер») или вручную.

Для ручной трансляции можно использовать таблицу П 1 (приведенную в Приложении), в которой приведены все команды микропроцессора КР580ВМ80. С помощью этой таблицы можно легко и быстро сопоставить мнемонику команды с ее кодом. Код каждой команды приведен в верхней горизонтальной строке (младшие разряды)

и в крайнем левом столбце (старшие разряды) в шестнадцатеричной системе счисления. Например, команда STAX D имеет код 12, команда JZ – код CA.

1.2. Программная модель микропроцессорной системы

Система команд микропроцессора – это полный перечень элементарных действий, которые может выполнить микропроцессор. Управляемый командами микропроцессор выполняет очень простые действия, однако с помощью этих действий (команд) можно запрограммировать любую сложную операцию.

Проектировщику необходимо запомнить весь перечень команд и хорошо представлять действия, которые будет выполнять микропроцессор при их обработке.

Программная модель системы, построенной на базе микропроцессора КР580ВМ80, состоит из следующих элементов:

- программно-доступных регистров и триггеров разрешения прерывания;
- программно-доступных восьмиразрядных ячеек памяти;
- программно-доступных восьмиразрядных регистров.

Программно-доступные регистры микропроцессора – это регистры общего назначения, регистр указателя стека, регистр признаков и регистр счетчика команд. Разряды регистров нумеруются справа налево целыми числами, начиная с нуля.

Регистр общего назначения (РОН) – это один из шести восьмиразрядных регистров микропроцессора, обозначенных буквами В, С, D, E, H, L, или аккумулятор, обозначенный буквой А. Регистры В и С, D и E, H и L в некоторых командах рассматриваются как шестнадцатиразрядные регистры, называемые регистровыми парами. Регистры В, D и H образуют старшие восемь разрядов регистровых пар, а регистры С, E, L – младшие.

Регистр указателя стека (УС) – это шестнадцатиразрядный регистр, который содержит адрес вершины стека. Стек – это динамическая последовательная структура данных в ОЗУ, организованная таким образом, что очередная запись данных всегда осуществляется в вершину (начало) стека. Максимальный разряд стека равен адресуемой емкости памяти.

При записи данных в стек содержимое регистра указателя стека уменьшается на 2, а при считывании – увеличивается на 2.

Регистр признаков (F) – это восьмиразрядный регистр, содержащий признаки результата выполнения команды (рисунок 1).

Таблица 1 – Программная модель микропроцессорной системы

	0	1	2	3	4	5	6	7	8
0	NOP				MOV B,B	MOV D,B	MOV H,B	MOV M,B	ADD B
1	LXI B	LXI D	LXI H	LXI SP	MOV B,C	MOV D,C	MOV H,C	MOV M,C	ADD C
2	STAX B	STAX D	HLD	STA	MOV B,D	MOV D,D	MOV H,D	MOV M,D	ADD D
3	INX B	INX D	INX H	INX SP	MOV B,E	MOV D,E	MOV H,E	MOV M,E	ADDE
4	INR B	INR D	INR H	INR M	MOV B,H	MOV D,H	MOV H,H	MOV M,H	ADD H
5	DCR B	DCR D	DCR H	DCR M	MOV B,L	MOV D,L	MOV H,L	MOV M,L	ADD L
6	MVIB	MVID	MVI H	MVI M	MOV B,M	MOV D,M	MOV H,M	HTL	ADD M
7	RLC	RAL	DAA	STC	MOV B,A	MOV D,A	MOV H,A	MOV M,A	ADD A
8					MOV C,B	MOV E,B	MOV L,B	MOV A,B	ADC B
9	DADB	DADD	DADH	DAD SP	MOV C,C	MOV E,C	MOV L,C	MOV A,C	ADC C
A	DA B	DA D	HL	LDA	MOV C,D	MOV E,D	MOV L,D	MOV A,D	ADC D
B	DCX B	DCX D	DCX H	DCX SP	MOV C,E	MOV E,E	MOV L,E	MOV A,E	ADC E
C	INR C	INR E	INR L	INR A	MOV C,H	MOV E,H	MOV L,H	MOV A,H	ADC H
D	DCR C	DCR E	DCR L	DCR A	MOV C,L	MOV E,L	MOV L,L	MOV A,L	ADC L
E	MVIC	MVIE	MVIL	MVI A	MOV C,M	MOV E,M	MOV L,M	MOV A,M	ADC M
F	RRC	RAR	CMA	CMC	MOV C,A	MOV E,A	MOV L,A	MOV A,A	ADC A

Продолжение таблицы 1

	9	A	B	C	D	E	F
0	SUB B	ANA B	DRA B	RNZ	RNC	RPO	RP
1	SUB C	ANA C	DRA C	POP B	POP D	POP H	POP RSW
2	SUB D	ANA D	DRA D	JNZ	JNC	JPO	JP
3	SUB E	ANA E	DRA E	JMP	OUT	XTHL	DI
4	SUB H	ANA H	DRA H	CNZ	CNC	CPO	CP
5	SUB L	ANA L	DRA L	PUSH B	PUSH D	PUSH H	PUSH PSW
6	SUB M	ANA M	DRA M	ADI	SUI	ANI	ORI
7	SUB A	ANA A	DRA A	RST 0	RST 16	RST 32	RST 48
8	SBB B	XRA B	CMP B	RZ	RC	RPE	RM
9	SBB C	XRA C	CMP C	RET		PCHL	SPHL
A	SBB D	XRA D	CMP D	JZ	JC	JPE	JM
B	SBB E	XRA E	CMP E		IN	CHG	EI
C	SBB H	XRA H	CMP H	CZ	CC	CPE	CM
D	SBB L	XRA L	CMP L	ALL			
E	SBB M	XRA M	CMP M	ACI	SBI	XRI	CPI
F	SBB A	XRA A	CMP A	RST 8	RST 24	RST 40	RST 56

7	6	5	4	3	2	1	0	S – признак знака
S	Z	0	V	0	P	1	C	Z – признак нуля
Распределение признаков по разрядам.								V – признак дополнительного переноса
								P – признак четности
								C – признак переноса

Рисунок 1 – Регистр признаков

Признаки устанавливаются следующим образом:

- признак S – единица, если седьмой разряд результата равен единице, в противном случае – ноль;
- признак нуля Z – единица, если во всех разрядах результата ноли, в противном случае – ноль;
- признак дополнительного переноса V – единица при переносе из третьего разряда или при заеме в третий разряд результата, в противном случае – ноль;
- признак четности P – единица, если результат в двоичном коде содержит четкое количество единиц, в противном случае – ноль;
- признак переноса C – единица при переносе из седьмого разряда или при заеме в седьмой разряд результата, в противном случае – ноль.

Аккумулятор и регистр признаков образуют слово состояния процессора, обозначенное буквами PSW. Аккумулятору соответствуют восемь старших разрядов, а регистру признаков – младшие.

Регистр счетчика команд (СК) – это шестнадцатиразрядный регистр, указывающий адрес следующей команды, которая должна быть выполнена микропроцессором.

Триггер разрешения прерывания используется для управления прерываниями микропроцессора. Если триггер установлен в единицу, то прерывание разрешается, если триггер установлен в ноль, то прерывание

запрещается.

Программно-доступные восьмиразрядные ячейки памяти используются в качестве памяти микропроцессорной системы (МПС). Разряды ячейки памяти нумеруются справа налево целыми числами, начиная с нуля. Максимальная емкость памяти, реализуемой запоминающим устройством, равна 65536 байт.

Программно – доступные восьмиразрядные регистры используются для ввода и вывода. Максимальное число регистров для ввода данных составляет 256, для вывода данных – столько же.

Команды микропроцессора используют только те элементы данных, обработка которых осуществляется непосредственно микропроцессором, т.е. восьми и шестнадцатибитовые. Восьмибитовые данные – это восемь бит (один байт), которые хранятся в РОН или ячейке памяти и обрабатываются микропроцессором как одно целое.

Шестнадцатибитовые данные – это 16 бит (два байта) данных, которые могут храниться в шестиразрядном регистре или в двух последовательных ячейках памяти и обрабатываются микропроцессором как одно целое, причем в памяти старшие восемь бит записываются по адресу на единицу большему, чем адрес младших. При адресации шестнадцатибитовых данных указывается адрес младшей ячейки памяти, содержащей эти данные.

Операнды команд могут храниться в программно-доступных регистрах микропроцессора или в памяти МПС. Для указания операнда в регистре используются: регистровая адресация и регистровая неявная адресация.

При регистровой адресации в коде команды существует поле для указания регистра, содержащего операнд. При регистровой неявной адресации специального поля в коде команды нет.

Для указания операнда в памяти используются адресации: непосредственная, прямая, косвенная и стековая.

При непосредственной адресации операнд задается в команде. При прямой адресации в команде указывается адрес операнда. При косвенной регистровой адресации адрес операнда указывается в регистровой паре или для команды ХНТЛ – в регистре указателя стека.

1.3. Команды микропроцессора КР580ВМ80

Команды микропроцессора делятся на группы: команды пересылки и загрузки данных, команды обмена данными, арифметические команды, команды поразрядной обработки данных, команды передачи управления, команды управления микропроцессором. Команды могут быть длиной один, два и три байта. При описании команд (далее по тексту) байты будем разделять штрихами.

Команды пересылки и загрузки данных:

- MOV R1, R2 – пересылка из регистра в регистр;
- LDAXB(D) – пересылка из ячейки памяти, адрес которой записан в регистровой паре BC (BE);
- STAX B(D) – пересылка из аккумулятора в ячейку памяти, адрес которой записан в регистровой паре BC (DE);
- LDA B2 B3 – пересылка из ячейки памяти, адрес которой записан во втором и третьем байтах команды, в аккумулятор;
- STA B2 B3 – пересылка из аккумулятора в ячейку памяти, адрес которой указан во втором и третьем байтах команды;
- SPHL – пересылка данных из регистровой пары HL в указатель стека;
- PCHL – пересылка данных из регистровой пары HL в счетчик команд;
- LHLD B2 B3 – пересылка данных из ячеек памяти с адресами, записанными во втором и третьем байтах команды и на единицу больше, в регистровую пару HL;
- SHLD B2 B3 – пересылка данных из регистровой пары HL в ячейки памяти с адресами, записанными во втором и третьем байтах команды и на единицу больше;

- MVIR B2 – загрузка второго байка команды в регистр R;
- LXIB (D, H) B2 B3 – загрузка второго и третьего байтов команды в регистровую пару BC (DE, HL).
- Команды обмена данными:
- XCHG – обмен данными между парами регистров HL и DE;
- XTHL – обмен данными между парой регистров HL и вершиной стека (L) \leftrightarrow [UC], (H) \leftrightarrow [UC+ 1].

Группа арифметических команд

Команды сложения:

- ADD R – сложение содержимого аккумулятора с содержимым регистра R $(A) \leftarrow (A) + (R)$;
- ADI B2 – сложение содержимого аккумулятора со вторым байтом команды $(A) \leftarrow (A) + B2$;
- ADC R – сложение содержимого аккумулятора с содержимым регистра R и признаком (C); $(A) \leftarrow (A) + (R) + (C)$;
- ACI _ B2 – сложение содержимого аккумулятора со вторым байтом команды и признаком (C) $(A) \leftarrow (A) + B2 + (C)$;
- DAD B (D, H, SP) – сложение содержимого пары регистров HL с содержимым пары регистров BC CDE или HL, или указатели стека и запись результата в пару HL.

Команды вычитателя:

- SUB R – вычитание из содержимого аккумулятора содержимого регистра R: $(A) \leftarrow (A) - (R)$;
- SUI B2 – вычитание из содержимого аккумулятора второго байта команды: $(A) \leftarrow (A) - B2$;
- SBB R – вычитание из содержимого аккумулятора содержимого регистра R и признака (C): $(A) \leftarrow (A) - (R) - (C)$;
- SBI B2 – вычитание из содержимого аккумулятора второго байта команды и признака (C): $(A) \leftarrow (A) - B2 - (C)$.

Операции инкремента и декремента:

- INR R – увеличение содержимого регистра на 1: $(R) + 1 \rightarrow (R)$;
- INX B (D, H, SP) – увеличение содержимого пары регистров BC (DE), (HL), (SP) на 1;
- DCR R – уменьшение содержимого регистра R на 1: $(R) - 1 \rightarrow (R)$;
- DCX (D, H, SP) – уменьшение содержимого пары регистров BC (DC), (HL), (SP) на 1.

Группа команд поразрядной обработки данных

Логические команды:

- ANA R – операция «И» между содержимым аккумулятора и содержимым регистра $(A) \leftarrow (A) \wedge (R)$;
- ANI B2 – операция «И» между содержимым аккумулятора и вторым байтом команды $(A) \leftarrow (A) \wedge (B)$;
- ORA R – операция «ИЛИ» между содержимым аккумулятора и содержимым регистра R: $(A) \leftarrow (A) \vee (R)$;
- ORI B2 – операция «ИЛИ» между содержимым аккумулятора и вторым байтом команды: $(A) \leftarrow (A) \vee B2$;
- XRA R – операция «ИЛИ – НЕ» между содержимым аккумулятора и содержимым регистра R: $(A) \leftarrow (A) \vee (R)$;
- XRI B 2 – операция «ИЛИ – НЕ» между содержимым аккумулятора и вторым байтом команды $(A) \leftarrow (A) \vee B2$.

Команды сравнения:

- CMP R – операция $(A) - (R)$; если $(A) = (R)$, то $Z = 1$; если $(A) < (R)$, то $C=1$;
- CPI B2 – операция $(A) - B2$, если $(A) = B2$, то $Z= 1$; если $(A) < B2$, то $C= 1$.

Команды сдвига:

- RLC – сдвиг влево (каждый бит сдвигается на один разряд влево, а 7 бит переносится в 0 и одновременно записывается в признак (C)): Dm

- $D_{m+1}, D_7 \rightarrow D_0, D_7 \rightarrow C$;
- RRC – сдвиг вправо (каждый бит сдвигается на один разряд вправо, а 0 бит переносится в 7 и одновременно записывается в признак (C)): $D_{m+1} \rightarrow D_m, D_0 \rightarrow D_7, D_0 \rightarrow C$;
- RAL – сдвиг влево через перенос (каждый бит сдвигается на один разряд влево, 7 бит записывается в признак (C), а бит из (C) записывается в 0 бит): $D_m \rightarrow D_{m+1}, D_7 \rightarrow C, C \rightarrow D_0$;
- RAR – сдвиг вправо через перенос (каждый бит сдвигается на один разряд вправо, 0 бит записывается в признак (C), а бит (C) записывается в 7 бит): $D_{m+1} \rightarrow D_m, D_0 \rightarrow C, C \rightarrow D_7$.

Команда инверсии содержимого аккумулятора – CMA $(A) \leftarrow (\bar{A})$.

Команды безусловного перехода:

- JMP B2 B3 – записать информацию из второго и третьего байта команды в счетчик команд;
- PCHL – записать содержимое пары регистров HL в счетчик команд.

Команды условного перехода:

- JNC – если признак $Z = 0$, то перейти по адресу,
- JC – если признак $Z = 1$ записанному во втором и
- JNZ – если признак $C = 0$ третьем байтах команды,
- JZ – если признак $C = 1$ иначе перейти к следующей команде.
- JPO – если признак $P = 0$
- JPE – если признак $P = 1$
- JP – если признак $S = 0$
- JM = если признак $S = 1$.

Команды безусловного вызова подпрограммы:

- CALL B2 B3 – вызов подпрограммы с начальным адресом, записанным во втором и третьем байтах команды.
- Команды условного вызова подпрограммы:
- CNZ – если $Z = 0$, то перейти к подпрограмме
- CZ – если $Z = 1$ с начальным адресом,

- CNC – если $C=0$ указанным во втором и
- CC – если $C=1$ третьем байтах команды,
- CPO – если $P=0$ иначе перейти к следующей команде.
- CPE – если $P=1$
- CP – если $S=0$
- CM – если $S=1$

Команда безусловного возврата из подпрограммы – RET

Команды условного возврата из подпрограммы:

- RNZ – если $Z=0$, то возвратиться к команде,
- RZ – если $Z=1$ следующей за командой
- RNC – если $C=0$ вызова подпрограммы,
- RC – если $C=1$ иначе продолжить
- RPO – если $P=0$ выполнение подпрограммы.
- RPT – если $P=1$
- RP – если $S=0$
- RM – если $S=1$

Команды операций со стеком:

- PUSH B (DE, HL) в стек (B,D,H) \rightarrow [УС-1], (C, E, L) \rightarrow [УС-2],
(УС)=(УС-2);
- PUSH HSW – запись слова состояния в стек (A) \rightarrow [УС-1], (F) \rightarrow [УС-2],
(УС)=(УС-2);
- POP B (DH) – запись из стека в регистровую пару BC (DE, HL)
- [УС] \rightarrow (C, E, L), [УС+1] \rightarrow (B, D, H), (УС) = (УС) + 2;
- POP PSW – запись слова состояния из стека [УС] \rightarrow (F), [УС+1] \rightarrow (A),
(УС) = (УС)+ 2.

Команды управления микропроцессором:

- CMC инверсия признака (C): (C) \rightarrow (C);
- STC – запись 1 в признак (C): 1 \rightarrow (C);
- EI – разрешение прерываний;
- NOP – команды «нет операций»;

– HLT – останов.

Ввод – вывод через порты МПС осуществляется специальными командами микропроцессоров: IN ADDR и OUT ADDR, где ADDR – адрес внешнего устройства. Так как адрес внешнего устройства указывается в одном байте, то микропроцессор может обмениваться информацией не более чем с 256 внешними устройствами.

Число различных команд микропроцессора составляет 78, и для них зарезервировано 78 имен команд, перечисленных выше. Многие базовые команды из числа 78 порождают несколько различных кодов операций, поэтому общее число кодов команд составляет 244 (Приложение).

Время выполнения команды, оцениваемое числом тактов микропроцессора, составляет 4 - 18 тактов. Для подавляющего числа команд время выполнения команды (число тактов) есть величина постоянная, однако существуют команды, для которых это время может меняться. Такими командами являются команды условного вызова и возврата из подпрограмм. Время выполнения команды зависит от того, выполняется условие или нет. Если условие выполняется, время выполнения команды увеличивается.

Программирование на ассемблере можно осуществить используя таблицу, приведенную в Приложении. В этой таблице приведены команды по группам. Приведены мнемокод, двоичный код команды или двоичный код, содержащий символы источника или приемника данных, число байтов в команде, число тактов, выполняемые действия, характер изменения битов регистра признаков.

2. ОБЩИЕ СВЕДЕНИЯ ОБ ЭЛЕМЕНТАХ МПС СИСТЕМ

Микропроцессорная система управления является важным дополнением технологического оборудования, так как обеспечивает согласованность функционирования устройств, входящих в подсистемы, гибкость переналаживания и надежность оборудования, качество выпускаемой продукции.

Технические средства микропроцессорной системы управления должны реализовывать функции управления, измерять параметры подсистем (технологических процессов, изделий, создаваемых структур и т.п.), преобразовывать и обрабатывать полученную информацию и формировать управляющие воздействия на объект управления.

Основным элементом архитектуры МПС является регистр, т.е. некоторое физическое устройство, способное воспринимать, хранить и выдавать слово информации. Разрядность всех регистров системы равна или кратна разрядности МП. Ячейки памяти и устройства ввода/вывода также являются регистрами и поэтому любую МПС можно рассматривать как набор огромного числа регистров, а ее функционирование заключается в передаче слов информации между этими регистрами. Порядок передачи информации и ее преобразование определяется последовательностью команд МП. Структурная схема микропроцессорной системы МПС показана на рисунке 2. Центральный процессор выполняет функции управления системой и является активным компонентом. Для синхронизации работы всех устройств системы в нее введен генератор тактовых импульсов ГТИ.

Память МПС состоит из энергонезависимого постоянного запоминающего устройства ПЗУ и оперативного запоминающего устройства ОЗУ. ПЗУ предназначено для долговременного хранения данных, например, управляющей программы. ОЗУ предназначено для временного хранения данных, например, программы пользователя, оперативных данных и т.д. При снятии напряжения питания с микросхемы ОЗУ вся хранящаяся в нем информация теряется.

Для организации ввода и вывода информации из МПС в ней предусмотрены специальные устройства ввода/вывода, называемые интерфейсами ввода/вывода. Одни интерфейсы предназначены для обмена дискретной информацией в параллельном формате, другие – в последовательном формате. Также в системе может присутствовать устройство ввода/вывода аналоговой информации.

На устройства ввода/вывода возлагаются функции сопряжения внешних сигналов с системной шиной МПС; промежуточного хранения сигналов в процессе ввода/вывода; выполнения некоторых специфичных функций с целью перезагрузки МП.

Интерфейсы клавиатуры и дисплея позволяют организовать диалог между оператором и микро-ЭВМ.

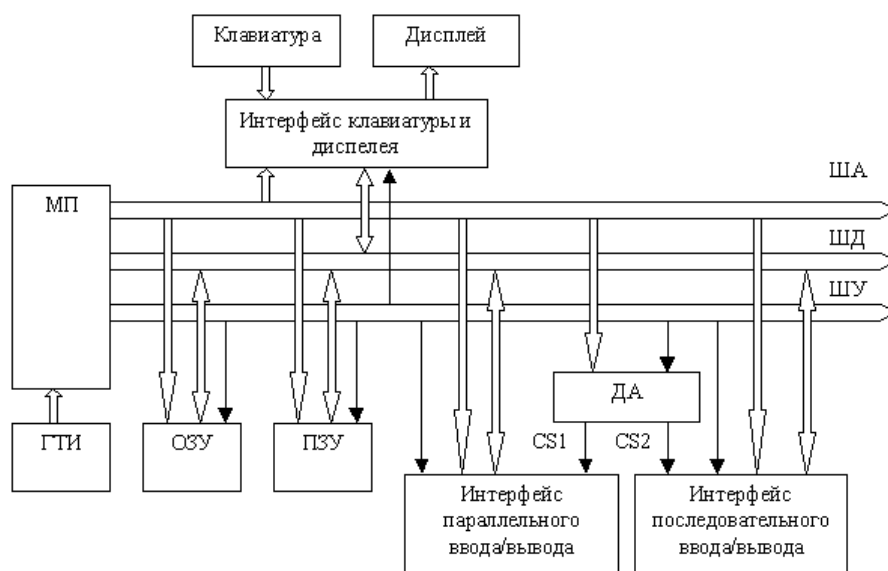


Рисунок 2 – Структурная схема МПС

2.1. Центральное процессорное устройство на примере КР580ВМ80

Микропроцессор КР580ВМ80А (МП) представляет собой монолитный параллельный 8-битный процессор, имеющий фиксированную систему команд и классическую архитектуру с отдельными шинами адреса и данных. Он рассчитан на самое разнообразное применение в качестве ядра микропроцессорной системы (МП-системы), выполняющего функции преобразования данных. КР580 выполнен по NMOP-технологии и выпус-

кается в 40-контактном керамическом корпусе с двусторонним расположением выводов (рисунок 3). Микропроцессор является активным компонентом системы и реализует следующие функции:

- управляет выполнением команд программы;
- управляет обменом информацией между компонентами МП-системы;
- реагирует на разнообразные внешние сигналы.

