

Счетчик второго канала СТ2 сетевого таймера запрограммирован на работу в пятом режиме. Он вырабатывает строб отрицательной полярности СТР. После инвертирования на элементе D13 этот строб совместно с сигналом АG3 обеспечивает выработку сигнала IMP, который запоминается на триггере D21. Выход этого триггера (сигнал СDT) используется в узле обработки коллизий адаптера.

В то же время, сигнал с одновибратора D12 через инвертор D16 поступает на вход разрешения работы счетчика нулевого канала таймера GATE0, блокируя на определенное время выработку тактовых сигналов T.

Другой вариант построения схемы детектора коллизий дан в [5].

2.7. Блок связи с системной шиной

Этот блок выполняет функцию адресации портов блоков приема и передачи кадров сетевого адаптера. Подключение шины данных адаптера сетевой станции происходит по линиям D7 - D0. Использование дешифратора адресного порта ввода - вывода позволяет формировать сигналы программного выбора микросхем адаптера, адресуемых как порты. Распределение адресного пространства ввода - вывода сетевого адаптера представлено в табл. 2.1.

Сигнал прерывания работы процессора IRQ от сетевого адаптера может возникнуть в следующих случаях:

- буфер FIFO1 передатчика пуст ($X1 = 1$) или полон на половину ($X3 = 1$);
- в буфер FIFO2 принят кадр ($X8 = 1$);
- кадр не может быть передан в сеть из-за превышения счетчика числа попыток максимального значения ($X6 = 1$).

Таблица 2.1

Адресация портов сетевого адаптера

Узел	Адрес порта	Назначение
FIFO1	0200H	Буфер передатчика
FIFO2	0202H	Буфер приемника
PI1 (0 канал)	0204H	Узел обработки коллизий
PI1 (1 канал)	0206H	Узел обработки коллизий
PI1 (2 канал)	0208H	Узел декодера
PI2 (0 канал)	020AH	Узел детектора коллизий
PI2 (2 канал)	020CH	Узел детектора коллизий
PC	020EH	Регистр состояния адаптера

Программа обработки сетевого прерывания считывает и анализирует содержимое регистра состояния адаптера. Выяснив причину прерывания, она производит его обработку, после чего возвращает управление основной программе.

2.8. Расчет временных параметров

Время подготовки кадра для передачи в моноканал:

$$T_{пк} = 6 \cdot T_f + 3 \cdot T_r + T_{кc},$$

где T_f - время считывания одного байта кадра из буфера FIFO1;
 T_r - время считывания одного байта кадра из памяти ROM;
 $T_{кc}$ - время вычисления и записи в регистр РК контрольной суммы кадра.

Время выдачи кадра в моноканал: $T_k = 88 \cdot T_1$;

где T_1 - период следования тактовых импульсов ТхС.

Для сети Ethernet скорость передачи данных составляет 10 МГц, то есть $T_1 = 100$ нс.

Период передачи кадров с учетом межкадровых интервалов составляет:

$$T = T_{пк} + T_k + T_{пн}.$$

Время записи одного кадра из RAM в FIFO1 в режиме ПДП:

$$T_{пдп} = T_{пн} + 6 (T_{dma} + T_{ram} + T_{fifo}),$$

где $T_{пн}$ - время захвата шины контроллером DMA;

T_{dma} - время выдачи DMA управляющих сигналов \overline{IOW} и \overline{MR} ;

T_{ram} - время считывания байта кадра из основной памяти;

T_{fifo} - время записи байта кадра в буфер FIFO.

Аналогичные временные соотношения могут быть составлены и для режима приема кадра из сети. Воспользовавшись справочными данными на временные параметры микросхем [15 - 18], можно выполнить в первом приближении количественную оценку полученных результатов и сделать соответствующие выводы.

В заключении по второй части курсовой работы следует выбрать схему генератора тактовых сигналов схемы синхронизации и привести временные диаграммы работы блоков передачи и приема кадров сетевой станции.

3. РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СТАНЦИИ

3.1. Структура программного обеспечения

Программное обеспечение станции ЛВС с методом доступа CSMA / CD представляет собой набор программных модулей, предназначенных для

управления прикладным и канальным сетевыми уровнями. Информационные потоки между сетевыми уровнями отражает рис. 3.1.

