

МИНИСТЕРСТВО ОБРАЗОВАНИЯ УКРАИНЫ
ДОНЕЦКИЙ
ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

**и задания к курсовой работе
по дисциплине
«Сети ЭВМ»**

МИНИСТЕРСТВО ОБРАЗОВАНИЯ УКРАИНЫ

ДОНЕЦКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Утверждено

на заседании кафедры

электронных вычислительных

машин

Протокол № 1 от 31.08.96г.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

и задания к курсовой работе по дисциплине

“Сети ЭВМ”

(для студентов специальности 7.09.15.01 дневной

и вечерней форм обучения)

Рекомендовано к изданию

методической комиссией

специальности 7.09.15.01

в количестве 250 экземпляров.

Протокол № 7 от 24.06.96г.

УДК 681.142

Методические указания и задания к курсовой работе по дисциплине
“Сети ЭВМ” (для студентов специальности 7.09.15.01. дневной и вечерней
форм обучения) / Сост. Ю. В. Губарь - Донецк : ДГТУ, 1996 г. - 60 с.

Приведены задания и общие требования к курсовой работе, описан
порядок ее выполнения, рассмотрено примерное содержание пояснительной
записки.

Составитель

доц. Губарь Ю. В.

Ответственный

за выпуск

проф. Святный В. А.

Рецензент

доц. Ладыженский Ю. В.

1. ЗАДАНИЯ К КУРСОВОЙ РАБОТЕ

Целью курсовой работы по дисциплине "Сети ЭВМ" является углубление и обобщение знаний в области проектирования средств вычислительных сетей, применение знаний, полученных студентами за время обучения, к системному решению задачи разработки и исследования локальной вычислительной сети (ЛВС). Курсовое проектирование направлено также на приобретение навыков выполнения научно-исследовательской работы и на ознакомление с научной и справочной литературой по специальности.

В задании на курсовое проектирование предусмотрено разработать аппаратную часть и программное обеспечение станции ЛВС, реализующей заданный протокол взаимодействия сетевых узлов. В качестве элементной базы предложено использовать микросхемы средней степени интеграции, такие как i8250, i8253, i8237, i16450, i16550, либо сетевые контроллеры и сопроцессоры ЛВС, выполненные на СБИС - i82586, i82588, i82590, WD2840 и другие.

Курсовая работа состоит из трех частей. В первой части разрабатывается оконный интерфейс для задания режимов работы и параметров станции сети. С помощью системы меню пользователь имеет возможность выбрать определенные сетевые параметры, которые в дальнейшем будут использованы при настройке конфигурации станции.

Во второй части разрабатываются подробные функциональные электрические схемы станции сети и отдельных ее узлов, таких как блок доступа к среде, кодер, декодер, буферы FIFO приема и передачи, блок контроля и другие. Дается описание работы каждого сетевого блока в различных режимах и приводятся временные диаграммы его функционирования.

В третьей части выполняется разработка алгоритмов и программ, реализующих заданный протокол работы сетевой станции. Эта работа осуществляется с использованием языков программирования Си или ассемблера.

Отлаженные программы по первой и третьей частям курсовой работы должны быть проверены экспериментально в лаборатории. Протоколы экспериментов подписываются руководителем курсовой работы.

Курсовая работа выполняется по индивидуальным заданиям. Для выбора исходных данных на проектирование студент должен получить у преподавателя номер N своего варианта.

1. Исходные данные для проектирования оконного интерфейса

1.1. Доступ к экрану монитора :

0 1 2 3 (N) mod 4

0 - через функции BIOS ;

1 - прямой доступ к видеопамяти ;

2 - с использованием фрейма меню ;

3 - с использованием Turbo Vision .

1.2. Цвет экрана меню :

0 1 2 3 4 5 6 7

ч с з г к ф р б (N) mod 8

где ч - черный ; с - синий ; з - зеленый ; г - голубой ;

к - красный ; ф - фиолетовый ; р - коричневый ; б - белый .

1.3. Координаты левого верхнего угла :

0 1 2 3 4

10, 20 5, 30 20, 15 17, 17 8, 12 (N) mod 5

1.4. Задание рамки под меню :

0 1

одинарная

двойная

(N) mod 2

1.5. Минимальное количество меню :

0 1 2

2 3 4

(N) mod 3

В качестве пунктов меню по согласованию с руководителем могут быть выбраны следующие параметры : адреса взаимодействующих станций , номера линий сетевых прерываний , размер буфера FIFO , межкадровый интервал , число попыток передать кадр после коллизии или другие параметры .

2. Исходные данные для проектирования сетевой станции

В зависимости от реализуемого метода доступа и топологии сети составлены три варианта заданий . Общими исходными данными для всех трех вариантов являются следующие .

2.1. Среда передачи кадров :

0 1

коаксиальный кабель

витая пара

(N) mod 2

2.2. Размер буфера FIFO для передачи и приема кадров :

0 1 2 3

64 128 512 1024

(N) mod 4

2.3. Межкадровый интервал (в микросекундах) :

0 1 2 3 4

6 8 10 12 14

(N) mod 5

2.4. Базовый адрес станции :

0	1	2	3	
0320h	0430h	0540h	0360h	(N) mod 4

2.5. Линия прерывания от сетевого адаптера :

0	1	2	3	4	
3	5	7	10	12	(N) mod 5

2.6. Элементная база : K155 , K1810 , i8250 , i8253 , i8237 ,
i82586 , i82588 , i82590 , i82592.