10	D0	CPU	A0	25
9	D1		A1	26
8	D2		A2	27
7	D3		A3	29
3	D4		A4	30
4	D5		A5	31
5	D6		A6	32
6	D7		A7	33
24	WAIT		A8	34
23	READY		A9	35
22	F1		A10	1
15	F2		A11	40
19	SYNC		A12	37
12	RESET		A13	38
17	DBIN		A14	39
18	WR		A15	36
20	+5 B		HLDA	21
11	- 5 B		HOLD	13
28	+12B		INT	14
2	GND		INTE	16

Рисунок 3 – Микропроцессор КР 580ВМ80

Микропроцессор КР580 имеет 16 адресных линий (А0÷А15), что позволяет ему адресовать до 2^{15} (65536) ячеек памяти и 256 устройств ввода и 256 устройств вывода. Для обмена информацией имеется 8-разрядная шина данных (линии D0÷D7).

Управляющие сигналы микропроцессора можно разделить на следующие группы:

Группа сигналов управления состоянием микропроцессора:

1. Входной сигнал сброса (инициализации) RESET, заставляющий процессор начать выполнение программы с нулевой ячейки памяти.
2. Входной сигнал готовности READY, поступающий от других компонентов МП-системы, который служит для остановки действий микропроцессора до тех пор, пока компонент не будет готов к обмену данными с МП.
3. Выходной сигнал ожидания WAIT, свидетельствующий о том, что МП приостановлен.

Группа сигналов управления шинами адреса и данных:

1. Выходной сигнал считывания (приема) DBIN, определяющий направление передачи по шине данных в МП.
2. Выходной сигнал записи (выдачи) WR, определяющий направление передачи по шике данных от МП».
3. Входной сигнал запроса шины (захвата) HOLD, информирующий МП о том, что быстродействующее периферийное устройство должно использовать шины адреса и данных для прямого обмена данными с памятью системы без участия МП.
4. Выходной сигнал подтверждения запроса шин (захвата) HLDA, означающий, что буфера шин адреса и данных МП переведены в высокоимпедансное состояние и шинами может распоряжаться периферийное устройство.

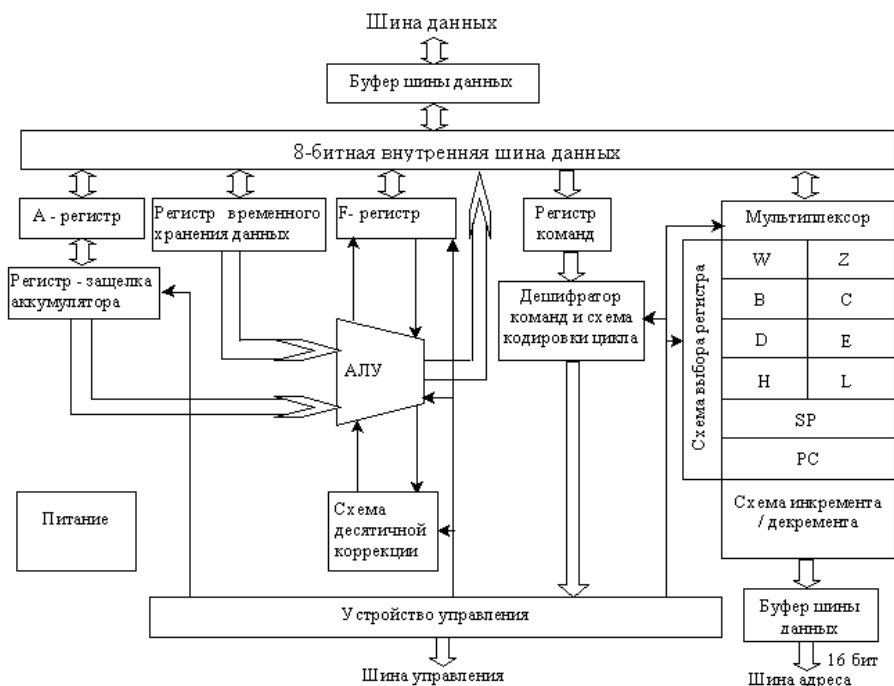
Группа сигналов, связанных с прерываниями, т.е. с запросами периферийных устройств на обмен информацией:

1. Входной сигнал запроса прерывания INT, означающий готовность периферийного устройства к обмену данными.
2. Выходной сигнал разрешения прерывания INTE, определяющий возможность или невозможность обслуживания МП запросов периферийных устройств.

Микропроцессор имеет три линии синхронизации. На входы F1 и F2 подаются двухфазные сигналы, не перекрывающиеся от главного генератора синхронизации. Частота сигналов составляет 2 МГц.

Выходной сигнал SYNC идентифицирует начало каждого машинного цикла. Одновременно с этим сигналом МП выдает на шину данных байт состояния, содержащий информацию о действиях МП в текущем машинном цикле.

Архитектура микропроцессора КР580 ВМ80А



На рисунке 4 приведена структурная схема МП КР580ВМ80А.

Рисунок 4 – Структурная схема МП КР580ВМ80А

Микропроцессор имеет следующие функциональные узлы: арифметико-логическое устройство (АЛУ), устройство управления (УУ), дешифратор команд и схему кодирования цикла, блок регистров со схемой выборки регистра к мультиплексором, регистр команд, аккумулятор (А-регистр), регистр флажков (F-регистр), регистр временного хранения на входе АЛУ, схему десятичной коррекции, буферы шины данных и шины адреса.

Блок регистров организован в шесть 16-битных регистров: программный счетчик PC, указатель стека SP, регистр временного хранения W и Z

и три пары регистров общего назначения В, С, D, E, H, L. Программный счетчик хранит адрес текущей ячейки памяти. Указатель стека адресует последнюю занятую ячейку стека. Регистр А является главным регистром МП. Все основные арифметические и логические операции производятся с использованием этого регистра.

Арифметико-логическое устройство представляет собой сложную двухвходовую комбинационную схему, которая сигналами от УУ коммутируется на выполнение определенной операции.

Шина данных МП, состоящая из 8 электрических линий (D0-D7), предназначена для передачи слова информации (1 байт). Она является двунаправленной, т.е. возможен как вывод информации из МП, так и ввод.

Состав шины управления зависит от способов организации и уровня сложности МПС. Однако во всех системах обязательно наличие таких сигналов как:

- RESET – (сброс) сигнал установки всех устройств системы в исходное состояние;
- MEMR – сигнал чтения информации из памяти;
- MEMW – сигнал записи информации в память;
- С – тактовый сигнал;
- IOR – сигнал чтения данных из устройства ввода информации в МП;
- IOW – сигнал записи данных из МП в устройство вывода информации.

Система команд МП

Ячейки памяти и порты ввода-вывода можно рассматривать как регистры, и поэтому любая МП-система состоит из огромного числа регистров, а функционирование ее заключается в передаче данных между регистрами. Порядок передач, часть из которых сопровождается преобразованием данных, определяется последовательностью команд программы. Команды состоят из двух полей: поля операции (оператор), которое опре-

деляет выполняемое действие или операцию, и поля операнда – данные, участвующие в операции. Длина оператора составляет 1 байт, и содержимое его называется кодом операции (КОП).

Команда МП может быть одно-, двух- или трехбайтной (рисунок 5). Первый байт представляет собой код операций, остальные – данные. Система команд может быть разбита на 5 групп:

1. Команды пересылок.
2. Команды арифметических операций.
3. Команды логических операций.
4. Команды перехода.
5. Специальные команды.

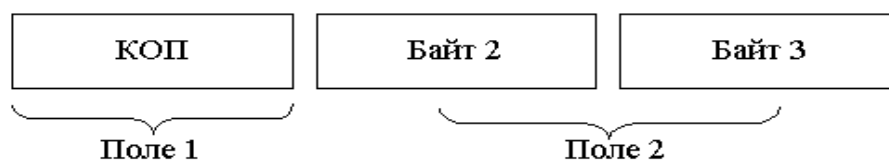


Рисунок 5 – Структура команды

В поле операнда всех команд каким-либо образом определяются данные, участвующие в операции. Способ определения операнда называется режимом адресации.

Наиболее простой, но и наиболее неэкономичной является прямая адресация: в поле операнда содержится полный 16-битный адрес памяти. С помощью прямой адресации можно обращаться к любой ячейке адресного пространства.

При необходимости включения в программу фиксированных значений применяется непосредственная адресация, в котором операндом является второй байт команды.

При использовании регистровой адресации операндом является содержимое адресуемого в команде регистра общего назначения процессора.

При выполнении 1-байтных команд с неявной адресацией подразумевается, что операнд находится в определенном внутреннем регистре МП -

аккумуляторе. При этом результат операции так же записывается в аккумулятор.

Удобным средством, позволяющим компактно адресовать все пространство памяти, является косвенная адресация. В этом режиме поле операнда определяет не регистр с данными, а указатель памяти, содержащий адрес операнда. Для быстрого выполнения команд с косвенной адресацией в качестве указателя памяти должен выступать один из внутренних 16-разрядных регистров МП. В МП КР580 основным указателем памяти служит регистровая пара HL.

2.2. Программируемый интерфейс ввода/вывода на примере КР580ВВ55А

Для связи устройств микропроцессорной системы с МП и другими устройствами системы предназначена системная шина МП, которая представляет собой набор электрических линий, сгруппированных по функциональному назначению. Разделяют следующие шины: адресная, данных и управления.

Шина адреса МП КР580ВМ80А состоит из 16-электрических линий (обозначаемых А0-А15), предназначенных для выбора необходимой ячейки памяти или устройства ввода/вывода. Шина адреса является однонаправленной (т.е. информация по ней поступает только от МП). Адресное пространство составляет 2^{16} или 65536 ячеек памяти. Для адресации устройств ввода/вывода используются лишь 8 первых адресных линий А0-А7, следовательно, возможна адресация 2^8 или 256 устройств ввода и 256 устройств вывода. Для того чтобы выбрать необходимое устройство МПС, в системе имеется дешифратор адреса ДА. Он преобразует код на шине адреса в сигнал выбора устройства (обозначается CS), который разрешает устройству обмен информацией с МП.

Микросхема КР580ВВ55А – программируемое устройство ввода/вывода параллельной информации, которая применяется в качестве элемента ввода/вывода общего назначения, сопрягающего различные типы периферийных устройств с магистралью данных систем обработки

информации. Условное графическое обозначение микросхемы приведено на рисунке 6, а структурная схема приведена на рисунке 7. Назначение выводов микросхемы показано в таблице 2.

Обмен информацией между шиной данных МПС и микросхемой осуществляется через 8-разрядный двунаправленный трехстабильный канал данных. Для связи с периферийными устройствами используются 24 линии ввода/вывода, сгруппированные в три 8-разрядных канала ВА, ВВ, ВС, направление передачи информации, режим работы которых определяются программным способом.

35	SR	ЮП	↔	
36	WR		↔	BA7 37
5	RD			BA6 38
6	CS			BA5 39
8	A0			BA4 40
9	A1			BA3 1
7	GND			BA2 2
26	U _{cc}			BA1 3
				BA0 4
				↔
			↔	BB7 25
				BB6 24
				BB5 23
				BB4 22
				BB3 21
				BB2 20
				BB1 19
				BB0 18
			↔	
			↔	BC7 10
				BC6 11
				BC5 12
				BC4 13
				BC3 17
				BC2 16
				BC1 15
				BC0 14
27	D7			
28	D6			
29	D5			
30	D4			
31	D3			
32	D2			
33	D1			
34	D0			

Рисунок 6 – Микросхема КР580ВВ55А

Микросхема может функционировать в трех основных режимах. В режиме 0 обеспечивается возможность синхронной программно-управляемой передачи данных через два независимых 8-разрядных канала ВА и ВВ и два 4-разрядных канала ВС.

В режиме 1 обеспечивается ввод и вывод информации из периферийного устройства через два независимых канала ВА и ВВ по сигналам квитирования. При этом линии канала ВС используются для приема и выдачи сигналов управления обменом.

В режиме 2 обеспечивается возможность обмена информацией с периферийными устройствами через двунаправленный 8-разрядный канал ВА по сигналам квитирования. Для передачи и приема сигналов управления обменом используется пять линий канала ВС.

Таблица 2 – Назначение выводов микросхемы КР580ВВ55А

Вывод	Обозначение	Тип вывода	Назначение
1-4	ВА3-ВА0	Вх/Вых	Канал А
37-40	ВА7-ВА4	-	-
5	RD	Вход	Чтение
6	CS	Вход	Выбор БИС
7	GND	-	Общий
8, 9	A0, A1	Выходы	Адресные линии
10-17	BC7-BC4 BC0-BC3	Вх/Вых	Канал С
18-25	ВВ0-ВВ7	Вх/Вых	Канал В
26	Ucc	-	+5 В
27-34	D7-D0	Вх/Вых	Канал данных
35	SR	Вход	Сброс
36	WR	Вход	Запись

Выбор соответствующего канала и направления передачи информации через канал определяется сигналами А0 и А1 (соединяемые обычно с младшими разрядами шины адреса системы) и сигналами RD, WR, CS в соответствии с таблицей 3.

Режим работы каждого из каналов ВА, ВВ, ВС определяется содержанием регистра управляющего слова (РУС). Произведя запись управляющего слова в РУС, можно перевести микросхему в один из режимов работы 0 – 2.

Рассмотрим режим работы 0. В этом режиме микросхема представляет собой обычную буферную схему, в которой осуществляется промежуточное хранение данных при вводе или выводе информации. В режиме 0 возможны 16 различных комбинаций ввода/вывода, которые приведены в таблице 4.

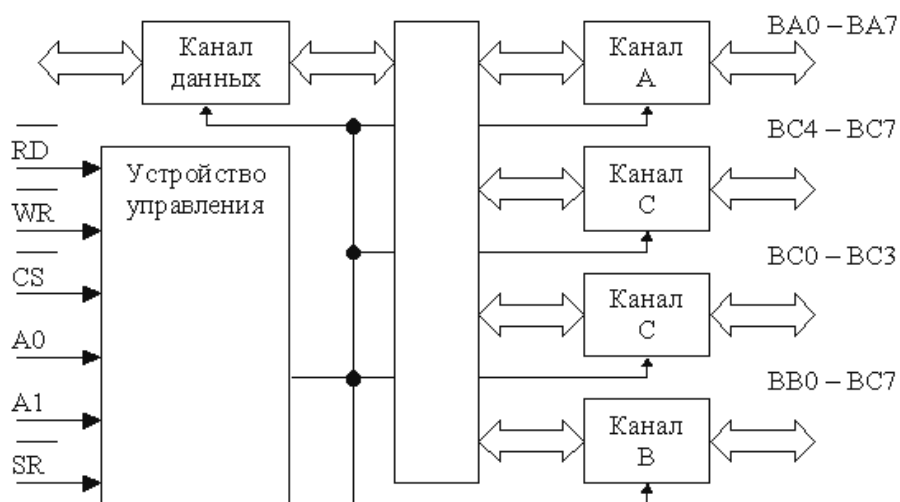


Рисунок 7 – Структурная схема микросхемы КР580ВВ55А

Таблица 3 – Выбор канала и направления передачи информации микросхемой KP580BB55A

Сигнал на входах					Направление передачи информации
A1	A0	RD	WR	CS	
Операция ввода					
0	0	0	1	0	BA → канал данных
0	1	0	1	0	BB → канал данных
1	0	0	1	0	BC → канал данных
0	0	1	0	0	канал данных → BA
0	1	1	0	0	канал данных → BB
1	0	1	0	0	канал данных → BC
1	1	1	0	0	канал данных → РУС
Операции блокирования					
X	X	X	X	1	канал данных → третье состояние
1	1	0	1	0	Запрещенная комбинация

Таблица 4 – Направление передачи в режиме 0.

Состояние разрядов управляющего слова				Направление передачи информации			
D4	D3	D2	D0	Канал BA	Канал BC 7-4	Канал BB	Канал BC 3-0
0	0	0	0	Вывод	Вывод	Вывод	Вывод
0	0	0	1	Вывод	Вывод	Вывод	Ввод
0	0	1	0	Вывод	Вывод	Ввод	Вывод
0	0	1	1	Вывод	Вывод	Ввод	Ввод
0	1	0	0	Вывод	Ввод	Вывод	Вывод

Таблица 4 (продолжение)

Состояние разрядов управляющего слова				Направление передачи информации			
D4	D3	D2	D0	Канал BA	Канал BC 7-4	Канал BB	Канал BC 3-0
0	1	0	1	Вывод	Ввод	Вывод	Ввод
0	1	1	0	Вывод	Ввод	Ввод	Вывод
0	1	1	1	Вывод	Ввод	Ввод	Ввод
1	0	0	0	Ввод	Вывод	Вывод	Вывод
1	0	0	1	Ввод	Вывод	Вывод	Ввод
1	0	1	0	Ввод	Вывод	Ввод	Вывод
1	0	1	1	Ввод	Вывод	Ввод	Ввод
1	1	0	0	Ввод	Ввод	Вывод	Вывод
1	1	0	1	Ввод	Ввод	Вывод	Ввод
1	1	1	0	Ввод	Ввод	Ввод	Вывод
1	1	1	1	Ввод	Ввод	Ввод	Ввод

3. УЧЕБНЫЙ МИКРОПРОЦЕССОРНЫЙ КОМПЛЕКТ

Учебный микропроцессорный комплект (УМК) состоит из следующих составных частей (рисунок 8): микро-ЭВМ, пульта оператора, блока питания.

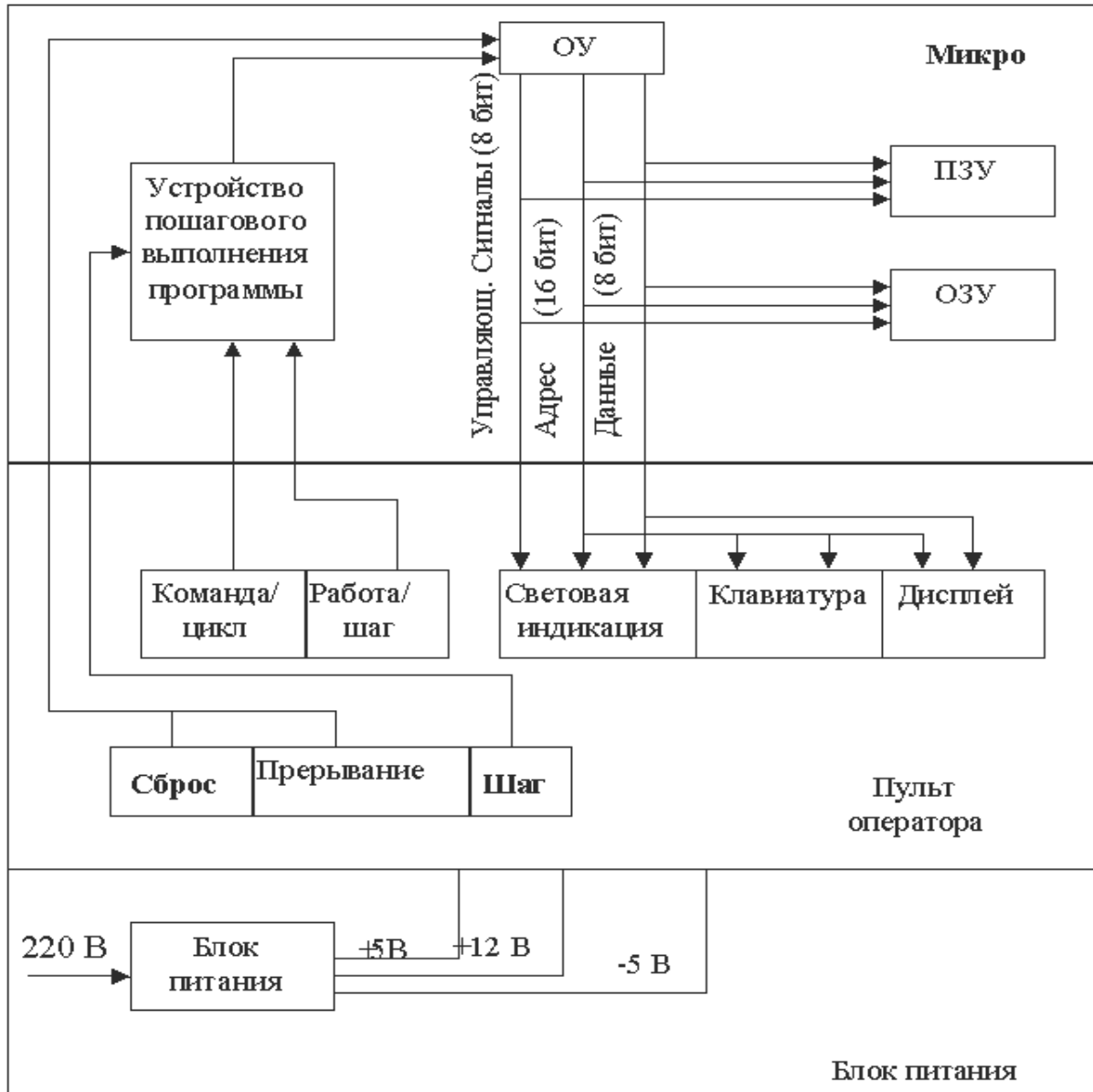


Рисунок 8 – Структурная схема УМК

Микро-ЭВМ является основной составной частью и управляет работой всего УМК. Все обращения к памяти, операции ввода – вывода, вычисления выполняются микро ЭВМ или им же инициируются.

Пульт оператора предназначен для взаимодействия оператора с микро – ЭВМ.

Блок питания обеспечивает постоянными стабилизированными напряжениями микро- ЭВМ, пульт, а также макетную плату.

Микро-ЭВМ состоит из операционного устройства (ОУ), постоянного запоминающего устройства (ПЗУ), оперативного запоминающего устройства (ОЗУ) и устройства пошагового выполнения программ.

Пульт оператора состоит из клавиатуры, шестиразрядного дисплея, световой индикации и управляющих кнопок: «кнопка сброса» («СБ»), «прерывание» («ПР»), «шаг» («ШГ»), а также переключателей: «работа / шаг» («РБ/ШГ») и «команда / цикл» («КМ/ЦК») (рисунок 8).

Основой микро-ЭВМ является ОУ, реализованное на базе МП КР 580 ВМ 80, которое производит все операции по обработке информации. Исходным состоянием ОУ является чтение информации по нулевому адресу ПЗУ. ОУ принимает это состояние после нажатия управляющей кнопки «СБ» на пульте оператора.

Информация о состоянии ОУ фиксируется в регистре состояния в начале каждого машинного цикла.

В ПЗУ записана программа «Монитор», обеспечивающая ввод информации с клавиатуры пульта оператора и вывод ее на дисплей.

Программа «Монитор» занимает 1 кбайт ПЗУ и использует последние 54 ячейки ОЗУ, еще 1 кбайт зарезервирован за пользователем. ОЗУ используется для хранения программ пользователя и имеет емкость 1 кбайт. Устройство пошагового выполнения программ переводит ОУ в состояние «Ожидание» после выполнения очередного шага. Возможны два пошаговых режима работы: покомандный и поцикловой шаг. Вызов пошагового режима работы осуществляется переключателем «РБ/ШГ», выбор величины шага – переключатели «КМ/ЦК». Для последующего шага необходимо нажать кнопку «ШГ», при этом после выполнения очередного шага на световой индикации отображается состояние адресной шины, шины данных и регистра состояния ОУ в двоичном коде.

Выполнение программы может быть остановлено нажатием управляющей кнопки «ПР». При этом состояние всех регистров ОУ сохраняется в ОЗУ, откуда они опять могут быть загружены в ОУ и выполнение программы продолжится, начиная с точки останова.

Внешний вид лицевой панели УМК отображен на рисунке 9.

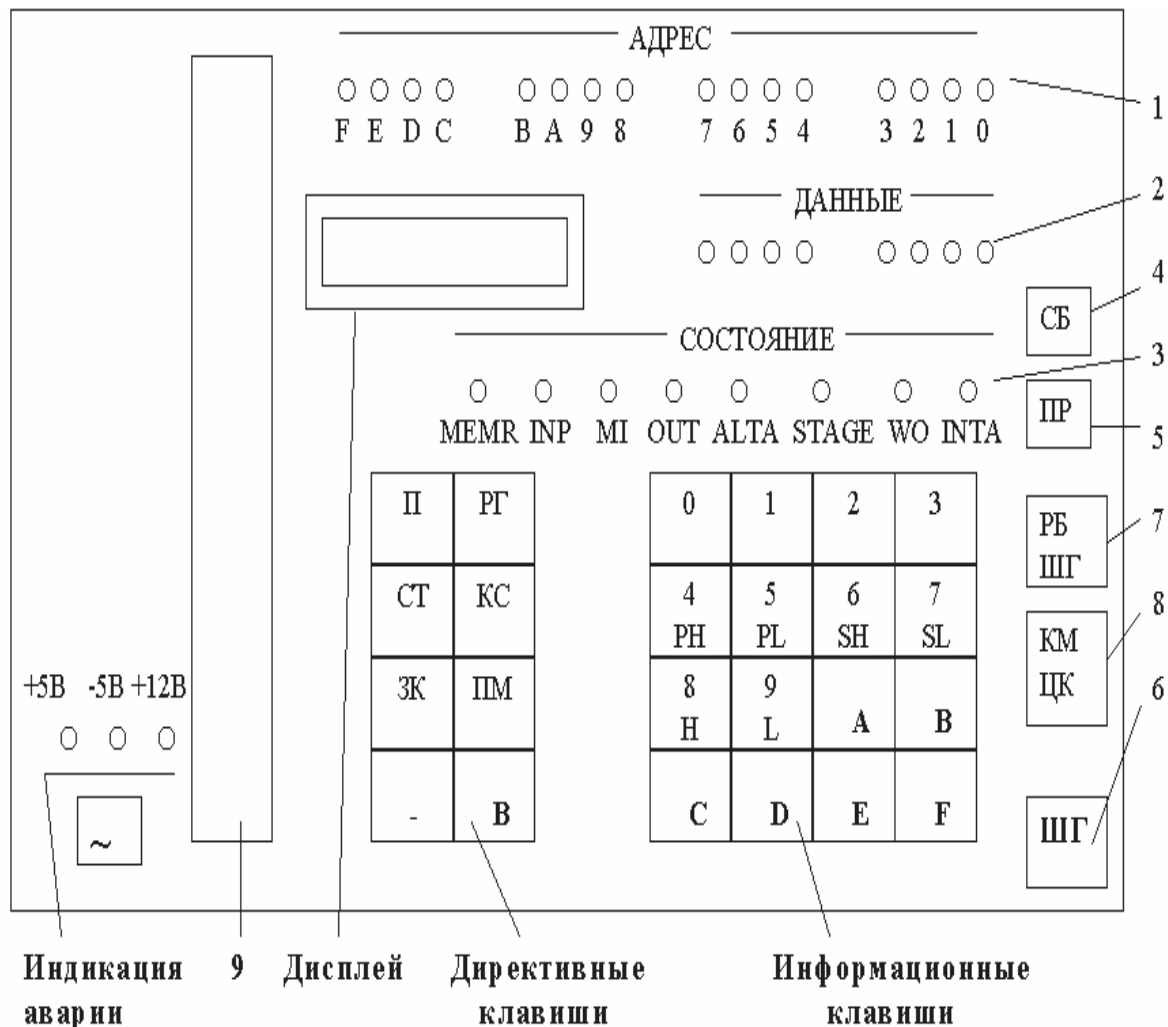


Рис. 3. Передняя панель УМК.

1- индикация шин адреса; 2 – индикация шины данных; 3 – индикация регистра состояния; 4 – СБ (сборс); 5 – ПР (прерывание); 6 – ШГ (шаг); 7 – РБ/ШГ (работа/шаг); 8 – КМ/ЦК (команда/цикл); 9 –разъем для подключения макетной платы.

Рисунок 9 – Внешний вид лицевой панели УМК

ЛАБОРАТОРНАЯ РАБОТА №1

ИЗУЧЕНИЕ УМК И ПРАВИЛ СОСТАВЛЕНИЯ ПРОСТЫХ ПРОГРАММ

Цель работы:

1. Изучение устройства и принципов функционирования УМК.
2. Освоение базовых рабочих процедур.
3. Закрепление навыков программирования на языке Ассемблер.

Устройство и принцип работы УМК

Устройство и принцип работы УМК описаны в разделе 2. Подробнее рассмотрим только функционирование устройства пошагового выполнения программы. Устройство пошагового выполнения программы переводит ОУ в состояние ожидания после выполнения очередного шага. Вызов пошагового режима работы осуществляется переключателем «РБ/ШГ», выбор величины шага переключателем «КМ/ЦК». Для последующего шага необходимо нажать кнопку «ШГ». При этом после выполнения очередного шага на светодиодном индикаторе отображается состояние адресной шины, шины данных и регистра состояния ОУ в двоичном коде.

Команды выполняются по машинным циклам (1÷5 циклов в команде). Имеется десять типов машинных циклов и соответственно десять слов состояния (смотрите таблицу 5).

Таблица 5

Состояние ОУ	Разряды регистра состояния ОУ							
	MEM	INR	MI	JUT	YLTA	STACK	WO	INTA
Выбор команды	1	0	1	0	0	0	1	0
Чтение памяти	1	0	0	0	0	0	1	0
Запись в память	0	0	0	0	0	0	0	0
Чтение стека	1	0	0	0	0	1	1	0
Запись в стек	0	0	0	0	0	1	0	0
Ввод	0	1	0	0	0	0	1	0
Вывод	0	0	0	1	0	0	0	0
Прерывание	0	0	1	0	0	0	1	1
Останов	1	0	0	0	1	0	1	0
Прерывание в останове	0	0	1	0	1	0	1	1

Перечислим органы управления.

Директивные клавиши служат для вызова директив и имеют следующие обозначения и назначения:

«П» (Память) – чтение и изменение содержимого ячеек памяти;

«РГ» (Регистр) – чтение и изменение содержимого регистров МП;

«СТ» (Старт) – запуск программы пользователя;

«КС» (Контрольная сумма) – определение контрольной суммы массива памяти: используется для контроля правильности ввода ранее отлаженной программы;

«ЗК» (Заполнение константой) – заполнение массива памяти константой;

«ПМ» (Перемещение массива) – перемещение массива в адресном пространстве памяти;

«ВП» (Выполнить) – означает конец работы с директивой;

« \square » (Пробел) – разделитель при вводе нескольких переменных.

Информационные клавиши служат для ввода адресов и данных в шестнадцатеричном коде и содержат шестнадцатеричные символы. Одновременно часть из них используется для ввода идентификаторов регистров МП:

A	– аккумулятор;
B, C, D, E	– одноименные регистры;
8/H	– регистр H;
9/L	– регистр L;
F	– регистр флагов (признаков);
4/PH	– старший байт счетчика команд;
5/PL	– младший байт счетчика команд;
6/SH	– старший байт указателя стека;
7/SL	– младший байт указателя стека.

На лицевой панели находятся пять кнопок управления:

«~» – включение / выключение УМК;

«СБ» (Сброс) – устанавливает нулевой адрес в счетчике команд;

«ПР» (Прерывание) – позволяет прервать выполнение программы;

«РБ/ШГ» (Работа/Шаг) – используется для перевода выполнения программы в пошаговом режиме;

«ШГ» (Шаг) – каждое нажатие на кнопку ведет к выполнению одного шага;

«КМ/ЦК» – переключатель дискретности шага (КМ – шаг соответствует выполнению одной команды, ЦК – шаг соответствует выполнению одного машинного цикла).

Дисплей содержит 6 разрядов: четыре левых отображают адрес, два правых информацию (в шестнадцатеричном коде).

При неправильной работе с клавиатурой в крайней левой позиции дисплея высвечивается символ «?». Нажмите кнопку «СБ» для возвращения в исходное состояние.

Светодиодная индикация состоит из набора светодиодов (Ø – свето-

диод не горит, 1 – горит). Верхний ряд отображает состояние адресной шины (16 бит).

Второй ряд отображает состояние шины данных (8 бит).

Третий ряд отображает разряды регистров состояний (8 бит).

Подготовка УМК к работе

Установите кнопку «~» в отжатое состояние.

1. Подключите УМК к сети переменного тока 220 В.
2. Переключатель «РБ/ШГ» установите в отжатое состояние «РБ».
3. Включите УМК, нажав кнопку «~».
4. Нажмите кнопку «СБ». При этом в крайней левой позиции дисплея должен появиться знак «←». После этого УМК готов к работе.

Базовые рабочие процедуры

Перед заданием директив с помощью управляющих клавиш необходимо нажать «СБ».

Индикация и изменение содержимого памяти.

Если необходимо узнать содержимое ячейки памяти ОЗУ с определенным адресом, то последовательно нажимают клавиши «П», «АДРЕС», «ВП». Где «АДРЕС» – адрес в шестнадцатеричном коде.

При этом в левых четырех разрядах дисплея высвечивается адрес, а в двух правых содержимое ячейки памяти.

Пользователю доступен массив ОЗУ с адресами 0800÷0BFF.

Если необходимо посмотреть содержимое следующей ячейки, то надо нажать клавишу «□».

Если информацию, хранящуюся в просматриваемой ячейке памяти надо изменить, то новые данные вводят с помощью информационных клавиш. При этом новые значения высвечиваются в двух правых разрядах дисплея. После этого нажимают клавишу «ВП» или для перехода к следующей ячейке клавишу «□». Директива завершается нажатием клавиши «ВП».

Пример 1: – просмотрите и измените содержимое ячеек памяти:

0800	на	3A
0801	на	00
0802	на	0B
0803	на	2F
0804	на	32
0805	на	01
0806	на	0B
0807	на	76

Индикация и изменение содержимого регистров.

Если надо узнать содержимое регистра, то последовательно нажимают клавиши:

«РГ», «Х», «ЗП»,

где «Х» – информационная клавиша идентифицирующая регистр. На дисплее высвечивается идентификатор регистра, а в двух правых разрядах его содержимое.

Идентификаторы регистров:

A	– аккумулятор;
B, C, D, E	– одноименные регистры;
8/H	– регистр H;
9/L	– регистр L;
F	– регистр флагов (признаков);
4/PH	– старший байт счетчика команд;
5/PL	– младший байт счетчика команд;
6/SH	– старший байт указателя стека;
7/SL	– младший байт указателя стека.

Все регистры по 8 бит. Если содержимое просматриваемого регистра изменять не надо, то нажимают « \square » и набирают идентификатор другого регистра. Если содержимое регистра надо изменить, то набирают новые данные в шестнадцатеричном коде (два символа) и нажимают клавишу « \square ». После этого можно набирать идентификатор следующего регистра.

Для завершения директивы нажимают клавишу «ВП».

Пример 2: – просмотрите содержимое регистров общего назначения.

Заполнение массива памяти константой.

Константа представляется в шестнадцатеричном коде двумя символами, адреса ячеек ОЗУ четырьмя символами.

Последовательность нажатия клавиш:

«ЗК» «АДРЕС 1», «□», «АДРЕС 2», «□», «К», «ВП»,

где АДРЕС 1 – адрес первой ячейки массива памяти;

АДРЕС 2 – адрес последней ячейки массива памяти;

«К» – константа в шестнадцатеричном коде (00 ÷ FF).

Необходимо помнить: заполнять только массив 0800÷0BFF.

Пример 3: – заполнить массив с адреса 0B00 до адреса 0BFF числом F8.

Проверьте содержимое соответствующих ячеек памяти. Перемещение массива памяти в адресном пространстве.

Последовательность нажатия клавиш:

«ПМ», «АДРЕС 1», «□», «АДРЕС 2», «□», «АДРЕС 3» «ВП»,

где АДРЕС 1 – начальный адрес перемещаемого массива;

АДРЕС 2 – конечный адрес перемещаемого массива;

АДРЕС 3 – новый начальный адрес массива.

Старый и новый массивы не должны перекрываться, иначе теряется информация.

Пример 4: –заполните массив с адреса 0800 до адреса 082F числом 1С;

– переместите массив на начальный адрес 0900;

– проверьте содержимое соответствующих ячеек.

Определение контрольной суммы массива памяти.

Последовательность нажатия клавиш:

«КС», «АДРЕС», «□», «АДРЕС 2», «ВП»,

где АДРЕС 1 – начальный адрес массива памяти;

АДРЕС 2 – конечный адрес массива памяти.

После выполнения директивы на дисплее высвечивается контрольная

сумма массива, представляющая собой содержимое ячеек массива по модулю 256.

Пример 5: – заполните массив с адреса 0B00÷0B10 константой 01;

- вычислите контрольную сумму массива;
- заполните массив 0B00÷0BFF константой 02;
- вычислите контрольную сумму массива.

Запуск программы пользователя.

Последовательность нажатия клавиш:

«СТ», «АДРЕС 1», «□», «АДРЕС 2», «□», «АДРЕС 3», «ВП»,

где АДРЕС 1 – начальный адрес программы;

АДРЕС 2 и АДРЕС 3 – адреса выполнения прерывания программы (могут отсутствовать).

Состояние регистров микропроцессора при достижении адресов 2 и 3 сохраняется в ОЗУ, и управление передается программе «Монитор». Пользователь может задавать любую директиву. Продолжение программы последует после повторного нажатия клавиши «ВП».

Необходимо помнить, что программа должна уместиться в пределах массива 0800÷0AFF (1 кбайт).

Пошаговое выполнение программы.

Имеются две разновидности пошагового выполнения программы: поцикловый режим «ЦК» и покомандный режим «КМ».

Для вызова пошагового режима:

- установите переключатель «РБ/ШГ» в положение «ШГ» (кнопка утоплена), при которой происходит подключение световой индикации;
- переключателем «КМ/ЦК» выберите один из режимов работы;
- передайте управление выполняемой программе.

После этого световая индикация отобразит начальный адрес программы, данные по этому адресу и содержимое регистра состояния.

Программа выполняется путем нажатия кнопки «ШГ». Для выхода из пошагового режима установите переключатель «РБ/ШГ» в состояние «РБ» (кнопка отжата) и нажмите кнопку «ШГ».

Пример 6: – запустите программу, находящуюся в памяти ОЗУ (адреса 0800÷0807) в пошаговом режиме. Следите за световой индикацией.

Прерывание выполнения программы.

Для прерывания выполнения программы необходимо нажать клавишу «ПР». После этого пользователь может вызвать выполнение любой из соответствующих директив. Выполнение прерванной программы возможно, начиная с адреса останова или любого другого адреса.

Запись и выполнение программы

Микропроцессор БИС КР580ВМ80 имеет фиксированный набор команд. Программа записывается последовательно в массив памяти ОЗУ 0800÷0AFF. Для записи данных исследуемых программ используются ячейки ОЗУ с адресами 0B00÷0BFF.

Система команд микропроцессора КР580ВМ80А приведена в Приложении.

Время выполнения команды измеряют в машинных тактах, равных периоду синхронизации (0,5 мкс при $f_T = 2$ МГц.)

Рассмотрим простейшую программу (программа 1), извлекающую число из ячейки памяти с адресом 0B00, инвертирующую его и записывающую результат в адрес памяти 0B01.

Программа 1 (в мнемокодах)

Мнемокод	Комментарий
LDA 0B00	Получить число из ячейки с адресом 0B00
CMA	Инвертировать число
STA 0B01	Записать результат по адресу 0B01
HLT	Прервать выполнение программы

При записи программ все числа представляются в шестнадцатеричной системе счисления.

Для записи программы в память необходимо перевести мнемокоды команд в машинные коды. Команды в программе могут быть одно-, двух- или трехбайтные и должны в памяти занимать соответственно один, два или три адреса.

Программа 1 (размещение по адресам памяти)

Адрес	Число	Комментарий
0800	3A	Код команды LDA
0801	00	Младший байт адреса
0802	0B	Старший байт адреса
0803	2F	Код команды CMA
0804	32	Код команды STA
0805	01	Младший байт адреса
0806	0B	Старший байт адреса
0809	76	Код команды HLT

В программе 1 используется прямой способ адресации. Приведем аналогичную программу с косвенной адресацией (программа 2).

Программа 2 (общий вид записи)

Адрес	Машин- ный код	Мнемо- код	Комментарий
0800	21 00 0B	LXI H 0B00	Записать в регистры H,L чис- ло 0B00
0803	7E	MOV A,M	Получить число из адреса указанного в регистрах H,L
0804	2F	CMA	Инвертировать число в акку- муляторе
0805	23	INX H	Увеличить на 1 число в реги- страх H,L
0806	77	MOV M,A	Записать число из аккумуля- тора по адресу указанному в регистрах H,L(0B)
0807	76	HLT	Прервать выполнение про- граммы.

Задание для домашней подготовки

1. Изучите состав и работу УМК.
2. Ознакомьтесь с языком программирования и структурой команд МП БИС КР580ВМ80.
3. Изучение команды пересылки и загрузки, команды арифметических и логических операций.
4. Определить результат выполнения программы 1 при записи различных однобайтовых команд по адресу 0803, и занести их в таблицу 6.

Таблица 6 – Результаты выполнения программы 1

Команда записанная по адресу 0803	Число записанное по адресу 0B00	Число записанное по адресу 0B01
CMA		
ADDA		
SUBA		
ANAA		
XRAA		
ORAA		
CMPA		
INRA		
DCRA		

Примечание: по адресу 0B00 запишите число соответствующее вашему порядковому номеру в журнале.

Разработайте программы:

Программа 3: увеличение на 5 числа, записанного по адресу 0B00 и записи результата по адресу 0B01 (программа 3);

Программа 4: сложение чисел, записанных по адресу 0B00 и 0B01, запись результата по адресу 0B02 (программа 4);

Программа 5: сравнение чисел, записанных по адресу 0B00 и 0B01, и записи большего из них в ячейку по адресу 0B02 (программа 5).

Экспериментальная часть

Работа с массивом данных.

1. Запишите по адресу 0B00 число 1.
2. Запишите по адресу 0B01 число 2.

3. Вычислите контрольную сумму и запишите ее по адресу 0B02.
4. Вычислите новую контрольную сумму и запишите ее по адресу 0B03 и т.д.
5. Последняя контрольная сумма вычисляется для массива 0B00÷0B06. Чему она равна?
6. Перенесите сформированный массив в область памяти с начальным адресом 0B50.
7. Определите контрольную сумму сформированного массива.
8. Запишите конбанту (14)10 в область памяти 0B07÷0B0F.
9. Определите контрольную сумму этого массива.
10. Запишите полученную контрольную сумму в регистр В.

Запись и выполнение простых программ.

Задание 1. Исследуйте программу 1.

Порядок выполнения задания;

1. Запишите программу 1 в ОЗУ.
2. Запишите по адресу 0B00 число.
3. Осуществите пуск программы с адреса 0800.
4. Проверьте результат выполнения программы путем вывода на дисплей числа, записанного по адресу 0B01 (предварительно нажмите кнопку «СБ»).
5. Исследуйте процесс выполнения команд в программе 1 по машинным циклам. Заполните таблицу:

Адрес	Команда	Циклы				
		1	2	3	4	5

6. Заменяя в программе 1 команду СМА на команды, приведенные в п. 4 домашнего задания, исследуйте результаты их выполнения.

Задание 2. Исследуйте программу 2. Порядок выполнения задания то же, что и в задании 1.

Задание 3. Исследуйте программу 3.