Далее приводятся исходные данные для каждого из трех вариантов .

Вариант 1 .

2.7. Топология сети : шина .

2.8. Метод доступа : CSMA / CD .

2.9. Реализуемые сетевые уровни : физический, канальный, прикладной.

2.10. Число попыток передать кадр после коллизии :

0	1	2	3	4	5	6	7	
1	2	3	4	5	6	7	8	(N) mod 8

2.11. Функции физического уровня : прием и передача кадров , манчестерское кодирование и декодирование , контроль несущей , обнаружение коллизий , контроль ошибок с использованием циклических кодов .

2.12. Функции канального уровня : сборка и разборка кадров , реализация протокола канального уровня , связь с физическим и прикладным уровнями , выявление и выдача сообщений об ошибках .

2.13. Формат кадра канального уровня :

F	DA	SA	CMD	L	D	CRC	F
---	----	----	-----	---	---	-----	---

где F - флаг (0 111 111 0) ;

DA - адрес станции назначения (в байтах) :

0	1	2	3	4	5	
1	2	3	4	5	6	(N) mod 6

SA - адрес станции отправителя кадра (размер поля SA равен DA) ;

CMD - поле , определяющее тип кадра :

а) информационный кадр :

0	N (S)	P / F	N (R)
---	---------	-------	---------

I - кадр

где 0 - определяет тип I - кадра ;

N (S) - номер последовательности передачи (порядковый номер данного I - кадра) . Размер этого поля в битах определяет

модуль нумерации информационных кадров :

$$\begin{matrix} 0 & 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 & 6 \end{matrix} \quad (N) \bmod 5$$

P / F - бит опроса / окончания ;

N (R) - номер последовательности приема (подтверждения)
принятых кадров до номера N (R) - 1 включительно .
Размер этого поля совпадает с размером поля N (S) .

б) супервизорный кадр :

1 0	S S	P / F	N (R)	S - кадр
-----	-----	-------	---------	----------

где 1 0 - определяют тип супервизорного кадра ;
Поле S идентифицирует один из четырех супервизорных кадров в соответствии с табл. 1 .

Таблица 1

Поле S	Тип кадра	Назначение кадра
0 0	RR	Готов к приему
0 1	RNR	Неготов к приему
1 0	REJ	Отказ
1 1	SREJ	Селективный отказ

в) служебный кадр :

1 1	L C	P / F	M
-----	-----	-------	---

где 1 1 - определяют тип служебного кадра ;

L C - указывает номер логического канала связи :

$$\begin{matrix} 0 & 1 & 2 & 3 \\ 0 & 1 & 2 & 3 \end{matrix} \quad (N) \bmod 4$$

M - определяет тип служебного кадра в соответствии с табл. 2 :

Таблица 2

Поле M	Тип кадра	Назначение кадра
0 0 0	SLC	Открыть логический канал
0 0 1	UA	Ненумерованное подтверждение
0 1 0	DM	Режим разъединения
0 1 1	DISC	Разъединить логический канал
1 0 0	FRMR	Отказ от кадра

L - поле длины информационной области (D) кадра . Размер этого поля связан с размером поля D кадра . Если , например , D

составляет 512 байт , то поле L имеет девять разрядов .

D - информационная область кадра . Ее размер определяется в байтах :

0	1	2	3	4	5	6	
16	32	64	128	256	512	1024	(N) mod 7

CRC - поле циклической проверочной последовательности длиной

в два байта . Порождающий полином имеет вид :

- | | | | | |
|----------------------------------|----|----|----|-------------|
| 1) $X^{16} + X^{12} + X^5 + 1$; | 0 | 1 | 2 | 3 |
| 2) $X^{14} + X^8 + X^3 + 1$; | 1) | 2) | 3) | 4) |
| 3) $X^{12} + X^7 + X^2 + 1$; | | | | |
| 4) $X^{11} + X^6 + X^5 + 1$. | | | | (N) mod 4 |

2.14. Формат кадра прикладного уровня .

а) управляющий кадр :

0	CMD	DA
---	-----	----

Y - кадр ,

где 0 - бит , определяющий тип управляющего кадра ;

CMD - код выполняемой команды (один байт) :

CMD = FFh - запрос каналному уровню установить логический канал связи со станцией DA .

Ответ от канального уровня означает следующее :

CMD = FFh - логический канал связи открыт ;

CMD = 00h - логический канал установить нельзя .

DA - поле адреса назначения кадра (размер этого поля совпадает с размером поля DA канального уровня) .

б) информационный кадр :

1	DA	C	L	D
---	----	---	---	---

D - кадр ,

где 1 - бит , определяющий тип информационного кадра ;

DA - адрес станции назначения ;

C - сокет (номер решаемой прикладной задачи) :

0	1	
1	2	(N) mod 2

L - длина информационной области кадра D (в байтах) ;

D - информационная область кадра в байтах .

Размеры полей L и D совпадают с размерами аналогичных полей канального уровня .