Порядок выполнения задания:

1. Введите программу.
2. Осуществите пуск и проверьте результат ее выполнения по числу, записанному по адресу 0B01 при числе FE, записанному по адресу 0B00.

Задание 4. Исследуйте программу 4.

Порядок выполнения задания.

1. Введите программу.
2. Проверьте результат выполнения программы по числу, записанному по адресу 0B02, записав по адресу 0B00 и 0B01 соответственно числа 0B и B0.

Задание 5. Исследуйте программу 5.

Порядок выполнения задания:

1. Введите программу.
2. Запишите по адресам 0B00 и 0B01 исследуемые числа.
3. Осуществите пуск и проверьте результат выполнения программы.

Содержание отчета

Отчет должен содержать:

1. Результаты выполнения задания для домашней подготовки.
2. Результаты выполнения и проверки пунктов экспериментальной части.
3. Алгоритмы, описывающие работу программ.

Контрольные вопросы

1. Какие машинные циклы Вы знаете?
2. Как определить время выполнения программы?
3. Какие способы адресации операндов в памяти Вы знаете?
4. Преобразуйте число 79 из десятичной системы счисления в двоичную, а затем в шестнадцатеричную.
5. Как выполняются основные рабочие процедуры в УМК?
6. Как исследовать процесс выполнения команды по циклам, используя

УМК?

7. Определите время выполнения программы 1.
8. Преобразуйте число 34 из десятичной системы счисления в двоичную, а затем в шестнадцатеричную.
9. Определите время выполнения программы 2.
10. Преобразуйте число 84 десятичной системы в двоичную, а затем в шестнадцатеричную.
11. Определите время выполнения программы 3.
12. Преобразуйте число 63 десятичной системы счисления в двоичную, а затем в шестнадцатеричную.
13. Определите время выполнения программы 4.
14. Составьте программу записи числа 17 в регистр В и пересылки числа из регистра В в регистр С.

ЛАБОРАТОРНАЯ РАБОТА №2

Изучение микропроцессора КР580ВМ80А

Цель работы:

1. Изучить устройство и принцип работы микропроцессора КР580ВМ80А. Ознакомится с системой команд микропроцессора к его программной моделью. Научится составлять и выполнять простейшие программы.

Программа работы

1. Изучение архитектуры микропроцессора.
2. Изучение системы команд и способов адресации.
3. Ввод и выполнение простейших программ.

Архитектура, принцип работы, система команд и способы адресации микропроцессора подробно рассмотрены в пункте 2.1.

Выполнение работы

1. Изучите теоретическую часть работы. Ответьте на контрольные вопросы.
2. Исследование работы МП-системы в пошаговом режиме.

Экспериментальная часть

Для подготовки к работе включите УМК в сеть. Нажмите клавишу "СБ". Введите программу, приведенную в таблице 7. Эта программа считывает число из ячейки памяти с адресом 0800h, инвертирует его и записывает по адресу 0801h. Для ввода программы нажмите клавишу "П" и затем адрес первой ячейки памяти, где будет располагаться программа 0800h. После ввода последней цифры кода программы нажмите "ВП". Далее, введите в ячейку памяти с адресом 0800 любое число.

Для перевода системы в режим пошагового выполнения программы нажмите клавиши "РБ/ШГ" и "КМ/ЦК" после чего запустите программу на выполнение последовательно введя: "СТ" 0800 "ВП". Используя показа-

ния светодиодных индикаторов на лицевой панели УМК, запишите состояние шины адреса и данных в двоичном и шестнадцатеричном виде. Запишите также показания индикаторов состояния.

Нажмите клавишу "ШГ". При этом произойдет увеличение адреса на 1, Запишите новое состояние индикаторов.

Продолжайте нажимать на клавишу "ШГ" и записывать показания светодиодных индикаторов.

Таблица 7

Адрес	Код команды	Мнемоника	Комментарий
0800	3A 00 0B	LDA 0B00	Загрузить в регистр А число из ячейки памяти с адресом 0800h
0803	2F	CMA	Инвертировать число
0804	32 01 0B	STA 0B01	Запомнить содержимое аккумулятора по адресу 0801h
0807	C7	RST 0	Выход из программы

Ввод и выполнение программы суммирования чисел.

Для изучения системы команд и способов адресации рассмотрим программу сложения массива чисел. Для этого введите программу (таблица 8) в память УМК, начиная с адреса 0800h. Данная программа берет числа из массива расположенного по адресу 0800h, причем число данных в массиве заносится в регистр В микропроцессора (команда MVI B, N, где N - число чисел в массиве). Перед выполнением программы необходимо заполнить: этот массив числами.

Таблица 8

Адрес	Код команды	Метка	Мнемоника	Комментарий
0800	21 00 0B	Main	LXI H 0800	Ввод начального массива
0803	06 05		MVI B, 05	Установка счетчика чисел
0805	CD 09 08		CALL ADD	Вызов подпрограммы сложения
0808	FF		RST 7	Выход из программы
0809	AF	ADD:	XRA A	Подпрограмма сложения
080A	4F		MOV C, A	Обнуление A и C
080B	86	CNT:	ADD M	Сложить A с содержимым C
080C	D2 10 08		INC TRM	Если нет переноса перейти на TRM
080F	07		INR C	Инкрементировать C
0810	23	TRM:	INX H	Инкрементировать HI
0811	05		DCR B	Декремент счетчика B
0812	C2 0B 08		INZ CNT	Если есть еще числа – переход на CNT
0815	C9		RET	Возврат из подпрограммы

В результате выполнения программы младший байт суммы будет храниться в аккумуляторе, а старший байт – в регистре C.

Просмотрите содержимое аккумулятора и регистра C. Проверьте правильность сложения.

Содержание отчета

Отчет должен содержать:

1. Тексты программ 1 и 2.

2. Описание состояния индикаторов в ходе выполнения программы 1 и комментарий к этим состояниям.
3. Для программы 2 должен быть подготовлен массив из 5 чисел и найдена их сумма.
4. Алгоритмы, описывающие работу программ.

Контрольные вопросы

1. Устройство МП КР580ВМ80А. Перечислите сигналы микропроцессора и их назначение.
2. Какие функции выполняет МП в микропроцессорных системах ?
3. Архитектура МП. Основные функциональные узлы; АЛУ, аккумулятор, УУ, F-регистр и тд.
4. Регистры микропроцессора : назначение, разрядность, обозначение.
5. Способы адресации в МП КР580: прямая, непосредственная, косвенная, регистровая.
6. Система команд микропроцессора. Структура команды, классификация, запись программ.
7. Регистр флагов МП. Назначение флагов.

ЛАБОРАТОРНАЯ РАБОТА №3

Программируемый интерфейс ввода/вывода КР580ВВ55А.

Ввод и вывод в режиме 0.

Цель работы:

1. Изучение устройств ввода/вывода параллельной информации в микропроцессорных системах. Исследование программируемого параллельного интерфейса ввода/вывода микропроцессора КР580ВВ55А. Изучение режимов работы этого интерфейса.

Программа работы

1. Изучение структуры микропроцессорной системы и организации ввода/вывода данных. Адресация памяти и портов ввода/вывода.
2. Изучение устройства и режимов работы БИС КР580ВВ55А.
3. Ввод/вывод данных через параллельный интерфейс.

Описание лабораторной установки

Исследование работы параллельного интерфейса проведем с использованием макетной платы, входящей в состав УМК. Принципиальная схема макета показана на рисунке 10.

Для настройки интерфейса на работу в режиме 0 необходимо ввести в регистр управляющего слова соответствующую команду. В данной схеме канал ВА работает на ввод информации, а канал ВВ – на вывод. Канал ВС не используется. Тогда слово состояния, в соответствии с таблицей 3 и рисунком 11, управляющее слово будет иметь вид: 1 0 0 1 0 0 0 0, или в шестнадцатеричном виде 90h.

В схеме макета в качестве дискретных датчиков используются микропереключатели Кн1 – Кн8 (Кн1 соответствует младшему значащему разряду, а Кн8 – старшему). Замкнутому состоянию контакта соответствует логический ноль на выходе линии ВА. Резисторы сопротивлением 2,7 кОм ограничивают максимальный ток.

Для отображения состояния линии ВВ в схеме используются светодиоды VD1 – VD8. Микросхема DD2 (К555ЛН1) представляет собой набор инверторов и выполняет функции усилителя тока.

В адресном пространстве УМК микросхеме параллельного порта ввода/вывода КР589ВВ55А соответствуют адреса 80 – 83h. По адресу 80h осуществляется доступ к каналу ВА, по адресу 81h – к каналу ВВ, по адресу 82h – к каналу 83h и по адресу 83h производится запись управляющего слова.

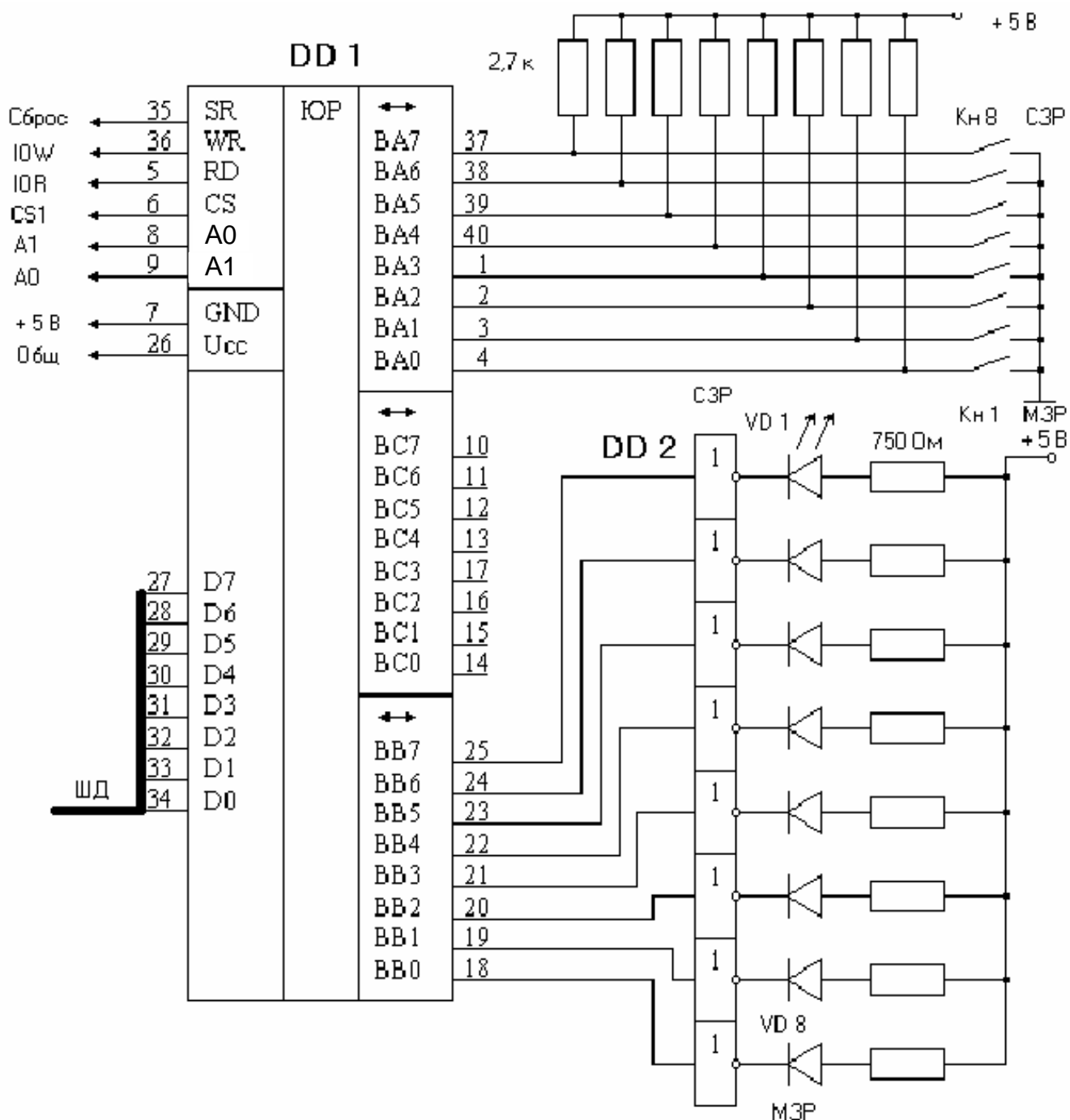


Рисунок 10 – Принципиальная схема макета лабораторной установки

D7	D6	D5	D4	D3	D2	D1	D0
----	----	----	----	----	----	----	----

Всегда 1	0 Для ре-	0 Для ре-	Канал ВА	Канал ВС	0 Для ре-	Канал ВВ	Канал ВС
----------	-----------	-----------	----------	----------	-----------	----------	----------

Рисунок 11 – Формат управляющего слова для режима 0 (если канал настраивается на ввод, то соответствующий разряд управляющего слова устанавливается в 1).

Выполнение работы

1. Ввод данных с микропереключателей

Для ввода данных с микропереключателей введите программу, приведенную в таблице 9, в память УМК, начиная с адреса 0800h. Переключите контакты на микропереключателе произвольным образом. Запустите программу на выполнение. Микропроцессор произведет считывание переключателей и сохранит байт информации о состоянии в регистре А. Не нажимая на сброс «СБ» просмотрите содержимое регистра А, нажав последовательно «РГ» «А» «ВП».

Запишите число содержащееся в регистре А. Сравните его с положением переключателей.

2. Вывод данных на индикаторы

Для вывода данных на светодиоды введите программу, приведенную в таблице 10 в память УМК, начиная с адреса 0800h. Запустите программу на выполнение. Запишите показания светодиодов. Переведите двоичный код индикаторов в шестнадцатеричный.

Введите по адресу 0805h любое другое число в шестнадцатеричном виде. Запустите программу. Проверьте правильность ввода в порт по показаниям светодиодов.

Таблица 9 – Программа для ввода данных с микропереключателей

Адрес	Код команды	Мнемоника	Комментарий
0800	3E 90	MVI A, 90	Запись в аккумулятор управляющего слова
0802	D3 83	OUT 83	Передача управляющего слова в порт
0804	DB 80	IN 80	Ввод данных из канала ВА
0806	FF	RST 7	Выход из программы

Таблица 10 – Программа для вывода данных на светодиоды.

Адрес	Код команды	Мнемоника	Комментарий
0800	3E 90	MVI A, 90	Ввод управляющего слова в регистр А микропроцессора
0802	D3 83	OUT 83	Вывод управляющего слова в порт
0804	3E FF	MVI A, FF	Ввод кода в аккумулятор
0806	D3 81	OUT 81	Вывод данных в канал ВВ
0808	FF	RST 7	Выход из программы

Экспериментальная часть

1. Ввод и вывод данных в режиме 0

Для более детального исследования возможностей ввода и вывода через порт КР580ВВ55А напишите и введите в УМК программу, которая считывает состояние микропереключателей и зажигает соответствующие разряды светодиодов.

Содержание отчета

Отчет должен содержать:

1. Тексты программ 1 и 2.
2. Алгоритмы, описывающие работу программ.
3. Результаты выполнения и проверки пунктов экспериментальной части.
4. Алгоритмы, описывающие работу программ.

Контрольные вопросы

1. Структура микропроцессорных систем. Основные элементы и функционирование.
2. Адресация памяти и устройств ввода/вывода. Какой объем памяти и сколько портов ввода/вывода может адресовать МП КР580ВМ80А?
3. Назначение интерфейса ввода/вывода параллельной информации КР580ВВ55А. Структура и режимы работы микросхемы.
4. Организация ввода параллельной информации в МПС с использованием параллельного интерфейса.
5. Организация вывода параллельной информации из МПС с использованием параллельного интерфейса.
6. Назначение и формат управляющего слова параллельного порта ввода/вывода.
7. Основные режимы работы интерфейса ввода/вывода КР580ВВ55А.

ПРИЛОЖЕНИЕ

Таблица П 1 – Система команд микропроцессора КР580ВМ80

Обозначение	Содержание		Обозначение	Содержание								
	Код	Обозначение регистра	C	Признак наличия переноса из 7-го разряда								
	000	B	Z	Признак нулевого результата								
	001	C	S	Признак старшего разряда результата								
	010	D	P	Признак четности								
	011	E	V	Признак наличия переноса из 3-го разряда								
	100	H	@	Содержимое регистра								
	101	L	(Di; Di+1)	Содержимое ячейки памяти с адресом (Di; Di+1)								
	110	M (память)	\wedge	Логическое «И»								
	111	A (аккумулятор)	\vee	Исключающее «ИЛИ»								
<B ₂ >	Второй байт команды		\vee	Логическое «ИЛИ»								
<B ₃ >	Третий байт команды		(Am)	m-й бит содержимого аккумулятора (m=0 – 7)								
DDD	Код регистра приемника		CK	Счетчик команд								
SSS	Код регистра источника		УС	Указатель стека								
			?	Пересылка								
Регистр признаков				Разряды	7	6	5	4	3	2	1	0
				Признаки	S	Z	O	V	O	P	I	C
Группа	Мнемокод	Код команды	V _B	V _T	Выполняемые действия	Признаки						
						Z	S	C	P			
Арифметические и логические операции	ADD R	10000SSS	1	4/7	(A) ? (A)+(R)	+	+	+	+			
	ADC R	10001SSS			(A) ? (A)+(R)+(C)	+	+	+	+			
	SUB R	10010SSS			(A) ? (A)-(R)	+	+	+	+			
	SBB R	10011SSS			(A) ? (A)-(R)-(C)	+	+	+	+			
	ANA R	10100SSS			(A) ? (A) \wedge (R)	+	+	0	+			
	XRA R	10101SSS			(A) ? (A) \vee (R)	+	+	0	+			
	ORA R	10110SSS			(A) ? (A) \vee (R)	+	+	0	+			
	CMP R	10111SSS			(A)-(R); ЕСЛИ (A) = (R), то z=1, если (A) < (R), то c=1	+	+	+	+			
	ADI	11000110	C6	2	7	(A) ? (A)+<B ₂ >	+	+	+	+		
	ACI	11001110	CE			(A) ? (A)+<B ₂ >+(C)	+	+	+	+		
	SUI	11010110	D6			(A) ? (A)-<B ₂ >	+	+	+	+		
	SBI	11011110	DE			(A) ? (A)-<B ₂ >-(C)	+	+	+	+		
	ANI	11100110	E6			(A) ? (A) \wedge <B ₂ >	+	+	0	+		
	XRI	11101110	EE			(A) ? (A) \vee <B ₂ >	+	+	0	+		
	ORI	11110110	F6			(A) ? (A) \wedge <B ₂ >	+	+	0	+		
	CPI	11111110	FE			(A)-<B ₂ >; ЕСЛИ (A) = <B ₂ >, то z=1, если (A) < <B ₂ >, то c=1	+	+	+	+		
	DAD B	00001001	09	1	10	(H)(L) ? (B)(C)+(H)(L)	-	-	+	-		
DAD D	00011001	19	(H)(L) ? (D)(E)+(H)(L)			-	-	+	-			
DAD H	00101001	39	(H)(L) ? (H)(L)+(H)(L)			-	-	+	-			
DAD SP	00111001	49	(H)(L) ? (B)(C)+(УС)			-	-	+	-			
CMA	00101111	29	1	4	(A) ? (A)							
Команды сдвига	RLC	00000111	1	4	D _m →D _{m+1} , D ₇ →D ₀ , D ₇ →C	-	-	+	-			
	RRC	00001111			0F	D _{m+1} →D _m , D ₀ →D ₇ , D ₇ →C	-	-	+	-		
	RAL	00010111			17	D _m →D _{m+1} , D ₇ →C, C→D ₀	-	-	+	-		
	RAR	00011111			1F	D _{m+1} →D _m , D ₀ →C, C→D ₇	-	-	+	-		
Команды обмена	XCHG	11101011	EB	1	4	(H)↔(D), (L)↔(E)						
	XTHL	11100011	E3	1	8	(L)↔[УС], (H)↔[УС+1]						

Таблица П 1 (продолжение)

Группа	Мнемокод	Код команды		V _B	V _T	Выполняемые действия	Признаки					
							Z	S	C	P		
Операции со стекком	PUSH	11SSS101		1	11	(B, D, H) → [УС-1], (C, E, L) → [УС-2], (УС)=(УС)-2	НЕ МЕНЯЮТСЯ ПРИЗНАКИ					
	PUSH PSW	11110101	F5			(A) → [УС-1], (F) → [УС-2], (УС)=(УС)-2						
	POP (B, D, H)	11DDD001		1	10	[УС] → (C, E, L); [УС+1] → (B, D, H), (УС)=(УС)+2						
	POP PSW	11110001	F1			[УС] → (F); [УС+1] → (A), (УС)=(УС)+2						+
Ввод - вывод	IN	11011011	DB	2	10	(УВВ) → (A), <B ₂ > - номер (УВВ)	НЕ МЕНЯЮТСЯ ПРИЗНАКИ					
	OUT	11010011	D3			(A) → (УВВ), <B ₂ > - номер (УВВ)						
Операции управления	STC	00110111	37	1	4	1 → (C)	-	-	+	-		
	CMC	00111111	3F			(C) → (C) инверсия C	-	-	+	-		
	EI	11111101	FD			Разрешение прерываний	НЕ МЕНЯЮТСЯ ПРИЗНАКИ					
	DI	11110011	F3			Запрет прерываний						
	NOP	00000000	00			Холостая операция						
	HLT	01110110	76			7						
	RST N	11AAA111		1	11	(СК) → [УС-1] [УС-2], (УС) → (УС) - 2 0000000000AAA000 → (СК)						

Таблица П 1 (продолжение)

Группа	Мнемокод	Код команды		V _B	V _T	Выполняемые действия	Признаки			
							Z	S	C	P
Операции пересылки и загрузки	STA	00110010	32	3	13	(A) → [$\langle B_3 \rangle$, $\langle B_2 \rangle$]				
	LDA	00111010	3A			$[\langle B_3 \rangle$, $\langle B_2 \rangle$] → (A)				
	LHLD	00101010	2A	3	16	$[\langle B_3 \rangle$, $\langle B_2 \rangle$] → (L)				
	SHLD	00100010	22			$[\langle B_3 \rangle$, $\langle B_2 \rangle + 1$] → (H)				
	MVIR	00DDD110		2	7/10	$\langle B_2 \rangle$ → (R)				
	LXI (B, D, H)	00DDD001		3	10	$\langle B_2 \rangle$ → (C, E, L), $\langle B_3 \rangle$ → (B, D, H)				
	PCHL	11101001	E9	1	5	(H)(L) → (CK)				
	SPHL	11111001	F9	(H)(L) → (YC)						
	MOV R1, R2	01DDDSSS		1	5/7	$(R_2) \rightarrow (R_1)$				
	STAX B	00000010	02	1	7	(A) → [(B)(C)]				
	STAX D	00010010	12			(A) → [(D)(E)]				
	LDAX B	00001010	0A			[(B)(C)] → (A)				
	LDAX D	00011010	1A			[(D)(E)] → (A)				
Операции инкремента декремента	INR R	00DDD100		1	5/10	$(R)+1 \rightarrow (R)$				
	DCR R	00DDD101				$(R)-1 \rightarrow (R)$				
	INX (B, D, H)	00DDD011				$(B, D, H)(C, E, L)+1 \rightarrow (B, D, H)(C, E, L)$				
	DCX (B, D, H)	00DDD011				$(B, D, H)(C, E, L)-1 \rightarrow (B, D, H)(C, E, L)$				
Операции перехода	JMP	11000011	C3	3	10	$\langle B_3 \rangle \langle B_2 \rangle \rightarrow (CK)$				
	JNZ	11000010	C2			Если z=0 Если z=1 Если c=0 Если c=1 Если p=0 Если p=1 Если s=0 Если s=1 ТО $[\langle B_3 \rangle$, $\langle B_2 \rangle + 1$] → (CK) ИНАЧЕ $(CK) = (CK) + 3$				
	JZ	11001010	CA							
	JNC	11010010	D2							
	JC	11011010	DA							
	JPO	11100010	E2							
	JPE	11101010	EA							
	JP	11110010	F2							
JM	11111010	FA								
Операции вызова подпрограмм	CALL	11001101	CD	3	17	$(CK) \rightarrow [YC-1][YC-2]$, $\langle B_3 \rangle \langle B_2 \rangle \rightarrow (CK)$, $(YC) = (YC) - 2$				
	CNZ	11000100	C4	3	11/17	Если z=0 Если z=1 Если c=0 Если c=1 Если p=0 Если p=1 Если s=0 Если s=1 ТО $(CK) \rightarrow [YC-1]$ $[YC-2]$; $\langle B_3 \rangle \langle B_2 \rangle \rightarrow$ (CK) ; $(YC) = (YC) - 2$; иначе $(CK) = (CK) + 3$ $V_T = 11$ при отсутствии перехода и $V_T = 17$ при переходе				
	CZ	11001100	CC							
	CNC	11010100	D4							
	CC	11011100	DC							
	CPO	11100100	E4							
	CPE	11101100	EC							
	CP	11110100	F4							
	CM	11111100	FC							
Возврат из подпрограмм	RET	11001001	C9							1
	RNZ	11000000	C0	Если z=0 Если z=1 Если c=0 Если c=1 Если p=0 Если p=1 Если s=0 Если s=1 ТО $[YC] [YC+1] \rightarrow$ (CK) , $(YC) = (YC) + 2$; иначе $(CK) = (CK) + 1$, $V_T = 11$ при возврате из подпрограммы						
	RZ	11001000	C8							
	RNC	11010000	D0							
	RC	11011000	D8							
	RPO	11100000	E0							
	RPE	11101000	E8							
	RP	11110000	F0							
RM	11111000	F8								

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное
образовательное учреждение высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ
УНИВЕРСИТЕТ»**

Невинномысский технологический институт (филиал)

**Методические указания по выполнению лабораторных
работ**

по дисциплине

«Микропроцессорные системы управления»

**15.03.04 «Автоматизация технологических процессов»
ЧАСТЬ - II**

Невинномысск, 2020

Методические указания предназначены для студентов очной и заочной форм обучения специальностей 15.03.04 «Автоматизация технологических процессов». Они содержат основы теории, описание опытных установок, порядок проведения лабораторных работ и обработки экспериментальных данных, перечень контрольных вопросов для самоподготовки и список рекомендуемой литературы. Объем и последовательность выполнения работ определяются преподавателем в зависимости от количества часов, предусмотренных учебным планом дисциплин как для очной, так и для заочной форм обучения.

Методические указания разработаны в соответствии с требованиями Государственного образовательного стандарта в части содержания и уровня подготовки бакалавров по направлению 15.03.04 «Автоматизация технологических процессов».

Составители: канд. техн. наук Карабак Ю.В.,
Евдокимов А.А.,
Щербина С.Ю.,
Ковалев М.В.,
Топчиев А.В.

Ответственный редактор: канд. техн. наук Болдырев Д.В.

Содержание

1. Общие сведения об элементах мп систем	68
2. Комплект лабораторного оборудования по электронной технике типа к32	93
Лабораторная работа №4	95
Лабораторная работа №5	98
Лабораторная работа №6	100
Лабораторная работа №7	103

Условные обозначения, используемые в методических указаниях:

ОЗУ – оперативное запоминающее устройство;

АЦП – аналого-цифровой преобразователь;

ЦАП – цифро-аналоговый преобразователь;

ПК – программатор кодов;

ЗУ – запоминающее устройство;

БУК – блок управления комплектом;

МЗР – младший значимый разряд;

СЗР – старший значимый разряд;

УС – устройство сменное;

ИС – интегральная микросхема.

ПСИ – программатор серии импульсов;

ПК – программатор кодов;

ГПИ – генератор прямоугольных импульсов;

ОУ – операционный усилитель;

ЦОС – цепь отрицательной обратной связи;

ИОН – источник опорного напряжения.

1. ОБЩИЕ СВЕДЕНИЯ ОБ ЭЛЕМЕНТАХ МП СИСТЕМ

1.1. Оперативное запоминающее устройство на примере ИС типа К155РУ2.

Функциональные возможности любой ЭВМ существенно зависят от объемов и структуры ее запоминающих устройств, или, иначе говоря, от памяти машины. Память современных вычислительных машин представляет собой комбинацию вариантов запоминающих устройств: магнитных накопителей на различных носителях (лента, жесткий и мягкий диск, компакт-кассета и т.д.), полупроводниковых запоминающих устройств (ЗУ).

Цифровые ЗУ предназначены для записи, хранения и выдачи информации, представленной в виде цифрового кода.

Основными характеристиками ЗУ являются: их информационная емкость, быстродействие и время хранения информации.

По выполняемым функциям различают следующие типы ЗУ: оперативные запоминающие устройства (ОЗУ, или RAM); постоянные запоминающие устройства (ПЗУ, или ROM); программируемые запоминающие устройства (ППЗУ, или PROM); репрограммируемые ПЗУ с возможностью многократного электрического перепрограммирования (РПЗУ, или EEPROM); РПЗУ с ультрафиолетовым стиранием и электрической записью информации (РПЗУ УФ или EPROM). Кроме того, существуют ассоциативные запоминающие устройства (АЗУ или CAM) – «безадресные» ЗУ, в которых поиск и выборка информации осуществляется по содержанию произвольного количества разрядов, хранящихся в АЗУ чисел, независимо от физических координат ячеек памяти.

ОЗУ – устройство памяти цифровой информации, объединенные со схемами управления, обеспечивающими режимы записи, хранения и считывания цифровой (двоичной) информации в процессе ее обработки. По способу хранения информации ЗУ делятся на статические и динамические. В динамических ЗУ для хранения информации используют инерци-

онные свойства реактивных элементов (например, емкости затвор-исток), что требует периодического восстановления (регенерации) состояния элементов памяти в процессе хранения информации. Элементы памяти статических ЗУ представляют собой бистабильные ячейки, что определяет потенциальный характер управляющих сигналов и возможность чтения информации без ее разрушения. Статические ЗУ бывают синхронными и асинхронными. Синхронные статические ЗУ имеют статический накопитель (матрицу элементов памяти) и динамические цепи управления, требующие синхронизации аналогично динамическим ЗУ.

Все параметры ЗУ также можно разделить на статические и динамические. Статические параметры ЗУ характеризуют его работу в установившемся режиме. Система статических параметров ЗУ представляет собой совокупность некоторых контрольных точек его вольт-амперной характеристики (ВАХ). Динамические параметры ЗУ определяются происходящими в нем временными процессами. Систему динамических параметров ЗУ составляет совокупность временных переходов входных и выходных сигналов, соответствующих границам правильного функционирования ЗУ. Структурная схема статического ОЗУ приведена на рисунке 1.

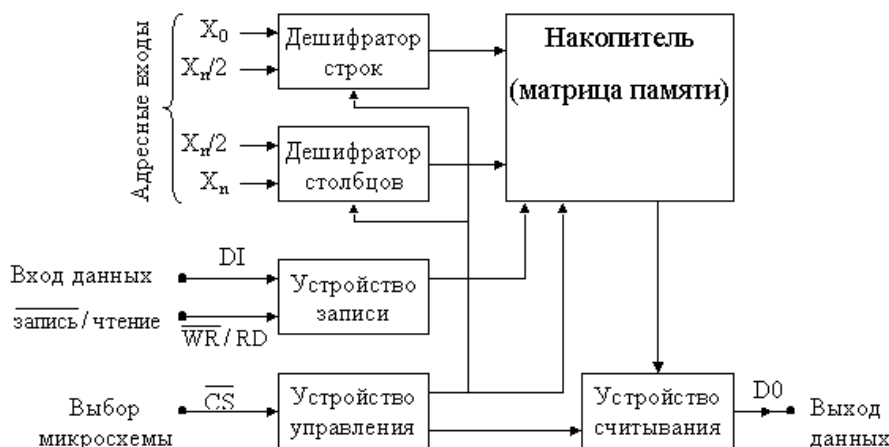


Рисунок 1 – Структурная схема статического ОЗУ

В данных методических указаниях рассмотрим статическое ОЗУ в интегральном исполнении серии К155РУ2. Микросхема РУ2 является

ОЗУ со схемами управления и полной дешифрацией адреса. Общий объем памяти 64 бита (организация – 16×4-разрядных слов). Логическая структура микросхемы приведена на рисунке 2.

Микросхема включает матрицу из 64 ячеек памяти, 4 усилителя записи-считывания, стробируемый дешифратор 4-разрядного двоичного кода адреса, выбирающий одно из 16 4-разрядных слов. Вход разрешения выборки несет функцию запрета при подаче на него напряжения высокого уровня.

В этом случае нельзя произвести ни запись в запоминающую ячейку, ни считывание из нее. Запись информации осуществляется прямым кодом параллельно по четырем информационным входам D1-D4 по требуемому адресу (рисунок 2). На входы ИС разрешения выборки и разрешения записи должен быть подан низкий уровень. Работа ИС описывается таблице 1.

Таблица 1 – Работа ИС

CS	W	D	Режим работы	C
1	1	0	Хранение	1
0	1	0	Считывание	Инверсный код выбранного числа
0	0	1	Запись 1	0
0	0	0	Запись 0	1
1	0	1	Запрет	0
1	0	0	Обращения	1

Считывание информации осуществляется при подаче на входы адреса A1-A4 прямого кода и на вход разрешения выборки (\overline{CS}) низкого уровня напряжения. На входе разрешения записи (\overline{W}) должен быть подан высокий уровень напряжения.

Записанная информация считывается параллельно по четырем выходам С1-С4 в инверсном коде. Выходные каскады усилителей считывания выполнены по схеме с открытым коллектором, что дает возможность информационные выходы различных ИС объединять в монтажное ИЛИ.

Достоверное считывание информации после записи может осуществляться через время $t_{вос.зп}$, в течение которого происходит восстановление усилителей записи и считывания.

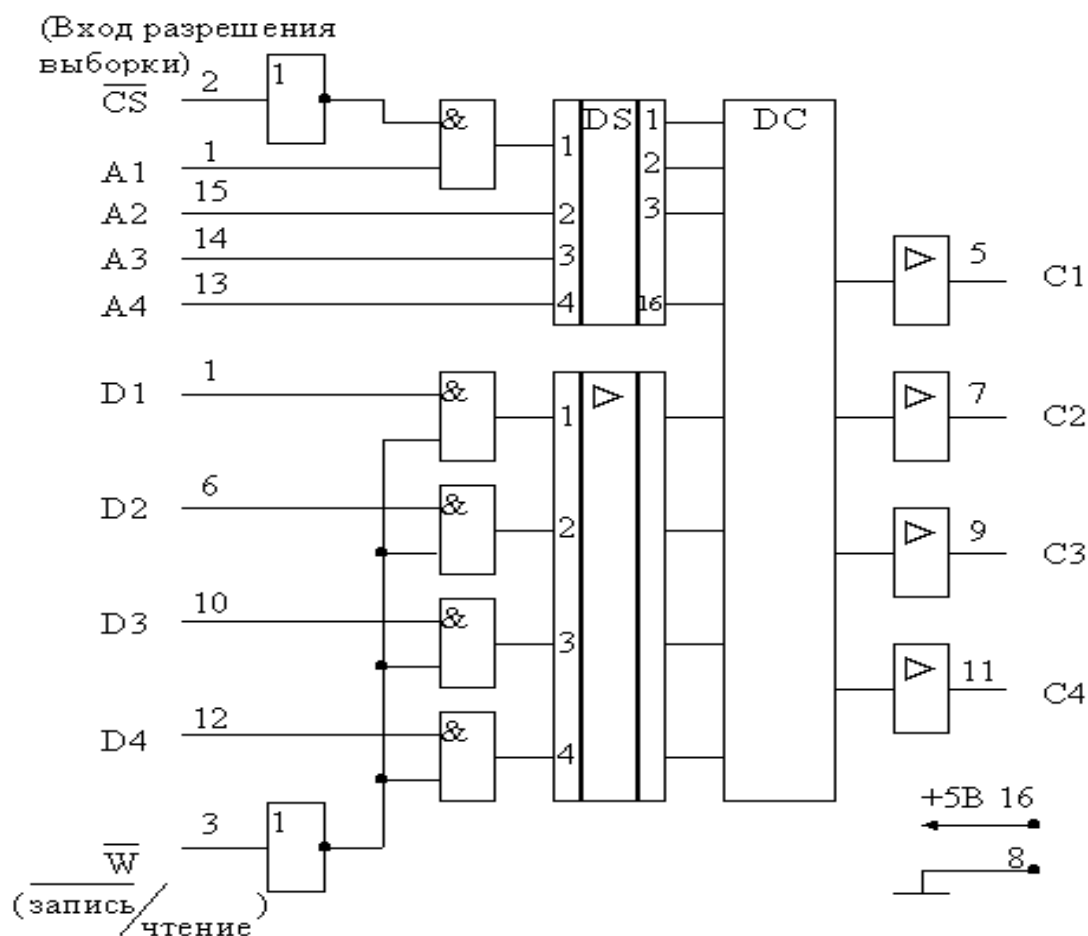


Рисунок 2 – Логическая структура микросхемы РУ2

1.2. Постоянное запоминающее устройство на примере ИС типа КР1601РР1

Постоянные запоминающие устройства (ПЗУ или ROM) – матрицы пассивных элементов памяти со схемами управления, предназначенные

для воспроизведения неизменной информации, заносимой в матрицу при изготовлении (в режиме хранения информации энергия не потребляется).

Программируемые постоянные запоминающие устройства (ППЗУ или PROM) – постоянные запоминающие устройства с возможностью однократного электрического программирования. Они отличаются от ПЗУ тем, что позволяют в процессе применения микросхемы однократно изменить состояние запоминающей матрицы электрическим путем по заданной программе.

Репрограммируемые постоянные запоминающие устройства РПЗУ (EEPROM) – постоянные запоминающие устройства с возможностью многократного электрического перепрограммирования. Они отличаются от ППЗУ тем, что допускают многократную электрическую запись информации, но число циклов записи и стирания ограничено (до 10^4 циклов).

Репрограммируемые постоянные запоминающие устройства с ультрафиолетовым стиранием и электрической записью информации (РПЗУ УФ или EPROM). Они отличаются от ПЗУ только способом стирания информации с помощью ультрафиолетового освещения, для чего в корпусе микросхемы имеется специальное окно.

ПЗУ состоит из ячеек, обратившись к которым можно вывести их содержимое. Отличие от ОЗУ состоит в том, что информация в ячейки записывается однократно, после чего в процессе эксплуатации используется лишь режим чтения.

По способу занесения информации ПЗУ делятся на два вида: ПЗУ, программируемые маской на предприятии-изготовителе, и ПЗУ, программируемые пользователем.

Во-первых, информация заносится в процессе изготовления микросхемы с помощью соответствующего фотошаблона. Очевидно такой способ записи пригоден в тех случаях, когда производится выпуск крупной партии ПЗУ с одной и той же записанной в них информацией. Промышленность выпускает такие ПЗУ, например, для использования в ка-

честве преобразователя двоичного кода в определенные двоично-десятичные коды и других преобразователей. В них входная кодовая комбинация служит адресом ячейки, а содержимое ячейки – выходной кодовой комбинацией (являющейся, например, кодовой комбинацией двоично-десятичного кода).

В ПЗУ, программируемых пользователем, запись информации производится непосредственно пользователем с помощью специальных устройств, называемых программаторами. Программатор выдает в микросхему соответствующие напряжения для записи информации, набираемой на клавиатуре, либо предварительно нанесенной путем пробивок на перфоленту. Этими напряжениями осуществляется прожигание плавких перемычек в элементах памяти. Очевидно однажды записанная в ПЗУ информация, в дальнейшем не может быть изменена. При необходимости изменить содержимое ПЗУ микросхемы с ранее записанной информацией заменяются новыми, в которые записываются новые данные.

Основная отличительная особенность микросхем РПЗУ заключается в их способности к многократному (от 100 до 10 тыс.) перепрограммированию самим пользователем. Это свойство микросхем обеспечено применением ЭП со свойствами управляемых «перемычек», функции которых выполняют транзисторы со структурой МНОП (металл Al-нитрид кремния Si_3N_4 – окисел кремния SiO_2 – полупроводник Si) и транзисторы п-МОП с плавающим затвором (ПЗ) с использованием механизма лавинной инжекции заряда ЛИЗМОП.

Всю номенклатуру выпускаемых микросхем РПЗУ можно разделить на две группы: РПЗУ с записью и стиранием электрическими сигналами (группа ЭС) и РПЗУ с записью электрическими сигналами и стиранием ультрафиолетовым излучением (группа УФ).

Микросхемы РПЗУ-ЭС содержат ЭП типа МНОП (К558, К1601) и ЛИЗМОП с двойным затвором (К573РР2, К1609РР1 и др.).

Микросхемы РПЗУ с ЭП на р-МНОП транзисторах КР558РР1, КР1601РР1, КР1601РР3 имеют сравнительно низкое быстродействие, вы-

сокое напряжение программирования (30...40 В) и требуют двух источников питания.

Для улучшения характеристик РПЗУ широко применяют технологию изготовления ЭП на n-МНОП транзисторах. Такие ЭП устроены аналогично рассмотренным, но имеют проводимость подложки p-типа, а истока и стока – n-типа. Микросхемы с ЭП на n-МНОП транзисторах КР558РР2, КР558РР3, К16111РР1 обладают втрое превосходящим быстродействием, сниженным до 22 В напряжением программирования, и работают от одного источника питания.

Устройство, принцип действия, режимы управления работой микросхем РПЗУ разных групп во многом аналогичны. Например, микросхемы К558РР2, К1609РР1, К573РР2, К573РФ2 емкостью 2Кх8 бит, относящиеся к разным группам РПЗУ по типу элемента памяти, имеют похожую структуру и одинаковую разводку выводов корпуса. Отличие между микросхемами групп ЭС и УФ состоит в способе реализации режима стирания.

Принцип построения и режим работы РПЗУ КР1601РР1 емкостью 1Кх4 с ЭП на p-МНОП транзисторах

Структурная схема (рисунок 3) содержит все элементы, необходимые для работы микросхемы в качестве ПЗУ: матрицу с элементами памяти, дешифраторами кода адреса строк и столбцов, селектор (ключи выбора столбцов), устройство ввода-вывода УВВ. Кроме того, в структуре предусмотрены функциональные узлы, обеспечивающие ее работу в режимах стирания и программирования (записи информации) – это коммутаторы режимов и формирователи импульсов напряжений требуемой амплитуды и длительности из напряжения программирования U_{PR} . По сравнению с микросхемами ПЗУМ и ППЗУ система управляющих сигналов дополнена сигналами программирования \overline{PR} и стирания \overline{ER} . Накопитель с матричной организацией содержит 64 строк и 64 столбцов, на пересечениях которых расположены 4096 элементов памяти. Управление накопителем

осуществляют шестью старшими разрядами адресного кода, который после дешифрирования выбирает строку с 64 элементами памяти. Сигналы, считанные с элементов выбранной строки, поступают на входы селектора, назначение которого состоит в выборе из 64-разрядного кода на входах четырех разрядов, которые далее поступают через УВВ на выходы микросхемы. Селектором управляют четыре младших разряда адресного кода, которые после дешифрирования обеспечивают выборку одного четырехразрядного слова из 16 слов, содержащихся в выбранной строке. Устройство управления под воздействием сигналов на своих входах обеспечивает работу микросхемы в одном из следующих режимов: хранения, считывания, стирания, записи (программирования). Управляющие сигналы имеют следующее назначение: CS - выбор микросхемы; \overline{PR} - разрешение на режим записи (программирования); U_{PR} - напряжение программирования; RD - сигнал считывания; \overline{ER} - сигнал стирания информации. Входы сигналов инверсные, поэтому разрешающим значением этих сигналов является 0. Многие микросхемы группы ЭС допускают избирательное стирание по адресу.

Процесс стирания начинается с момента подачи импульса \overline{ER} , который должен иметь длительность 200 мс. По окончании стирания все ЭП матрицы переходят в состояние, соответствующее логической 1. В этом режиме сигналы на адресных и информационных выводах могут иметь произвольные значения.

Микросхема КР1601РР1 допускает построчное стирание. Этот режим отличается от рассмотренного значением сигнала $\overline{PR} = 0$, наличием на всех информационных выводах сигналов с уровнем 1, а на адресных входах – сигналов адреса строки $A_4 - A_9$, по которому следует стереть информацию из всех 64 ЭП. Время избирательного стирания то же, что и общего.