2.15. Задача , решаемая на прикладном уровне : (N) mod 3 :

$$1) y = \prod_{i=1}^N a_i + b ,$$

где $a_1 = b = (N)_{\text{mod } 9}$, $a_{i+1} = a_i + 1$, $i = 2, 3, \dots, N$.

Значения a_i и b_i поступают из прикладного уровня станции с адресом DA. На прикладном уровне станции с адресом SA производится вычисление значения y , после чего полученный результат передается по сети прикладному уровню станции с адресом DA.

2). Из станции с адресом SA на станцию с адресом DA передается информационный файл длиной два килобайта. Содержимое файла представляет собой пронумерованные и заключенные в скобки последовательности фамилии, имени и отчества студента, набранные через 10 пробелов в виде:

(Ф И О)₀ (Ф И О)₁ . . . (Ф И О)₁₀₀₀

$$3). Y = \sum_{i=1}^N (a_i + b_i)^2,$$

где $a_1 = b_1 = (N) \bmod 7$,
 $a_{i+1} = a_i + 1$, $b_{i+1} = b_i - 1$.

Вариант 2.

2.16. Топология сети : шина .

2.17. Метод доступа : передача полномочий по логическому кольцу с помощью маркера .

2.18. Реализуемые сетевые уровни : физический , канальный , прикладной .

2.19. Задачи физического уровня : прием и передача кадров и маркера в сетевую среду , кодирование и декодирование кадров , буферизация кадров , контроль принятых кадров с использованием циклических кодов , связь с канальным уровнем .

2.20. Задачи канального уровня : инициализация логического кольца , подключение и отключение станций от логического кольца , анализ сетевых ошибок , обработка кадров и маркера , связь с прикладным и физическим уровнями .

2.21. Формат кадра канального уровня :

PR	SD	FC	DA	SA	L	D	CRC	ED
----	----	----	----	----	---	---	-----	----

где PR - преамбула (от одного до нескольких байт вида 10 10 10 10) :

$$\begin{array}{ccccc} 0 & 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 & 5 \end{array} \quad (N) \bmod 5$$

SD - начальный ограничитель кадра : JK 0 JK 0 0 0 (J и K - "нарушители" битового интервала манчестерского кода).

FC - поле управления (один байт). Назначение разрядов этого поля определено в табл. 3 .

Таблица 3

Поле FC	Тип кадра	Назначение кадра
0000 0000	ТМ	Требование маркера
0000 0001	ПСС1	Поиск следующей станции 1
0000 0010	ПСС2	Поиск следующей станции 2
0000 0011	КС	Кто следующий ?
0000 0100	РС	Разрешение спора
0000 1000	М	Маркер
0000 1100	УСС	Установка следующей станции

В поле данных кадров КС и УСС необходимо поместить адрес следующей станции.

DA - адрес станции назначения (в байтах) :

0	1	2	3	4		
2	3	4	5	6		(N) mod 5

SA - адрес станции отправителя кадра (размер этого поля совпадает с размером поля DA).

L - поле длины информационной области кадра (размер этого поля зависит от размера поля D).

D - информационная область кадра (в байтах):

0	1	2	3	4	5	6	
16	32	64	128	256	512	1024	(N) mod 7

CRC - поле циклической проверочной последовательности кадра (два байта). Вид порождающего полинома смотри п.2.13.

ED - конечный разделитель кадра : JK I JK I E ,

где I - промежуточный бит : I = 0 - последний кадр ;

I = 1 - продолжение передачи следует ;

E - бит контроля ошибок :

E = 0 - ошибок нет ; E = 1 - ошибки в кадре .

2.21. Формат маркера :

PR	SD	0000 1000	DA	SA	CRC	ED
----	----	-----------	----	----	-----	----

2.22. Формат кадра прикладного уровня и задача , решаемая на этом уровне , определены в пунктах 2.14 и 2.15 задания .

Вариант 3 .

2.23. Топология сети : кольцо .

2.24. Метод доступа : передача полномочий по кольцу с использованием маркера .

2.25. Реализуемые сетевые уровни : физический , канальный , прикладной.

2.26. Задачи физического уровня : прием и передача кадров и маркера , кодирование и декодирование кадров , буферизация кадров , контроль принятых кадров , связь с канальным уровнем .

2.27. Задачи канального уровня : выбор активного монитора , очистка кольца , обработка кадров и маркера , подключение к кольцу новой станции , определение соседних станций , сигнализация о неисправностях .

2.28. Формат кадра канального уровня :

SD	AC	FC	DA	SA	L	D	CRC	ED	FS
----	----	----	----	----	---	---	-----	----	----

где SD - начальный ограничитель кадра : J K 0 J K 0 0 0 .

AC - поле управления доступом : P P P T M R R R :

P P P - биты приоритета кадра :

T - бит идентификации маркера :

T = 0 - маркер ; T = 1 - кадр .

M - бит идентификации активного монитора :

M = 1 - активный монитор ;

M = 0 - любая другая станция .

R R R - биты резервирования приоритета .

FC - поле управления (один байт). Назначение разрядов этого поля определено в табл. 4 .