Режим хранения обеспечивают сигналом $CS = 1$, запрещающим обращение к микросхеме независимо от значений сигналов на других входах. Возможен второй вариант обеспечения режима хранения при использовании импульсного питания напряжением минус 12 В. Такой режим позволяет уменьшать потребляемую мощность. Когда в паузах между обращениями к микросхеме отключают напряжение питания, она переходит в режим хранения. Управление переключениями питания целесообразно осуществлять сигналом CS .

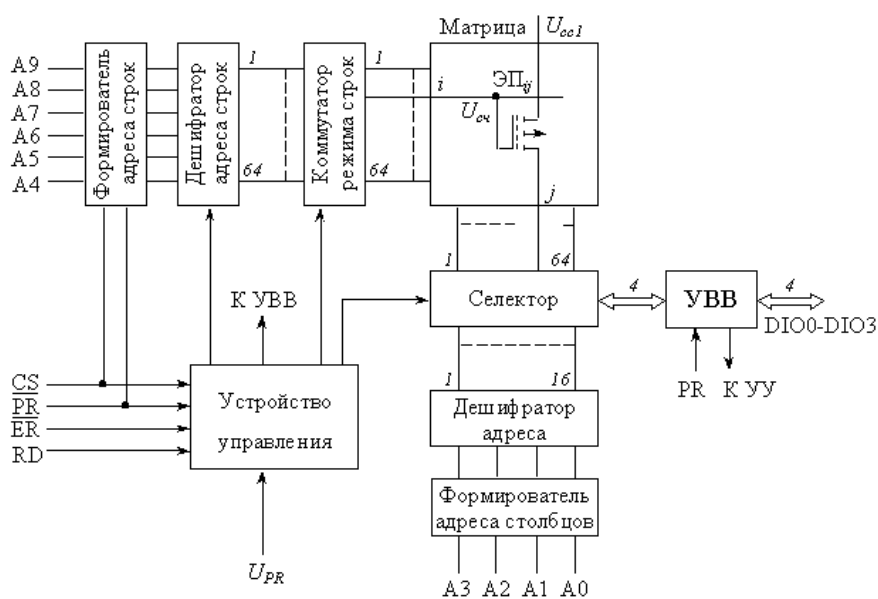


Рисунок 3 – Структурная схема РПЗ КР1601РР1

При эксплуатации микросхем РПЗУ необходимо обеспечить требуемый порядок включения и выключения напряжений питания и программирования; при включении вначале подают 5 В, затем – 12 В и последним - напряжение программирования $U_{PR} = - 32$ В. При выключении последовательность меняется на обратную. Можно все три напряжения включать и выключать одновременно.

Достоинством микросхем РПЗУ группы ЭС является возможность перепрограммирования без изъятия их из устройства, где они работают. Другим положительным свойством микросхем данной группы является значительное число циклов перепрограммирования, достигающее для большинства микросхем 10 тыс. Эти свойства в сочетании с энергонеза-

висимостью позволяют широко использовать их в аппаратуре в качестве встроенных ПЗУ со сменяемой информацией. Гарантийный срок сохранения информации при отключенном питании составляет от 3 тыс. ч. до 5 лет.

1.3. Аналого-цифровой преобразователь на примере ИС типа КР572ПВ2

Аналого-цифровой преобразователь – устройство, осуществляющее автоматическое преобразование (измерение и кодирование) непрерывно изменяющихся во времени аналоговых величин в эквивалентные значения числовых кодов.

При преобразовании напряжения в цифровой код используются три независимых операции: дискретизация, квантование и кодирование. Процедура аналого-цифрового преобразования непрерывного сигнала представляет собой преобразование непрерывной функции напряжения $u(t)$ в последовательность чисел $u(t_n)$, где $n = 0, 1, 2, \dots$, отнесенных к некоторым фиксированным моментам времени. При дискретизации непрерывная функция $u(t)$ преобразуется в последовательность ее отсчетов $u(t_n)$.

Вторая операция, называемая квантованием, состоит в том, что мгновенные значения функции $u(t)$ ограничиваются только определенными уровнями, которые называются уровнями квантования. В результате квантования непрерывная функция $u(t)$ принимает вид ступенчатой кривой $u_k(t)$.

Третья операция, называемая кодированием, представляет дискретные квантованные величины в виде цифрового кода, т.е. последовательности цифр, подчиненных определенному закону. С помощью операции кодирования осуществляется условное представление численного значения величины.

В основе дискретизации сигналов лежит принципиальная возможность представления их в виде взвешенных сумм:

$$u(t) = \sum_n a_n f_n(t), \quad (1.3.1)$$

где a_n – некоторые коэффициенты или отсчеты, характеризующие исходный сигнал в дискретные моменты времени,

$f_n(t)$ – набор элементарных функций, используемых при восстановлении сигнала по его отсчетам.

Дискретизация бывает равномерная и неравномерная. При равномерной дискретизации период отсчетов T остается постоянным, а при неравномерной – период может изменяться. Неравномерная дискретизация чаще всего обусловлена скоростью изменения сигнала и потому называется адаптивной.

В АЦП погрешность квантования определяется как единица младшего значащего разряда (МЗР).

Выходной величиной АЦП является цифровой код, т.е. последовательность цифр, с помощью которой представляются дискретные квантованные величины. В АЦП используют четыре основных типа кодов: натуральный двоичный, десятичный, двоично-десятичный и код Грея. Кроме этого, АЦП, предназначенные для вывода информации в десятичном коде, выдают на своем выходе специализированный код для управления семисегментными индикаторами.

Любой АЦП является сложным электронным устройством, которое может быть выполнено в виде одной интегральной микросхемы или содержать большое количество различных электронных компонентов. В связи с этим характеристики АЦП зависят не только от его построения, но и от характеристик элементов, входящих в его состав. Тем не менее большинство АЦП оценивают по их основным метрологическим показателям, которые можно разделить на две группы: статические и динамические.

К статическим характеристикам АЦП относят: абсолютные значения и полярности входных сигналов, входное сопротивление, значения и полярности выходных сигналов, выходное сопротивление, значения напря-

жений и токов источников питания, количество двоичных или десятичных разрядов выходного кода, погрешности преобразования постоянного напряжения и др. К динамическим параметрам АЦП относят: время преобразования, максимальную частоту дискретизации, аппретурное время, динамическую погрешность и др.

Рассмотрим некоторые из этих параметров более подробно.

Разрешающая способность является основной характеристикой АЦП, которую принято определять величиной, обратной максимальному числу кодовых комбинаций на выходе АЦП. Разрешающую способность можно выражать в процентах, в количестве разрядов или в относительных единицах. Например, 10-разрядный АЦП имеет разрешающую способность $(1024)^{-1} \approx 10^{-3} = 0,1\%$. Если напряжение шкалы для такого АЦП равно 10 В, то абсолютное значение разрешающей способности будет около 10 мВ.

Реальное значение разрешающей способности отличается от расчетного из-за погрешностей АЦП. Точность АЦП определяется значениями абсолютной погрешности, дифференциальной и интегральной нелинейности. Абсолютную погрешность АЦП определяют в конечной точке характеристики преобразования, поэтому ее обычно называют погрешностью полной шкалы и измеряют в единицах младшего разряда.

Дифференциальную нелинейность (DNL) определяют через идентичность двух соседних приращений сигнала, т.е. как разность напряжений двух соседних квантов: $DNL = h_i - h_{i+1}$.

Интегральная нелинейность АЦП (INL) характеризует идентичность приращений во всем диапазоне входного сигнала. Обычно ее определяют, как показано на рисунке 4, по максимальному отклонению сглаженной характеристики преобразования от идеальной прямой линии, т.е. $INL = u'_i - u_i$.

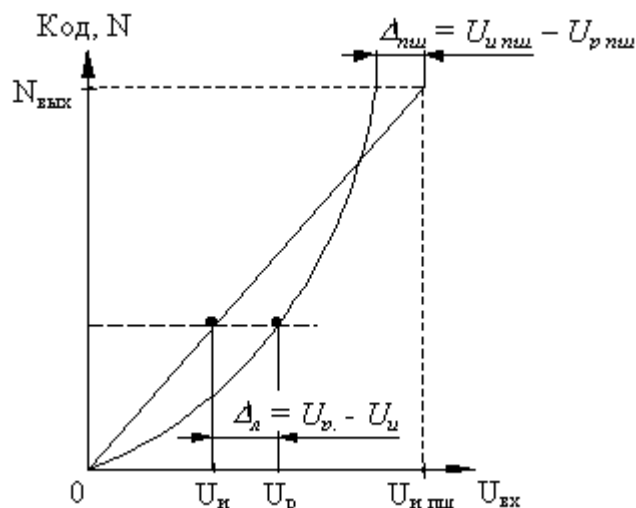


Рисунок 4 – Определение интегральной нелинейности и погрешности полной шкалы

Коэффициент нелинейности экспериментальной характеристики K_H :

$$K_H = \frac{\operatorname{tg}\alpha_{\max} - \operatorname{tg}\alpha_{\min}}{\operatorname{tg}\alpha_{\text{ср}}}, \quad (1.3.2)$$

где α_{\max} , α_{\min} , $\alpha_{\text{ср}}$ – максимальный, минимальный и средний углы наклона касательной к экспериментальной характеристике.

Степень отклонения реальной характеристики АЦП от расчетной определяется абсолютной погрешностью полной шкалы (рисунок 4).

Время преобразования $T_{\text{пр}}$ обычно определяют как интервал времени от начала преобразования до появления на выходе АЦП устойчивого кода входного сигнала. Для одних типов АЦП это время постоянное и не зависит от значения входного сигнала, для других АЦП это время зависит от значения входного сигнала. Если АЦП работает без устройства выборки и хранения, то время преобразования является апертурным временем.

Максимальная частота дискретизации – его частота, с которой возможно преобразование входного сигнала, при условии, что выбранный параметр (например, абсолютная погрешность) не выходит за заданные пределы. Иногда максимальную частоту преобразования принимают рав-

ной обратной величине времени преобразования. Однако это пригодно не для всех типов АЦП.

Все типы используемых АЦП можно разделить по признаку измеряемого значения напряжения на две группы: АЦП мгновенных значений и АЦП средних значений напряжения (интегрирующие АЦП). АЦП мгновенных значений можно разделить на следующие основные виды: последовательного счета, последовательного приближения, параллельные, параллельно-последовательные и с промежуточным преобразованием в интервалы времени.

АЦП средних значений напряжения (интегрирующий АЦП) можно разделить на следующие основные виды: с времяимпульсным преобразованием, с частотно-импульсным преобразованием и со статическим усреднением. Наибольшее распространение получили первые две группы АЦП.

Структурная схема интегрирующего АЦП с времяимпульсным преобразованием приведена на рисунке 5. Работу этой схемы можно разделить на три такта. В первом такте производится заряд интегратора, во втором – его разряд, а в третьем коррекция нулевого уровня интегратора. Графики, иллюстрирующие работу АЦП, приведены на рисунок 6.

В первом такте, имеющем фиксированную длительность T_0 замкнут ключ SI , а остальные ключи разомкнуты. В этом случае входное напряжение $u_{вх}$ через замкнутый ключ SI и сопротивление R_1 заряжает емкость C_1 интегратора и выходное напряжение растет линейно во времени, как показано на рисунке 6. К концу интервала T_0 напряжение на выходе интегратора будет равно:

$$u_1(T_0) = k \int_0^{T_0} u_{ex} dt = kT_0 U_{ex}, \quad (1.3.3)$$

где $k^{-1} = R_1 C_1$ – постоянная времени интегратора, $U_{вх}$ – среднее значение входного напряжения:

$$U_{ex} = T_0^{-1} \int_0^{T_0} u_{ex} dt. \quad (1.3.4)$$

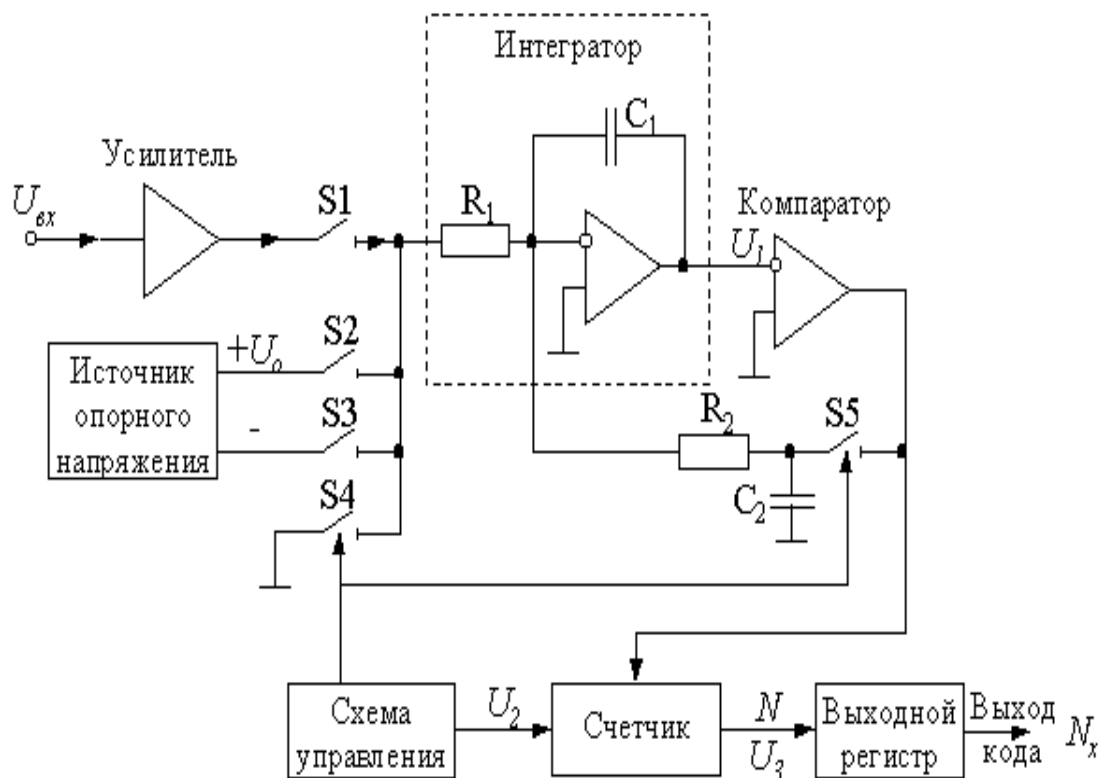


Рисунок 5 – Структурная схема интегрирующего АЦП с времяимпульсным преобразованием

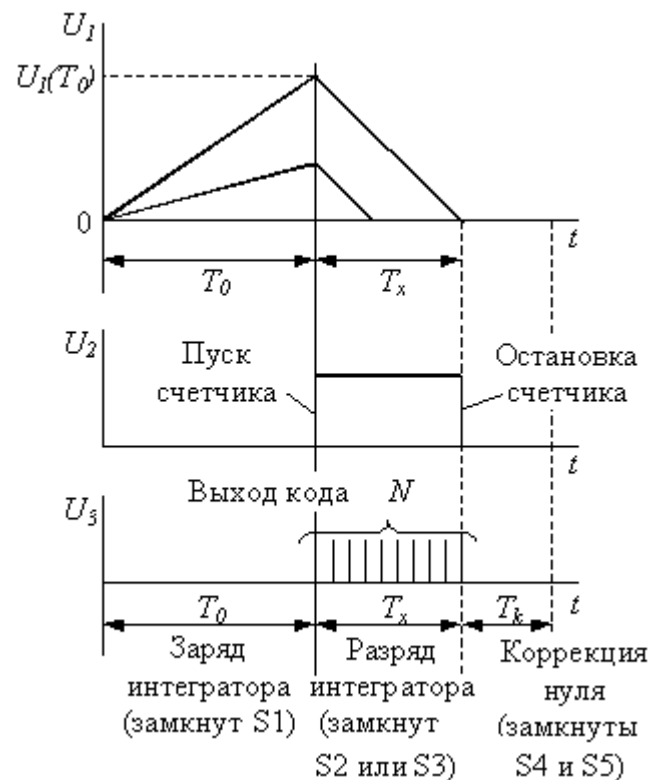


Рисунок 6 – Графики, иллюстрирующие работу АЦП

Во втором такте происходит разряд интегратора. При этом в зависимости от требуемой полярности источника опорного напряжения замыкается один из ключей S2 или S3. Разряд интегратора происходит с постоянной скоростью, которая не зависит от накопленного в интеграторе заряда, поэтому с увеличением накопленного заряда время разряда также увеличивается.

Конец разряда интегратора фиксируется компаратором К, после чего ключ S2 (или S3) размыкается.

Поскольку начало разряда определяет схема управления, а конец – компаратор, то длительность разряда интегратора можно определить по формуле:

$$U_{\text{вх}} = T_0^{-1} \int_0^{T_0} u_{\text{вх}} dt = 0, \quad (1.3.5)$$

откуда

$$kT_0 U_{\text{вх}} = kU_{\text{оп}} T_x \quad \text{или} \quad T_x = \frac{T_0}{U_{\text{оп}}} U_{\text{вх}},$$

что свидетельствует о пропорциональности интервала T_x среднему значению входного напряжения $U_{\text{вх}}$. Заполнение интервала T_x счетными импульсами, поступающими от схемы управления, позволяет найти числовой код $N_x = T_x f_0$. Для исследуемого АЦП $N_0 = f_0 T_0 = 1000$. Тогда:

$$N_x = 10^3 U_{\text{вх}} / U_{\text{оп}}. \quad (1.3.6)$$

К достоинствам интегрирующих АЦП следует отнести их высокую помехозащищенность. Если на входной сигнал наложена гармоническая помеха, то при равенстве периода помехи времени заряда интегратора $T_{\text{п}} = T_0$ среднее значение помехи к концу интервала интегрирования будет равно нулю, как показано штриховой линией на рисунке 6. Случайные

помехи и шумы также ослабляются интегрированием, хотя и в меньшей степени.

На третьем этапе производится коррекция нулевого уровня интегратора. Для этого замыкаются ключи S_4 и S_5 , а остальные ключи размыкаются. Так как вход интегратора через сопротивление R_1 соединен с общей шиной, то конденсатор C_2 через замкнутый ключ S_5 заряжается до напряжения ошибки, которое после размыкания ключей S_4 и S_5 вычитается из входного сигнала.

К недостаткам таких интегрирующих АЦП относится, прежде всего, сравнительно невысокое быстродействие. Кроме этого, при перегрузке АЦП большим входным сигналом происходит перезаряд интегрирующего конденсатора C_1 , поэтому после снятия перегрузки в течение нескольких циклов АЦП будет работать с большой погрешностью.

Полупроводниковые БИС интегрирующего АЦП КР572 ПВ2 (А, Б, В) предназначены для применения в измерительной аппаратуре различного назначения. Совместно с ИОН, несколькими резисторами и конденсаторами они выполняют функцию АЦП двойного интегрирования с автоматической коррекцией нуля и определением полярности входного сигнала.

Классификация микросхем КР572 ПВ2 по группам А, Б, В производится по погрешности преобразования.

Нумерация и назначение выводов микросхем КР572ПВ2: 1 – напряжение питания U_{cc1} ; 2 – цифровой выход d1; 3 – цифровой выход c1; 4 – цифровой выход b1; 5 – цифровой выход a1; 6 – цифровой выход f1; 7 – цифровой выход g1; 8 – цифровой выход l1; 9 – цифровой выход d10; 10 – цифровой выход c10; 11 – цифровой выход b10; 12 – цифровой выход a10; 13 – цифровой выход f10; 14 – цифровой выход l10; 15 – цифровой выход d100; 16 – цифровой выход b100; 17 – цифровой выход f100; 18 – цифровой выход l100; 19 – цифровой выход c1000; 20 – цифровой выход g1000; 21 – цифровая земля; 22 – цифровой выход g100; 23 – цифровой выход a100; 24 – цифровой выход c100; 25 – цифровой выход g10; 26 – напряжение питания U_{cc2} ; 27 – конденсатор интегратора $C_{инт}$; 28 – резистор инте-

гратора $R_{\text{инт}}$; 29 – конденсатор автокоррекции $C_{\text{ак}}$; 30 – аналоговый вход 1, низкопотенциальный; 31 – аналоговый вход 2, высокопотенциальный; 32 – общий аналоговый вход; 33 – опорный конденсатор $C_{\text{оп}}$; 34 – опорный конденсатор $C_{\text{оп}}$; 35 – опорное напряжение $U_{\text{оп}}$; 36 – опорное напряжение $U_{\text{оп}}$; 37 – контрольный вход; 38 – конденсатор генератора тактовых импульсов (ТИ) $C_{\text{ти}}$; 39 – резистор генератора ТИ $R_{\text{ти}}$; 40 – генератор ТИ.

Цифровая информация на выходе микросхем представляется в специальном коде, предназначенном для непосредственного управления 3,5-декадным цифровым табло с 7-ю сегментными полупроводниковыми индикаторами. Диапазон входного сигнала определяется значением внешнего опорного напряжения и соотношения $U_{\text{вх}} = \pm 1.999 U_{\text{оп}}$. Текущие показания цифрового табло соответствуют $1000(U_{\text{вх}}/U_{\text{оп}})$.

Максимальное число единиц счета преобразователя составляет ± 1999 . При превышении этого значения в эквиваленте входного преобразуемого сигнала три младшие цифры гаснут, а в старшем разряде остается единица.

1.4. Цифро-аналоговый преобразователь на примере ИС типа КР572

ПА1А

Цифро-аналоговым преобразователем (ЦАП) называется электронное устройство, предназначенное для преобразования цифровой информации в аналоговую. Они используются для формирования сигнала в виде напряжения или тока, функционально связанного с управляющим кодом. В большинстве случаев эта функциональная зависимость является линейной. Наиболее часто ЦАП используются для сопряжения устройств цифровой обработки сигналов с системами, работающими с аналоговыми сигналами. Кроме этого, ЦАП используются в качестве узлов обратной связи в аналого-цифровых преобразователях и в устройствах сравнения цифровых величин с аналоговыми.

Области применения ЦАП достаточно широки. Они применяются в системах передачи данных, в измерительных приборах и испытательных установках, в синтезаторах напряжения и генераторах сложных функций, для формирова-

ния изображений на экране дисплеев и др. В связи с этим разработано и выпускается большое количество интегральных микросхем ЦАП.

Схемы ЦАП можно классифицировать по различным признакам: принципу действия, виду выходного сигнала, полярности выходного сигнала, элементной базе и др. По принципу действия наибольшее распространение получили ЦАП следующих видов: со сложением токов, с делением напряжения и со сложением напряжений. В микроэлектронном исполнении применяются только первые два типа.

По виду выходного сигнала ЦАП делят на два вида: с токовым выходом и выходом по напряжению. Для преобразования выходного тока ЦАП в напряжение обычно используются операционные усилители (ОУ). По полярности выходного сигнала ЦАП принято делить на однополярные и двухполярные.

Управляющий код, подаваемый на вход ЦАП, может быть различным: двоичным, двоично-десятичным, Грея, унитарным и др. Кроме того, различными могут быть и уровни логических сигналов на входе ЦАП.

При формировании выходного напряжения ЦАП под действием управляющего кода обычно используются источники опорного напряжения. В зависимости от вида источника опорного напряжения ЦАП делят на две группы: с постоянным опорным напряжением и с изменяющимся опорным напряжением. Кроме этого, ЦАП делят по основным характеристикам: количеству разрядов, быстродействию, точности преобразования, потребляемой мощности.

Основные параметры ЦАП

Все параметры ЦАП можно разделить на две группы: статические и динамические. К статическим параметрам ЦАП относят: разрешающую способность, погрешность преобразования, диапазон значений выходного сигнала, характеристики управляющего кода, смещение нулевого уровня и некоторые другие.

К динамическим показателям ЦАП принято относить: время установления выходного сигнала, предельную частоту преобразования, динамическую погрешность. Рассмотрим некоторые из этих параметров.

Разрешающая способность ЦАП определяется как величина, обратная максимальному количеству градаций выходного сигнала. Так, например, если разрешающая способность ЦАП составляет 10^5 , то это означает, что максимальное число градаций выходного сигнала равно 10^5 . Иногда разрешающую способность ЦАП оценивают выходным напряжением при изменении выходного кода на единицу младшего разряда, т.е. шагом квантования:

$$\Delta_{\text{расч}} = \frac{U_{\text{оп}}}{2^n}, \quad (1.4.1)$$

где $U_{\text{оп}}$ – опорное напряжение, В; n – разрядность ЦАП.

Очевидно, что чем разрядность ЦАП, тем выше его разрешающая способность.

Погрешность преобразования ЦАП принято делить на дифференциальную и погрешность нелинейности. С ростом кода на входе ЦАП растет и выходное напряжение, однако при увеличении напряжения могут быть отклонения от линейной зависимости.

Дифференциальной погрешностью определяется отклонением разности двух аналоговых сигналов, соответствующих соседним кодам от номинального значения МЗР (рисунок 7). Дифференциальная погрешность идеального

преобразователя равна нулю. Допустимым отклонением считается $\pm\Delta_{\text{расч}}/2$ (1/2 МЗР).

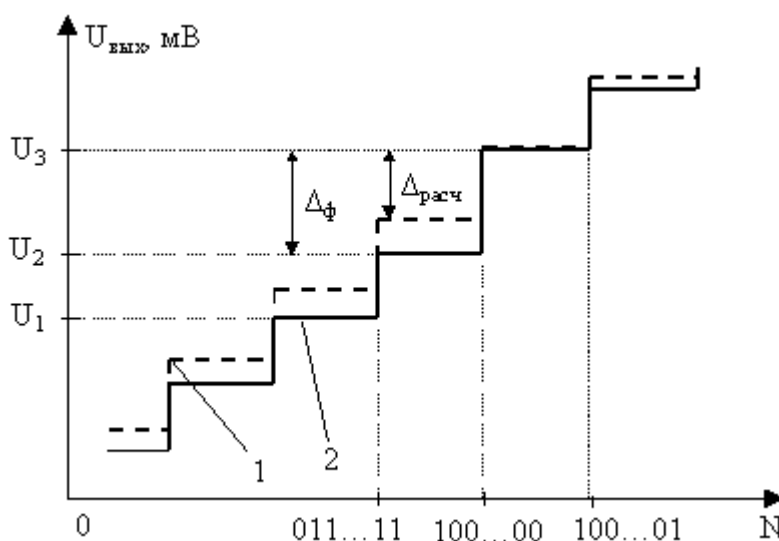


Рисунок 7 – Определение дифференциальной погрешности

1 – выходная характеристика идеального ЦАП,

2 – выходная характеристика реального ЦАП

Относительная дифференциальная погрешность определяется по формуле:

$$\delta_{\text{ндиф}} = \frac{(\Delta_\phi - \Delta_{\text{расч}})}{\Delta_{\text{расч}}} \cdot 100\%, \quad (1.4.2)$$

где Δ_ϕ – фактическое приращение выходного сигнала, вызванное изменением двоичного числа на 1.

Как правило, наибольшая динамическая погрешность наблюдается на так называемом главном переходе, когда увеличение двоичного числа на 1 приводит к смене предыдущего кода на код, соответствующий половине шкалы, при этом происходит выключение всех младших разрядов и включением старшего. Объясняется это тем, что при формировании выходного сигнала для кода 01111...1 происходит суммирование погрешностей всех разрядов.

Погрешностью нелинейности называют максимальное отклонение выходного напряжения от идеальной прямой во всем диапазоне преобразования (рисунок 8).

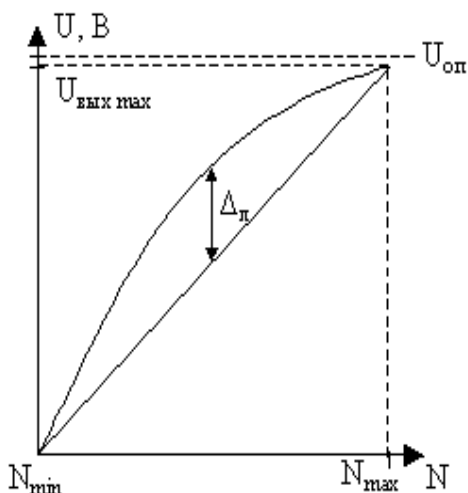


Рисунок 8 – Определение погрешности нелинейности

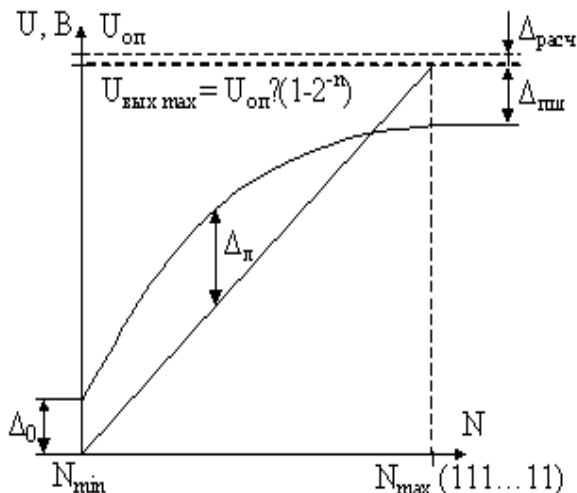


Рисунок 9 – Определение погрешностей полной шкалы, смещения нуля и нелинейности

Коэффициент нелинейности экспериментальной характеристики K_H :

$$K_H = \frac{\operatorname{tg} \alpha_{\max} - \operatorname{tg} \alpha_{\min}}{\operatorname{tg} \alpha_{\text{ср}}}, \quad (1.4.3)$$

где α_{\max} , α_{\min} , $\alpha_{\text{ср}}$ – максимальный, минимальный и средний углы наклона касательной к экспериментальной характеристике.

Напряжение смещения нуля определяется выходным напряжением при входном коде, соответствующем нулевому значению (рисунок 9).

$$\begin{aligned} \Delta_{\text{пш}} &= U_{\text{пшном}} - U_{\text{пшфакт}} = U_{\text{оп}} / 2^n (2^n - 1) - U_{\text{пшфакт}} = \\ &= (\Delta_{\text{расч}} - \Delta_{\text{ф}}^{\text{пш}}) (2^n - 1) \end{aligned} \quad (1.4.4)$$

$$\delta_{\text{пш}} = \frac{\Delta_{\text{пш}}}{U_{\text{пш}}} = \frac{(\Delta_{\text{расч}} - \Delta_{\text{ф}}^{\text{пш}})}{\Delta_{\text{расч}}}, \quad (1.4.5)$$

где $U_{\text{пшном}}$ – выходное напряжение полной шкалы идеального ЦАП;

$U_{\text{пшфакт}}$ – выходное напряжение полной шкалы реального ЦАП;

$\Delta_{\text{факт}}^{\text{пш}}$ – фактический шаг квантования.

Степень отклонения реальной характеристики ЦАП от расчетной определяется абсолютной погрешностью полной шкалы (рисунок 9).

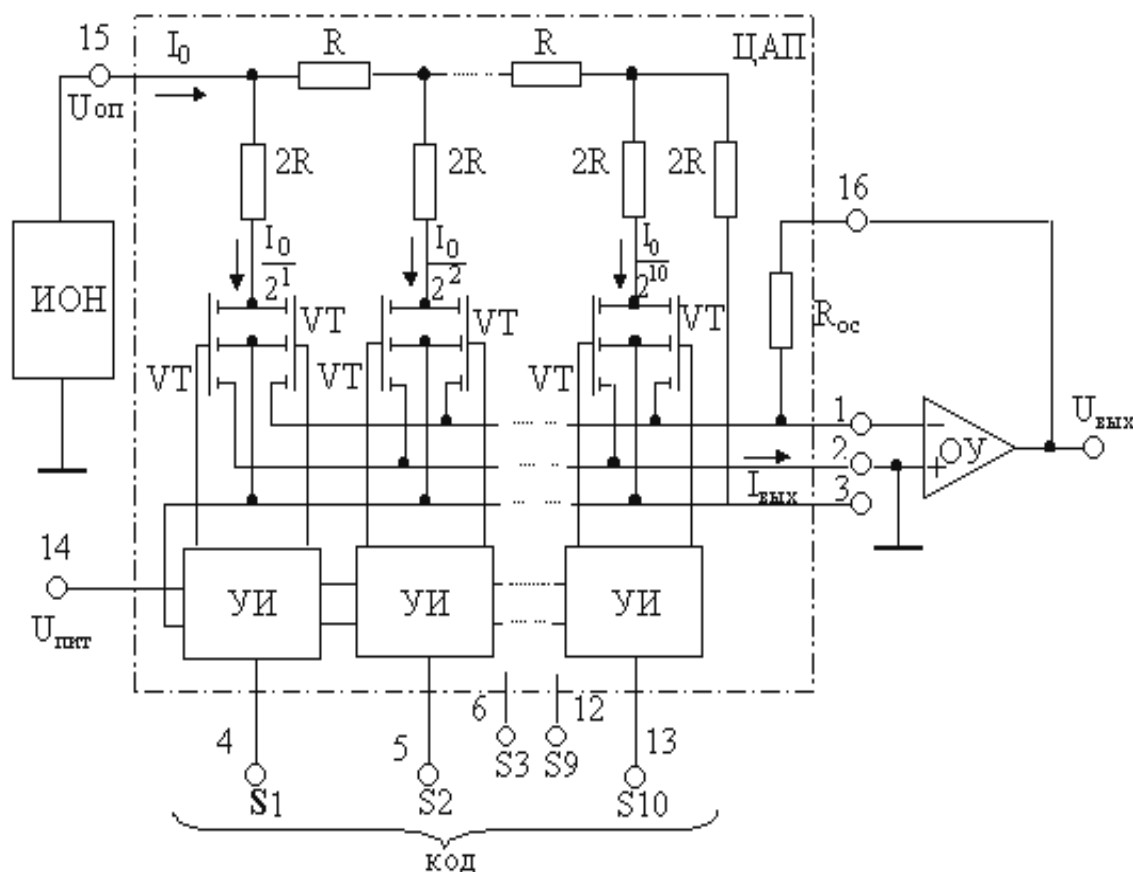


Рисунок 10 – Упрощенная функциональная электрическая схема ЦАП

K572PA1A

Относительная погрешность полной шкалы.

Время преобразования – это интервал времени от подачи входного кода до вхождения выходного сигнала в заданные пределы, определяемые погрешностью. Динамические свойства ИС характеризуются временем установления выходного напряжения (тока) на главном переходе. При этом наблюдается наиболее длительный и сложный переходный процесс со значительными выбросами.

Максимальная частота преобразования – наибольшая частота дискретизации, при которой все параметры ЦАП соответствуют заданным значениям.

Микросхема ЦАП КР572ПА1 предназначена для преобразования 10-разрядного прямого параллельного двоичного кода на цифровых входах в ток на аналоговом выходе, который пропорционален значениям кода и опорного напряжения. Она выполнена по КМОП технологии с поликремневыми затворами.

В состав ИС ЦАП КР572ПА1 входят прецизионные поликремневая резисторная (РМ) типа R – 2R, усилители – инверторы (УИ) для управления токовыми ключами, токовые двухпозиционные ключи, выполненные на КМОП транзисторах. Для работы в режиме с выходом по напряжению к ИС ЦАП КР572ПА1 подключаются внешние источник опорного напряжения (ИОН) и операционный усилитель (ОУ) с цепью отрицательной обратной связи (ЦОС), работающей в режиме суммирования токов.

Нумерация и назначение выводов микросхемы: 1 – аналоговый выход 1; 2 – аналоговый выход 2; 3 – общий вывод; 4 – цифровой вход 1 (СЗР); 5-12 – цифровые входы 2-9; 13 – цифровой вход 10 (МЗР); 14 – напряжение источника питания; 15 – опорное напряжение; 16 – вывод резистора обратной связи.

Метод преобразования, используемый в ИС К572ПА1, предполагает суммирование в соответствии с заданным значением двоичного кода всех разрядных токов взвешенных по двоичному закону и пропорциональных значению опорного напряжения на выводе 15. Двоичный закон распределения токов в ветвях РМ соблюдается при условии равенства потенциалов выходов 1 и 2. Это обеспечивается подключением выхода 1 к инвертирующему входу ОУ, охваченного отрицательной обратной связью. Неинвертирующий вход ОУ соединяется с выходом 2 и с шиной аналоговой земли. При этом осуществляется преобразование тока на выходе 1 в пропорциональное ему напряжение на выходе ОУ. Резистор $R_{ос}$ определяет

значение коэффициента преобразования и напряжения в конечной точке шкалы.

Связь между напряжением на выходе 1 схемы и двоичным кодом на цифровых входах ЦАП однозначна:

000...000	0
000...001	$-2 \cdot 10^{-10} U_{оп}$
...	...
100...000	$-2^{-1} U_{оп}$
...	...
111...111	$-(1-2^{-10}) U_{оп}$

2. КОМПЛЕКТ ЛАБОРАТОРНОГО ОБОРУДОВАНИЯ ПО ЭЛЕКТРОННОЙ ТЕХНИКЕ ТИПА К32

Комплект лабораторного оборудования К32 состоит из следующих составных частей (рисунок 11): блока управления комплектом (БУК);

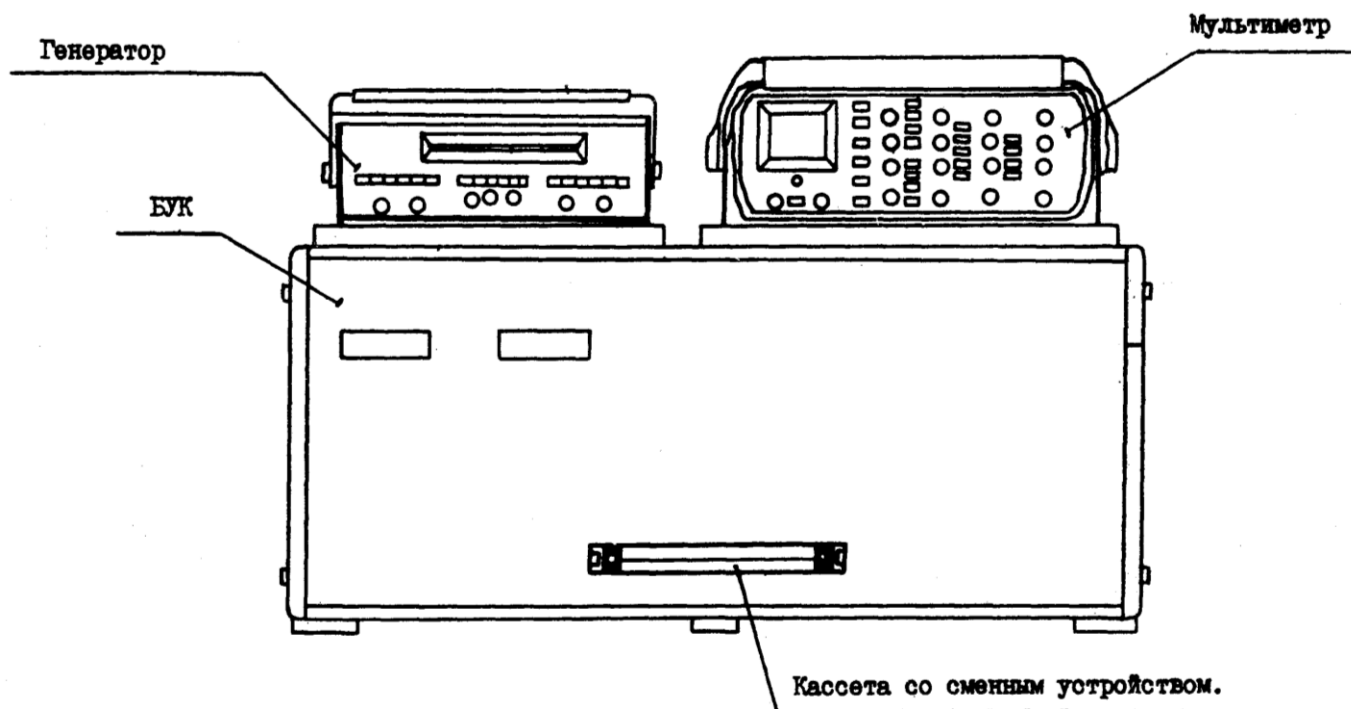


Рисунок 11 – Комплект лабораторного оборудования К32

блока мультиметра К32 (БМ К32) (далее мультиметр); устройства вспомогательное для осциллографов типа Л31 (далее генератор); кассеты и устройств сменных (УС).

БУК состоит из следующих частей (рисунок 12): передней панели (ПП); программатора серии импульсов (ПСИ); блока цифровой индикации (БЦИ); блока аналоговых сигналов (БАС); блока питания (БП).

Органы управления на ПП БУК объединены в группы согласно их функциональному назначению. Обозначение « $\overline{A|B}$ » у кнопок означает, что если кнопка не нажата, то выполняется функция А, а если нажата, то выполняется функция В.

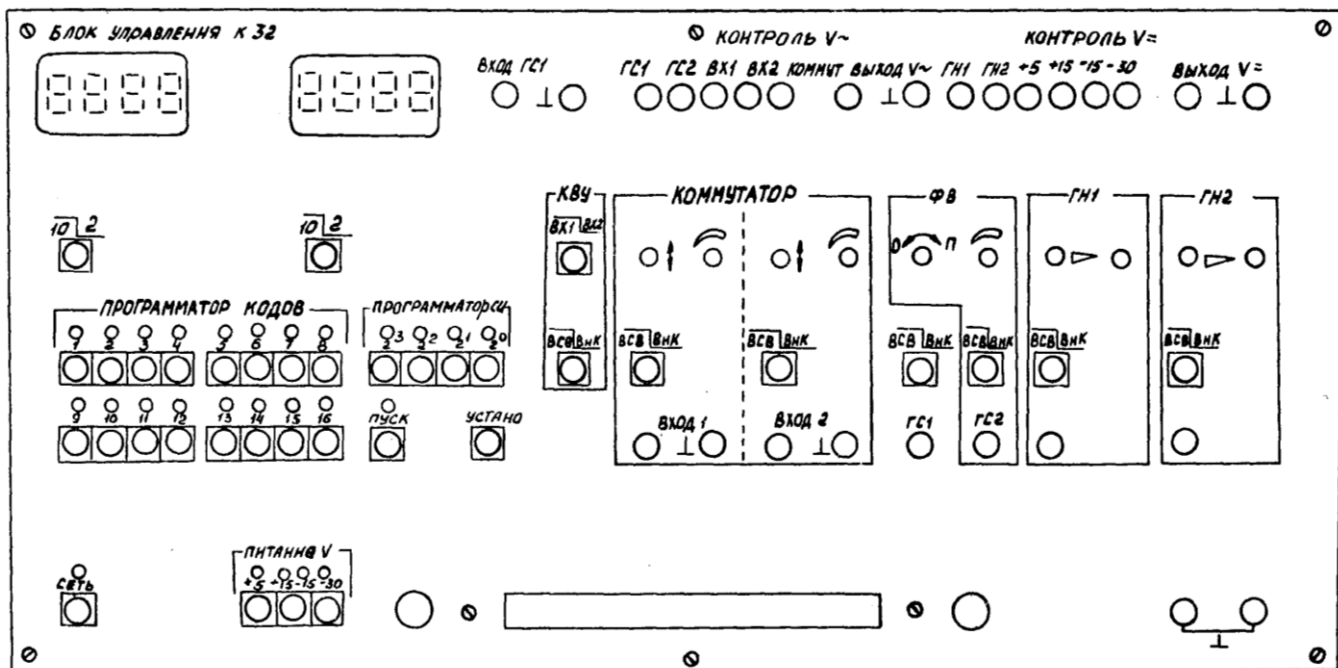


Рисунок 12 – Лицевая панель блока управления комплектом

Светодиоды над кнопками служат для индикации срабатывания кнопок – светодиоды светятся при нажатой кнопке (кроме кнопки «ПУСК») и сигнализации об исправности электрической цепи, которую коммутируют с помощью кнопочного переключателя. Кнопки под надписью «ПРОГРАММАТОР КОДОВ» служат для генерации комбинации сигналов «логический нуль» («0») – постоянного напряжения от 0 до 0.4 В – или «логическая единица» («1») – постоянного напряжения величиной от 2.4 до 5 В, генерация «1» происходит при нажатой кнопке, а «0» - при не-нажатой. Кнопки « $\overline{10} \underline{2}$ » служат для выбора режима работы БЦИ. Кнопки «2⁰», «2¹», «2²», «2³» под надписью «ПРОГРАММАТОР СИ» предназначены для набора двоичного кода количества импульсов в серии, генерируемой ГСИ, или номера канала прохождения импульсов (КПИ). Кнопка «ПУСК» служит для включения ГСИ. Кнопка «УСТАН 0» служит для установки ГСИ в исходное состояние. ПСИ предназначен для генерации пачек импульсов от одного до пятнадцати импульсов в серии с амплитудой от 2.4 до 5 В для коммутации цифровых сигналов, поступающих с плат сменных устройств на входы коммутатора и для генерации прямоугольных импульсов с амплитудой от 2.4 до 5 В.