DA - адрес станции назначения (в байтах) :

0	1	2	3	4	
2	3	4	5	6	(N) mod 5

SA - адрес станции отправителя кадра (размер этого поля совпадает с размером поля DA) .

L - поле длины информационной области кадра (зависит от размера области D кадра) .

D - информационная область кадра в байтах :

0	1	2	3	4	5	
16	32	64	128	256	523	(N) mod 6

Таблица 4

Поле FC	Тип кадра	Назначение кадра
0100 0011	ТМ	Требование маркера
0100 0000	ТДА	Тест дублирования адреса
0100 0101	АМС	Активный монитор существует
0100 0110	СРМ	Существует резервный монитор
0100 0010	С	Сигнал
0100 0100	О	Очистка

CRC - поле циклической проверочной последовательности кадра (два байта). Вид порождающего полинома приведен в пункте 2.13.

ED - конечный ограничитель кадра (смотри пункт 2.21).

FS - поле статуса кадра : A C 0 0 A C 0 0 ,

где A - бит распознавания адреса ; C - бит копирования .

2.29. Формат маркера :

SD	AC	ED
----	----	----

2.30. Формат кадра прикладного уровня и задача , решаемая на прикладном уровне , определены в пунктах 2.14 и 2.15 соответственно .

Техническое задание на курсовое проектирование (приложение 1) оформляется на первом занятии , подписывается студентом и руководителем работы , а также утверждается заведующим кафедрой ЭВМ .

II. ОБЩИЕ ТРЕБОВАНИЯ К КУРСОВОЙ РАБОТЕ

Курсовая работа содержит пояснительную записку и листинги разработанных программ . Пояснительная записка должна содержать 40 - 50 страниц рукописного текста и включать следующие обязательные элементы :

- титульный лист ;
- техническое задание ;
- реферат (со штампом в 40 мм) ;
- аннотацию (на украинском и английском языках) ;
- содержание (со штампом в 15 мм) ;
- основную часть ;
- листинги разработанных программ .

Примерное оглавление пояснительной записки :

В В Е Д Е Н И Е

1. РАЗРАБОТКА ОКОННОГО ИНТЕРФЕЙСА ДЛЯ ЗАДАНИЯ

РЕЖИМОВ РАБОТЫ СТАНЦИИ ЛОКАЛЬНОЙ СЕТИ

- 1.1. Общие сведения о программе
- 1.2. Логическая структура программы
- 1.3. Разработка основных программных модулей
- 1.4. Анализ полученных результатов

2. РАЗРАБОТКА ФУНКЦИОНАЛЬНЫХ ЭЛЕКТРИЧЕСКИХ СХЕМ СТАНЦИИ И ЕЕ УЗЛОВ

- 2.1. Функциональная схема станции и режимы ее работы
- 2.2. Функциональные узлы станции
 - 2.2.1. Передающий блок
 - 2.2.2. Приемный блок
 - 2.2.3. Кодер и декодер
 - 2.2.4. Анализаторы активности среды и коллизий
 - 2.2.5. Блок контроля и синхронизации

2.3. Расчет временных параметров рабочей станции

3. РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СТАНЦИИ

- 3.1. Структура программного обеспечения
- 3.2. Разработка программных модулей сетевой станции
 - 3.2.1. Модуль инициализации
 - 3.2.2. Модуль планирования
 - 3.2.3. Модуль обработки прерываний от сетевого порта
 - 3.2.4. Вспомогательные программные модули

4. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

- 4.1. Программные модули для работы с СОМ - портами
- 4.2. Анализ полученных результатов

ЗАКЛЮЧЕНИЕ

Список использованной литературы

Приложение 1. Листинг программы оконного интерфейса

Приложение 2. Протокол исследований оконного интерфейса

Приложение 3. Листинг программного обеспечения сетевой станции

Приложение 4. Протокол исследований работы сетевых станций

Пояснительная записка должна быть оформлена в соответствии с действующими стандартами ЕСКД и ЕСПД.

III. ПОРЯДОК ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ

Процесс выполнения курсовой работы разбивается на ряд последовательных этапов в соответствии со следующим графиком :

1 неделя : составление и анализ технического задания .

2 - 4 недели : разработка оконного интерфейса сетевой станции .

- 5 - 6 недели : разработка функциональных схем узлов станции .
- 7 - 8 недели : разработка алгоритмов работы станции .
- 9 - 10 недели : разработка программного обеспечения станции .
- 11 - 12 недели : экспериментальные исследования .
- 13 неделя : оформление пояснительной записки .
- 14 неделя : защита курсовой работы .

IV. ЗАЩИТА КУРСОВОЙ РАБОТЫ

Выполненная и проверенная руководителем курсовая работа защищается перед комиссией кафедры . После краткого доклада о проделанной работе студент отвечает на вопросы членов комиссии . В случае обнаружения принципиальных ошибок работа возвращается на доработку и назначается повторная защита , либо выставляется неудовлетворительная оценка и выдается новое задание на проектирование .

V. СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

Литература для проектирования оконного интерфейса :

1. Язык СИ для профессионалов . По материалам книги Г. Шилдта . - М. : И. В. К. - СОФТ , 1992г. - 320 с.
2. Кардышев С. В. , Капкин А. М. Интерфейс программиста Турбо - Си . - М. : Радио и связь , 1992 г. - 192 с.
3. Turbo Vision для C++ . - М. : СП "Интерквадро" , 1991г. - 165 с.
4. Страуструп Б. Язык программирования C++ . : В 2-х частях . - К. : "Диа Софт" , 1993 г. - 168 с.

Литература для проектирования сетевой станции :

5. Овчинников В. В. , Рыбкин И. И. Техническая база интерфейсов локальных вычислительных сетей . - М. : Радио и связь , 1989 г.
6. Овчинников В. В. Архитектура распределенных вычислительных микропроцессорных систем . - М. : Энергоатомиздат , 1988 г.
7. Бойченко Е. В. Методы схемотехнического проектирования распределенных информационно- вычислительных микропроцессорных систем . - М. : Энергоатомиздат , 1988 г.
8. Бойченко Е. В. , Кальфа В. , Овчинников В. В. Локальные вычислительные сети . - М. : Радио и связь , 1985 г.
9. Щербо В. К. и др . Стандарты по локальным вычислительным сетям . - М. : Радио и связь , 1990 г.
10. Дэвис Д. и др . Вычислительные сети и сетевые протоколы . - М. : Мир , 1982 г.
11. Аничкин С. А. и др . Протоколы информационно- вычислительных

- сетей : Справочник . - М . : Радио и связь , 1990 г .
12. Райс Л . Эксперименты с локальными сетями микроЭВМ . - М . : Мир , 1990 г .
 13. Блэк Ю . Сети ЭВМ : протоколы , стандарты , интерфейсы . - М . : Мир , 1990 г .
 14. Шварц М . Сети связи : протоколы , моделирование и анализ : В 2 - х томах . - М . : Наука , 1992 г .
 15. Микропроцессорный комплект К1810 : Структура , программирование , применение : Справочная книга / Под ред . Ю . М . Казаринова . - М . : Высш . шк . , 1990 г .
 16. Самофалов К . Г . , Викторов О . В . Микропроцессоры . Библиотека инженера . - К . : Техника , 1989 г .
 17. Алексенко А . Г . , Галицын А . А . , Иванников А . Д . Проектирование радиоэлектронной аппаратуры на микропроцессорах . - М . : Радио и связь , 1984 г .
 18. Шило В . Л . Популярныe цифровые микросхемы . - М . : Радио и связь , 1988 г .
 19. Тули М . Справочное пособие по цифровой электронике . - М . : Энергоатомиздат , 1990 г .
 20. Фролов А . В . , Фролов Г . В . Программирование модемов . - М . : Диалог - МИФИ , 1993 г .
 21. Касаткин А . В . Профессиональное программирование на языке Си . Системное программирование . - Минск , Выс . шк . , 1993 г .
 22. Джордейн Р . Справочник программиста компьютеров типа IBM PC , XT и AT . - М . : Финансы и статистика , 1992 г .
 23. Personal Computer Hardware Library Serial / Parallel Adapter . - IBM Co . , USA , 1986 .
 24. Local Area Networking (LAN) Component Users Manual . - Intel Corp . : Santa Clara , CA , USA , 1988 .
 25. RS - 232 Card Eusers Manual . - IBM Corp . , USA , 1987 .

VI. ПРИМЕРНОЕ СОДЕРЖАНИЕ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ

Введение

Во введении следует очертить круг задач , решаемых с использованием локальных вычислительных сетей (ЛВС) . По материалам литературных источников целесообразно рассмотреть типы современных ЛВС , их характеристики и области применения . Могут быть также кратко проанализированы известные методы доступа в ЛВС , протоколы и стандарты , реализованные в современных сетях . Здесь же должны быть определены цель

курсового проектирования , возможные области применения проектируемой станции сети , контроль ее работоспособности .

1. ПРОЕКТИРОВАНИЕ ОКОННОГО ИНТЕРФЕЙСА ДЛЯ ЗАДАНИЯ РЕЖИМОВ РАБОТЫ СТАНЦИИ ЛОКАЛЬНОЙ СЕТИ

1.1. Общие требования к программе

Программа `win_kurs.c` представляет собой оконный интерфейс на основе доступа через вызовы функций BIOS. Программа предназначена для задания трех рабочих параметров сетевой станции : режима работы станции (прием или передача) , номера станции , с которой будет установлен логический канал связи , и номер сокета , на котором решается прикладная задача . Программа написана на языке Borland C++.

Для работы программы требуется примерно 50 К байт основной памяти компьютера , а также цветной или монохромный адаптер дисплея , работающего в текстовом режиме 80 x 25 символов .

1.2. Структура программы

Вызовы BIOS используют программные прерывания . Для доступа к экрану используется прерывание 16 (10 H) . Оно имеет несколько режимов , которые определяются по значению регистра AH . Если вызываемая функция возвращает значение , оно заносится в регистр AL .

Для доступа к прерываниям BIOS в компиляторе языка C++ используется функция `int 86 ()` . Эта функция имеет форму [1 , 2] :

```
int 86 ( num , inregs , outregs ) ;  
    int num ; /* номер прерывания */  
    union REGS * inregs ; /* входные значения регистров */  
    union REGS * outregs ; /* выходные значения регистров */
```

REGS - это объединение двух структур : WORDREGS и BYTEREGS , которые описаны в заголовке файла `dos.h` .

Для задания режима работы дисплея используется нулевая функция 16-го прерывания BIOS . Так , например , для задания текстового цветного режима 80 x 25 знакомест в регистр AL следует записать тройку . Тогда программный код примет вид :

```
union REGS in , out ;  
in . h . ah = 0 ;  
in . h . al = 3 ;  
int 86 ( 16 , & in , & out ) ;
```

После запуска программы `win_kurs.exe` на экране дисплея появляется окно , выделенное красным цветом в одинарной рамке , вид которого

показан на рис. 1.1. Заставка с фамилией автора курсовой работы находится в центре экрана.

При нажатии на клавишу Enter на экране появляется следующее окно (рис. 1.2). В поле окна отображаются не только пункты меню, но и органы управления отдельными пунктами меню.

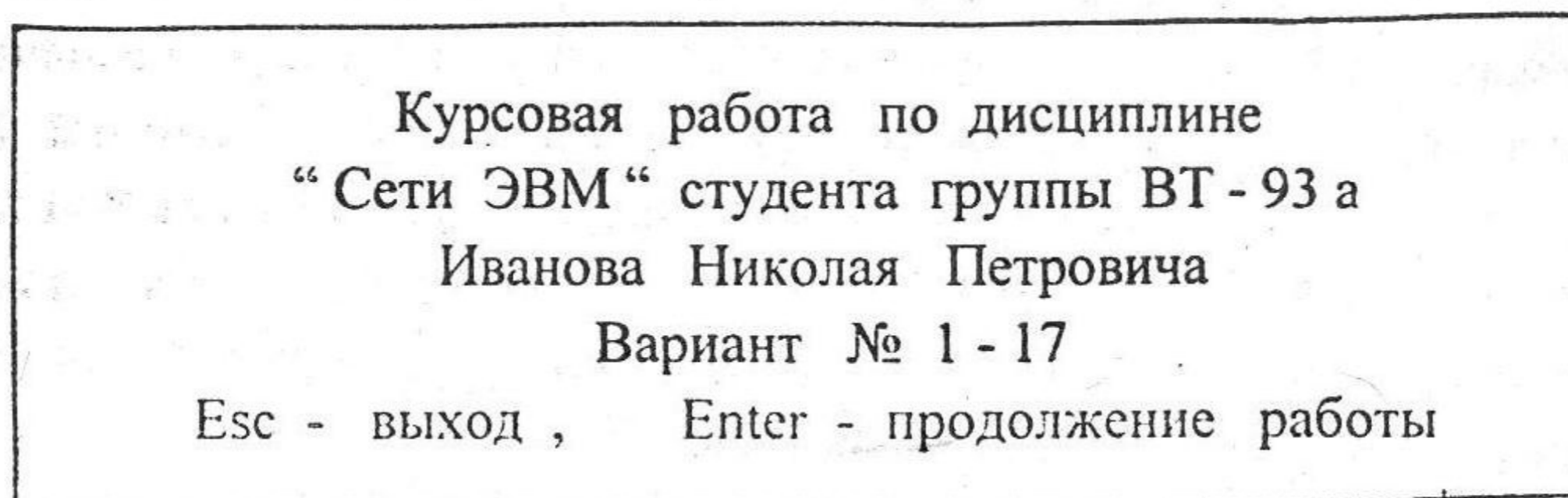


Рис. 1.1. Вид экрана дисплея после запуска программы

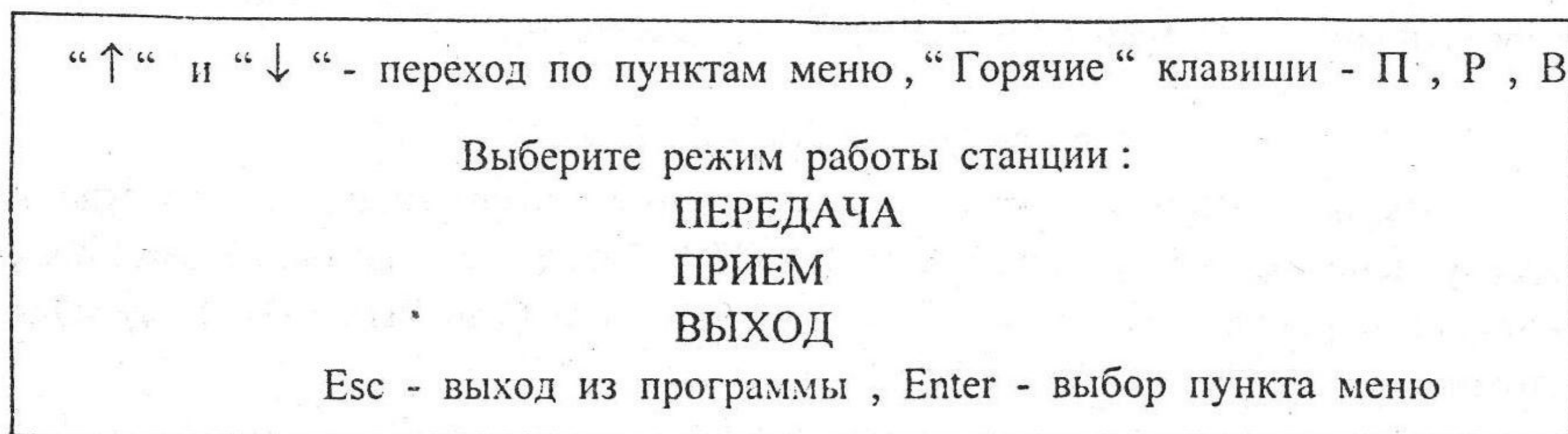


Рис. 1.2. Вид экрана дисплея после запуска первого меню

Так, например, клавиши управления курсором "↑" и "↓" могут быть использованы для перехода по пунктам меню. Выбор пункта меню (отображается в инверсном режиме) можно осуществить нажав на клавишу Enter, либо на одну из "горячих" клавиш: "П" - передача, "Р" - прием или "В" - выход. Если пользователь нажмет на клавишу Esc, программа завершит свою работу и передаст управление операционной системе.