ЛАБОРАТОРНАЯ РАБОТА №4

Исследование работы оперативного запоминающего устройства

Цель работы:

1. Исследовать основные режимы работы оперативного запоминающего устройства (ОЗУ) статического типа.
2. Получить навыки работы с программатором кодов (ПК), входящего в состав блока управления комплекта типа К32.

Программа работы

1. Исследование функциональных возможностей ОЗУ интегральной микросхемы типа К155РУ2.

Описание лабораторной установки

Исследуемая микросхема К155РУ2 (D6) расположена на печатной плате сменного устройства (УС) №14, входящего в комплект лабораторного оборудования К32. Управляющие сигналы подаются с помощью кнопок программатора кодов, расположенных на лицевой панели БУК. Причем адресация осуществляется кнопками «1» ÷ «4» (СЗР ÷ МЗР), а набор данных – кнопками «5» ÷ «8» (СЗР ÷ МЗР). Выбор микросхемы производится с помощью кнопки «13», а запись данных – «14». Генерация «логической единицы» («1») происходит при нажатой кнопке, а «логического нуля» («0») – при не нажатой. Выходные сигналы индуцируются на индикаторах левого цифрового табло в десятичном коде и на правом табло в двоичном коде.

Выполнение работы

1. Во входной разъем БУК вставить печатную плату сменного устройства №14.
2. Подключить блок управления к сети напряжения (~220В).
3. Для подачи питания на микросхему нажать кнопки «Сеть» и «+5В».

- 4 Исследование режимов записи и хранения.
 - 4.1 На входы ИС выбор микросхемы и разрешение записи подать высокий уровень.
 - 4.2 Набрать адрес строки и данные соответствующими кнопками.
 - 4.3 Для записи данных подать на входы ИС выбора микросхемы и разрешения записи напряжение низкого уровня.
 - 4.4 Нажатием кнопок «13» и «14» перевести микросхему в режим хранения.
 - 4.5 Заполнить все строки ЗУ в соответствии с таблицей 2.
- 5 Исследование режима считывания.
 - 5.1 В режиме хранения задать адрес необходимой строки.
 - 5.2 Отжатием кнопки «13» (выбор микросхемы) осуществить считывание записанной информации.
 - 5.3 Считать информацию по всем адресам и заполнить таблицу 3.
 - 5.4 Повторно считать данные по 5-6 адресам, сравнить результат с п.3 и сделать выводы о влиянии процесса считывания на хранящуюся в ячейках информацию.
- 6 По завершении работы отжать кнопки «Сеть» и «+5В», вынуть УС из разъема БУК.

Таблица 2

Адрес				Данные			
A1	A2	A3	A4	D1	D2	D3	D4
СЗР			МЗР	СЗР			МЗР
0	0	0	0	1	1	1	1
0	0	0	1	1	1	1	0
0	0	1	0	1	1	0	1
0	0	1	1	1	1	0	0
0	1	0	0	1	0	1	1
0	1	0	1	1	0	1	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	0	0
1	0	0	0	0	1	1	1
1	0	0	1	0	1	1	0
1	0	1	0	0	1	0	1
1	0	1	1	0	1	0	0
1	1	0	0	0	0	1	1
1	1	0	1	0	0	1	0
1	1	1	0	0	0	0	1
1	1	1	1	0	0	0	0

Таблица 3

Адрес				Данные			
A1	A2	A3	A4	D1	D2	D3	D4
СЗР			МЗР	СЗР			МЗР

Контрольные вопросы

1. Основные типы ЗУ.
2. Основные режимы работы ОЗУ.
3. Чем характеризуются статические и динамические параметры ЗУ?

ЛАБОРАТОРНАЯ РАБОТА №5

Исследование работы репрограммируемого постоянного запоминающего устройства

Цель работы:

1. Исследовать основные режимы репрограммируемого постоянного запоминающего устройства (РПЗУ).

Программа работы

1. Исследование функциональных возможностей РПЗУ (интегральной микросхемы (ИС) типа КР1601РР1)

Описание лабораторной установки

Исследуемая интегральная микросхема (ИС) КР1601РР1 (DD15) расположена на печатной плате сменного устройства (УС) № 19, входящего в комплект лабораторного оборудования К32. Управляющие сигналы подаются с помощью кнопок программатора кодов (ПК) и программатора серии импульсов (ПСИ). Адресация осуществляется кнопками «7» - «16» (МЗР + СЗР). Данные, предназначенные для записи, формируются на выходах асинхронного счетчика DD 4 (К155 ИЕ5), на тактовый вход которого поступают импульсы с ПСИ. Содержание счетчика отображается на левом цифровом табло. Обнуление счетчика производится нажатием кнопки «6». Считываемые данные по выбранному адресу после нажатия кнопки «4» индицируются на правом табло. Запись информации осуществ-

вляется кнопкой «3». Кнопки «1» и «2» предназначены соответственно для общего и построчного стирания. Выбор кристалла производится нажатием кнопки «5».

Выполнение работы

1. Во входной разъем БУК вставить сменное устройство №19.
2. Подключить блок управления к сети напряжения (~220).
3. Для подачи питания на микросхемы нажать кнопки «Сеть», «+5В», «±15В» и «30В».
4. Исследование режима программирования.
На вход ИС «выбор микросхемы» подать высокий уровень.
Нажатием кнопки «1» произвести общее стирание информации в ячейках ЗУ (при этом во все ячейки запишутся единицы). Обнулить счетчик.
С помощью ПСИ заполнить счетчик.
Набрать произвольный адрес кнопками ПК.
Осуществить запись нажатием кнопки «3».
Обнулить счетчик.
Произвести запись данных по 15-произвольным адресам, соблюдая последовательность в действиях согласно п.п. 4.3. – 4.6.
Исследование режима хранения информации.
Подать на вход «выбор микросхемы» низкий уровень.
Отключить питание. Необходимо соблюсти следующую последовательность отключения: «30В», «±15В», «+5В».
Подать питание на микросхему. При этом последовательность включения будет обратной последовательности п. 4.1.
5. Исследование режима считывания.
Задать адрес необходимой строки.
Нажать кнопку «4» и проверить содержимое ячеек.
Согласно п.п. 6.1. – 6.2. считать записанную ранее информацию. Сделать выводы.
6. Исследование режима построчного стирания.

Осуществить построчное стирание по любым 5 ранее задействованным адресам.

Убедиться, что стирание информации было произведено только по данным адресам. Сделать выводы.

7. Исследование режима общего стирания.

После нажатия кнопки «1» убедиться в том, что информация удалена.

8. Отключить питание согласно п. 5.2, отжать кнопку «Сеть». Вынуть УС из разъема БУК.

Контрольные вопросы

1. Основные типы ЗУ и их отличительные особенности.
2. Основные режимы работы исследуемого РПЗУ.
3. Почему соблюдается строгая последовательность включения/отключение питания ИС КР1601РР1.

ЛАБОРАТОРНАЯ РАБОТА №6

Исследование работы аналого-цифрового преобразователя

Цель работы

1. Исследовать работу аналого-цифрового преобразователя (АЦП) двухтактного интегрирования

Программа работы

1. Исследования функциональных возможностей АЦП (интегральной микросхемы (ИС) типа КР572ПВ2).

Описание лабораторной установки

Принципиальная электрическая схема лабораторной установки приведена на рисунке 13.

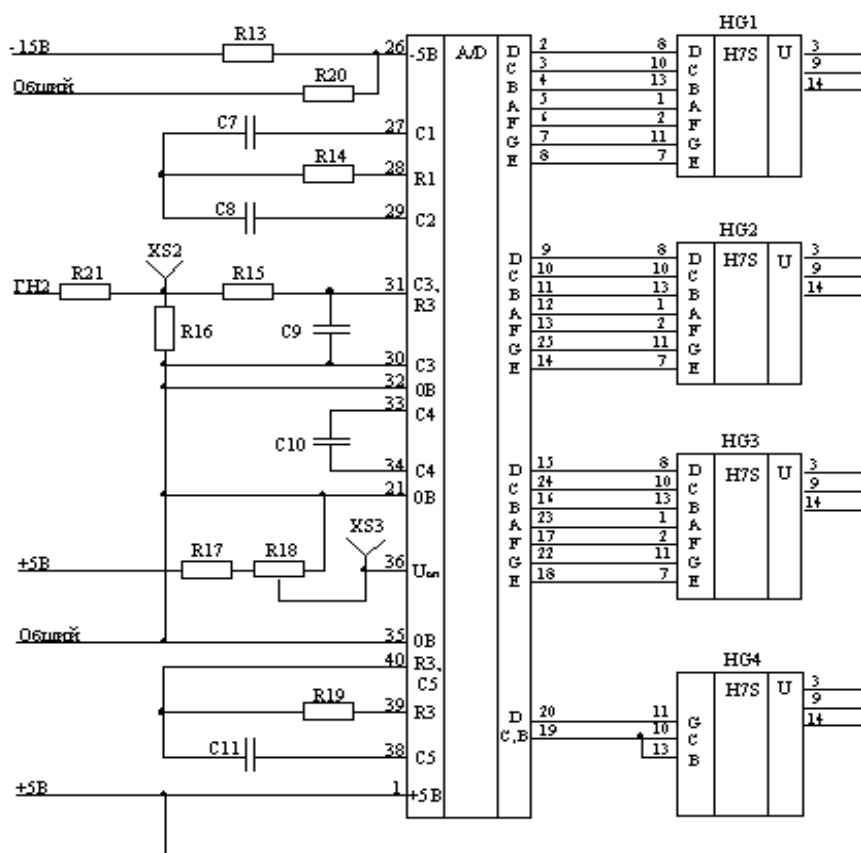


Рисунок 13 – Принципиальная электрическая схема исследования работы ИС АЦП КР572ПВ2

Исследуемая интегральная микросхема (ИС) КР572ПВ2 (DD6) расположена на печатной плате сменного устройства (УС) №18, входящего в комплект лабораторного оборудования К32. Аналоговый сигнал на входе АЦП регулируется с помощью генератора ГН2 (ручки расположены на лицевой панели блока управления БУК). Входное напряжение контролируется в гнезде XS2. Для контроля опорного напряжения используется гнездо XS3 (смотри рисунок 13). Выходной код отображается на индикаторах HG1 – HG4, расположенных на УС №18.

Выполнение работы

1. Во входной разъем БУК вставить печатную плату сменного устройства №18.

2. Подключить блок управления к сети напряжения (~ 220 В).

3. Для подачи питания на микросхемы нажать кнопки «Сеть», «+5В» и « ± 15 В».

4. Определение разрешающей способности АЦП при $U_{оп} = \text{const}$.

Измерить опорное напряжение в гнезде XS3.

С помощью генератора напряжения ГН2 установить входное напряжение равное $U_{оп}$ и снять показание индикаторов НГ1 – НГ4, расположенных на УС.

Повторить действия по п. 4.2., изменяя входное напряжение на величину $0.1U_{оп}$ (достаточно снять 5 значений).

Для полученных значений найти отношение приращений входного сигнала и выходного кода. Найти среднее значение разрешающей способности АЦП.

Воспользовавшись формулой (1.3.6), определить разрешающую способность АЦП и сравнить с экспериментальным значением. Сделать выводы.

5. Определение интегральной нелинейности АЦП.

Установить значение входного напряжения, выходной код при котором равен «-1999».

Зафиксировать значения выходного кода при увеличении входного сигнала с шагом $0.2U_{оп}$. Предельное значение входного напряжения должно соответствовать коду «1999».

Построить в одних осях зависимости $N_x = f(U_{вх})$ для идеального и реального АЦП (в относительных единицах).

Определить графически погрешность интегральной нелинейности АЦП. Сделать выводы.

На основании построенных графиков рассчитать значения

$$\text{tg } \alpha_{\min}, \text{tg } \alpha_{\text{ср}}, \text{tg } \alpha_{\max}.$$

Определить погрешность нелинейности по формуле (1.3.2) и сделать

выводы.

Определение погрешности полной шкалы.

Изменяя входное напряжение установить на цифровом табло код «1999». Замерить напряжение соответствующее этому коду.

Найти расчетное значение напряжения полной шкалы.

Определить абсолютную и относительную погрешности полной шкалы. Сделать выводы.

6. Отключить питание «+5В» и «±15В», отжать кнопку «Сеть». Вынуть УС из разъема БУК.

Контрольные вопросы

1. Назначение аналого-цифровых преобразователей.
2. Какие виды АЦП существуют.
3. Основные параметры АЦП.
4. Критерии выбора ИС АЦП.

ЛАБОРАТОРНАЯ РАБОТА №7

Исследование работы цифро-аналогового преобразователя

Цель работы

1. Исследовать работу умножающего цифро-аналогового преобразователя (ЦАП).
2. Получить навыки работы с программатором серии импульсов (ПСИ), входящего в состав блока управления комплекта (БУК) типа К32.

Программа работы

1. Исследование функциональных возможностей ЦАП (интегральной микросхемы (ИС) типа КР572ПА1А).

Описание лабораторной установки

Принципиальная электрическая схема лабораторной установки приведена на рисунке 14.

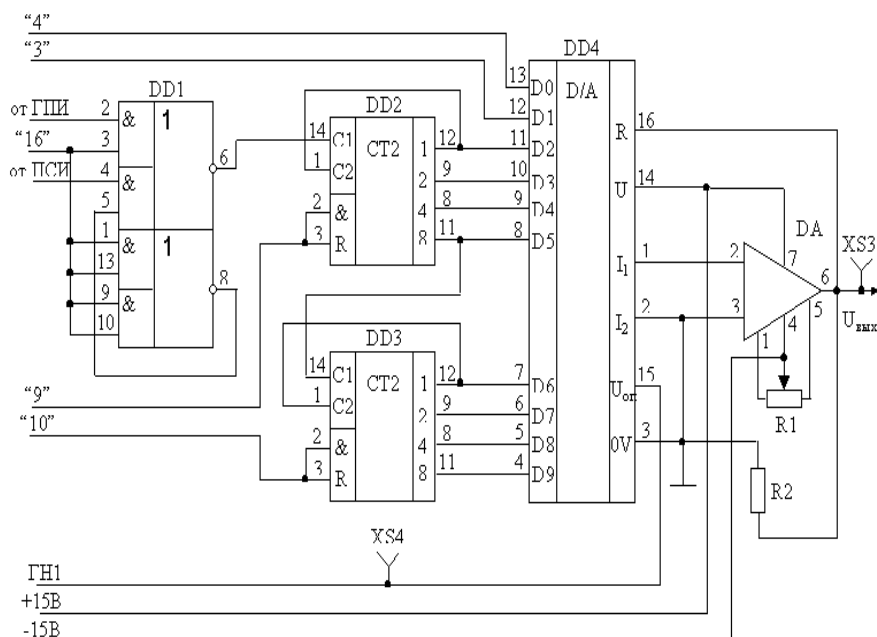


Рисунок 14 – Принципиальная электрическая схема исследования работы ИС ЦАП КР572ПА1А (DD4)

Исследуемая интегральная микросхема (ИС) КР572ПА1А (DD4) расположена на печатной плате сменного устройства (УС) №11, входящего в комплект лабораторного оборудования К32. Управляющие сигналы подаются с помощью кнопок «4», «3» программатора кодов (ПК), ПСИ и генератора прямоугольных импульсов (ГПИ) через буферные счётчики DD2 и DD3 (К155ИЕ5) (рисунок 14).

Импульсы, поступающие с ГПИ на тактовый вход асинхронного счётчика, приводят к изменению двоичного кода на цифровых входах ЦАП. В случае работы ПСИ количество тактовых импульсов определено набранным двоичным кодом посредством кнопок « 2^0 », « 2^1 », « 2^2 », « 2^3 », расположенных на передней панели БУК. Для генерации серии импульсов необходимо нажать кнопку «ПУСК» (над ней начинает светиться светодиод). При нажатии кнопки «УСТАН 0» светодиод перестаёт светиться и ПСИ готов к передаче новой пачки импульсов. Например, требуется

сгенерировать серию из 11-импульсов. Для этого необходимо осуществить следующую последовательность нажатия кнопок:

$$\langle \text{УСТАН } 0 \rangle \rightarrow \langle 2^3 \rangle + \langle 2^1 \rangle + \langle 2^0 \rangle = 11 \rightarrow \langle \text{ПУСК} \rangle.$$

Работу ПСИ можно прекратить кнопкой “16” (нажатое состояние), обеспечивая непосредственное прохождение импульсов с ГПИ на тактовый вход счётчика D6 и непрерывное изменение его содержимого, а значит информации на цифровых входах ЦАП (D2 – D9).

Сброс счётчика осуществляется либо автоматически при поступлении импульса на вход заполненного счётчика, либо нажатием кнопок “9” и “10” независимо от его состояния.

Выходной сигнал исследуется с помощью мультиметра в гнезде XS3. Величина опорного напряжения ($U_{он}$) контролируется в гнезде XS4. Левое цифровое табло индицирует содержимое счётчиков в десятичном коде. На третьем и четвертом индикаторах правого цифрового табло отображается номер состояния двух младших разрядов ЦАП (D0 и D1) также в десятичном коде (таблица 4).

Таблица 4 – Индикация состояний, соответствующих положениям кнопок “3” и “4”, на правом цифровом табло.

№ состояния	Кн. “3”	Кн. “4”
01	0	0
02	0	1
03	1	0
04	1	1

Выполнение работы

1. Во входной разъём БУК вставить печатную плату сменного устройства №11.
2. Подключить блок управления к сети напряжения ($\sim 220 \text{ В}$).
3. Для подачи питания на микросхемы нажать кнопки “Сеть”, “+5 В” и “±15 В”.

4. Определение разрешающей способности ЦАП при $U_{оп} = const$.

Кратковременным нажатием кнопок «9» и «10» обнулить содержимое счётчиков DD2 и DD3.

Вращением ручек генератора напряжения “ГН 1” установить величину опорного напряжения $U_{оп} = 10 В$.

Измерить с помощью вольтметра напряжение на выходе ОУ при соответствующем двоичном коде на входах ЦАП.

Повторить действия по пп. 5.4.1 ÷ 5.4.3 для $U_{оп} = 7; 5; 3 В$. Заполнить таблицу 5.

Определить среднее приращение выходного напряжения при изменении входного кода на единицу младшего разряда по экспериментальным данным для всех значений $U_{оп}$ (таблица 5).

Найти расчётное значение шага квантования и сравнить с экспериментальным.

Найти разрешающую способность ЦАП как величину, обратную максимальному количеству градаций выходного сигнала.

Таблица 5 – Результаты измерений выходного напряжения ЦАП при $U_{оп} = 10; 7; 5; 3 В$

Двоичный код на входе ЦАП				Напряжение на выходе ОУ, мВ			
D3	D2	D1	D0	$U_{оп} = 10В$	$U_{оп} = 7В$	$U_{оп} = 5В$	$U_{оп} = 3В$
0	0	0	1				
0	0	1	0				
0	0	1	1				
0	1	0	0				
0	1	0	1				
0	1	1	0				
0	1	1	1				

5. Определение дифференциальной нелинейности ЦАП.

С помощью ПСИ и кнопок “3”, “4” последовательно подать на вход

ЦАП два кода: 0111111111 и 1000000000, зафиксировав соответствующие значения выходного напряжения при $U_{оп} = 10; 7; 5; 3$ В

По формуле (1.4.2) рассчитать дифференциальную нелинейность и сравнить с паспортным значением. Сделать выводы.

6. Определение погрешности нелинейности при $U_{оп} = 10$ В.

Заполнить таблицу 6.

Таблица 6 – Результаты измерения $U_{вых}$ при $U_{оп} = 10$ В

Двоичный код на входе ЦАП	Выходное напряжение $U_{вых}$, В	Двоичный код на входе ЦАП	Выходное напряжение $U_{вых}$, В
0000000000		1000010000	
0000110000		1001000000	
0001100000		1001110000	
0010010000		1010100000	
0011000000		1011010000	
0011110000		1100000000	
0100100000		1100110000	
0101010000		1101100000	
0110000000		1110010000	
0110110000		1111000000	
0111100000		1111111111	

Построить графики зависимостей $U_{вых} = f(N)$ идеального ЦАП и, воспользовавшись данными таблицы 6, реального ЦАП (в относительных единицах).

Определить графически погрешность нелинейности. Сделать выводы.

На основании построенных графиков рассчитать значения

$$\text{tg } \alpha_{\min}, \text{tg } \alpha_{\text{ср}}, \text{tg } \alpha_{\max}.$$

Определить погрешность нелинейности по формуле (1.4.3) и сделать выводы.

7. Определить погрешность нуля.

8. Определение погрешности полной шкалы.

Из таблицы 6 взять значение выходного напряжения при входном коде 111111111.

Найти расчётное значение напряжения полной шкалы $U_{\text{пш}}$.

Определить абсолютную и относительную погрешности полной шкалы. Сделать выводы.

9. Определение тактовой частоты ГПИ при $U_{\text{оп}} = 10 \text{ В}$.

Нажать кнопку “16”.

С помощью осциллографа измерить интервал времени Δt , за который выходное напряжение изменяется от нуля до своего максимального значения.

Определить тактовую частоту ГПИ по формуле:

$$f_T = \frac{2^n}{\Delta t}, \text{ где } n = 8.$$

Контрольные вопросы

1. Назначение цифро-аналоговых преобразователей.
2. Какие виды цифро-аналоговых преобразователей существуют? Критерии классификации.
3. Основные параметры ЦАП.
4. Критерии выбора ИС ЦАП.