При выборе режима работы сетевой станции (прием или передача) на экране дисплея появится следующее окно меню. В этом окне приведен список станций, с которыми может установить логический канал связи данная станция. После выбора номера взаимодействующей станции в программе происходит переход к третьему меню.

1.3. Разработка программных модулей

Основная функция программы - функция main(). В ней производится проверка текущего видеорежима, очистка экрана и "заливка" его фоновым

цветом , а также инициализация трехуровневого вертикального меню и информационных окон . При выходе из функции `main ()` происходит восстановление первоначального вида экрана .

Программа выполнена в виде отдельных программных модулей (функций и процедур) . Блок схема алгоритма взаимодействия этих модулей показана на рис . 1. 3. Рассмотрим назначение и состав отдельных модулей программы .

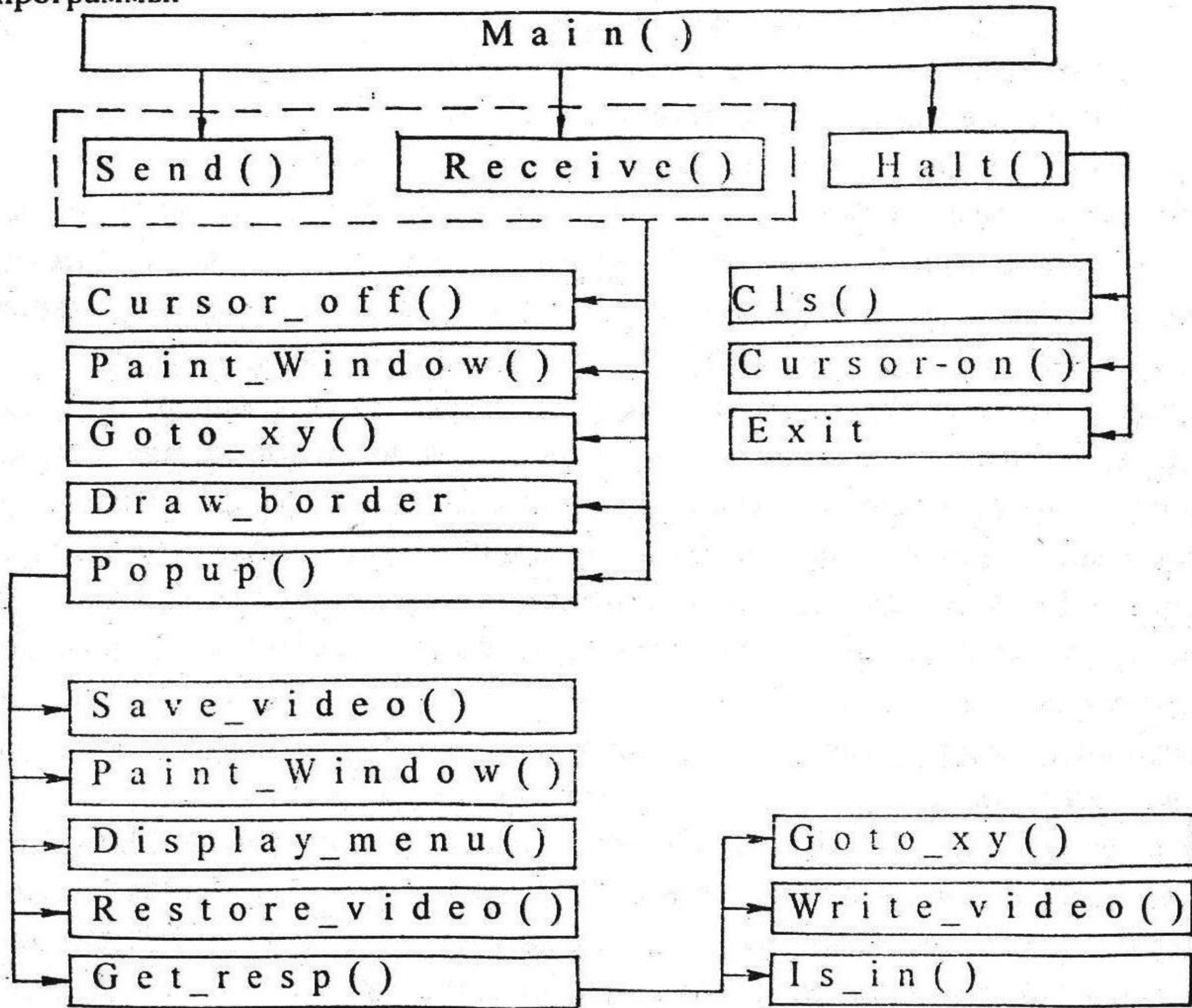


Рис . 1.3. Блок схема взаимодействия программных модулей

Вывод меню на экран и возврат в основную программу происходит с помощью функции `popup ()` , имеющей прототип [1] :

`popup (menu , keys , count , x , y , border) ,`

где `char * menu []` - текст меню ;

`char * keys` - " горячие " клавиши ;

`int count` - число альтернатив меню ;

`int x , y` - координата левого верхнего угла ;

`int border` - признак наличия рамки под меню .

Функция `popup ()` выполняет следующие действия :

- 1) проверяет выход координат X и Y за границы экрана ;
- 2) вычисляет размеры меню и проверяет , не выходит ли оно за границы экрана ;
- 3) динамически выделяет память для временного сохранения экрана , отведенного под меню ;
- 4) сохраняет область экрана под меню ;
- 5) высвечивает рамку под меню ;
- 6) высвечивает меню на своем месте ;
- 7) получает ответ пользователя ;
- 8) восстанавливает часть экрана , занимаемого меню .

Функция `save_video ()` сохраняет часть экрана , отведенного под меню , в выделенный динамический буфер и очищает эту часть экрана . Функция 8 прерывания 16 BIOS возвращает символ из текущей позиции курсора в регистр AL , а его атрибут - в регистр AH . Параметр `buf_ptr` - это целый указатель на область памяти , которая содержит информацию .

После сделанного из меню выбора необходимо восстановить вид экрана . Для этого предварительно запомненная в динамическом буфере информация восстанавливается назад в видеопамять . Функция `restore_video (` использует прерывание 16 BIOS функции 9) помещает информацию из буфера , на который указывает `buf_ptr` , на экран , заданный начальной и конечной координатами X и Y .

Функция `display_menu ()` высвечивает каждый элемент меню . Для этого она просто индексирует указатель как массив `menu [i]` .

Процедура `draw_border ()` высвечивает рамку вокруг меню . Для этого задаются координаты верхнего левого и правого нижнего углов меню . Процедура для создания рамки использует символы стандартного графического набора , используемого в машинах IBM PC .

Функция `get_resp ()` позволяет пользователю сделать выбор пункта меню . Пользователь с помощью клавиш "↑" , "↓" и "пробел" может перемещаться по пунктам меню . При этом выбранная строка освещается в инверсном режиме . Если нажать на "горячую" клавишу или клавишу Enter , то будет выбран один из пунктов меню и произойдет возврат в основную программу .

Функция `write_video ()` используется для записи строки символов на экран в позицию с координатами X и Y . Если строка выбрана (освещена) , она выводится на экран в инверсном режиме , если не выбрана - в нормальном режиме .

Функция `is_in ()` возвращает позицию нажатой “горячей” клавиши в строке. Если пользователь нажал не “горячую” клавишу, то в основную программу возвращается нуль.

С помощью функции `cls ()` происходит очистка экрана дисплея.

Функция `goto_xy ()` позволяет установить курсор в позицию экрана с координатами X и Y.

Функции `cursor_off ()` (гашение курсора) и `cursor_on ()` (включение курсора) используются в программе для управления видимостью курсора на экране дисплея.

Процедура `halt ()` очищает экран, высвечивает курсор и возвращает управление операционной системе.

Функция `paint_window ()` выполняет запись символа с заданным атрибутом на определенное место экрана, заданное координатами левого верхнего и правого нижнего углов.

Процедуры `send ()` и `receive ()` позволяют организовать работу с рассмотренными окнами меню. Во время вызова одной из этих процедур на экран выдаются следующие сообщения:

“Выберите номер станции, с которой вы желаете работать:”,

“Горячие клавиши - 1, 2, 3, 4”.

Далее происходит обращение к функции `popup ()` для отображения на экране второго меню. При выборе пользователем пункта этого меню на экране появляется новое сообщение: “Выберите номер сокета:”, а после обращения к функции `popup ()` - третье меню с пунктами предлагаемых номеров сокетов. После сделанного выбора управление передается в основную программу.

1.4. Анализ полученных результатов

Листинг программы оконного интерфейса, написанной на языке Borland C++ приведен в приложении 1. Результаты экспериментальных исследований подтвердили работоспособность программы `win_kufs.exe`. По результатам исследований составлен протокол (приложение 2). Разработанный оконный интерфейс позволяет задавать режимы работы станции ЛВС, такие как “Прием” и “Передача” кадров, а также определять адрес взаимодействующей станции и сокет прикладной задачи пользователя. В программе предусмотрена система подсказок и кратких пояснений по работе с меню. Перед началом работы программы выдается заставка об авторе курсовой работы. Сделанный пользователем выбор сетевых параметров будет использован в основной программе работы ЛВС.