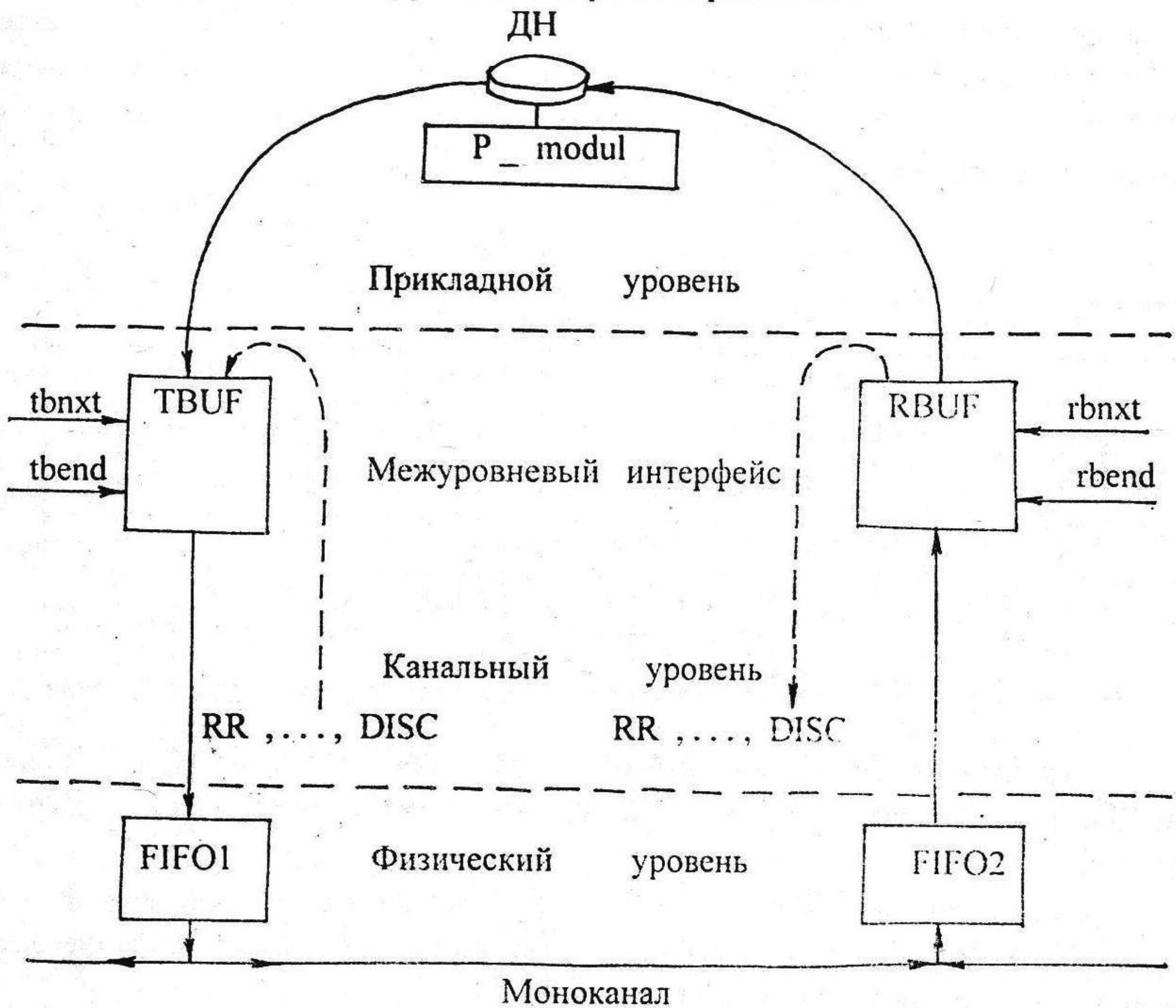


Рис. 3.1. Взаимодействие сетевых уровней системы

Прикладной уровень (ПУ) осуществляет пересылку информационных I - кадров между дисковым накопителем (ДН) и промежуточными буферами передачи TBUF и приема RBUF , которые входят в состав межуровневого интерфейса . На прикладном уровне происходит также решение прикладной задачи пользователя (P_modul) .

Канальный уровень (КУ) осуществляет открытие и закрытие логического канала связи между сетевыми станциями , контроль по заданному модулю прохождения I - кадров , выявление и идентификацию возникших ошибок .

На физическом уровне (ФУ) происходит сборка и разборка кадров , выполняется вставка и удаление нулей из битовой последовательности кадра , осуществляется передача и прием кадров из сетевой среды .

Промежуточные буферы TBUF и RBUF входят в состав оперативной памяти сетевого компьютера. Они выполнены многобайтными, переменной длины. С помощью указателей tbnxt и rbnxt происходит адресация ячеек буферов при выполнении операций записи или считывания очередного байта кадра.

Работа канального уровня сети организована на основе протокола HDLC [9, 10, 13]. Рассмотрим состав и назначение кадров, используемых в сетевом программном обеспечении (табл. 3.1.).

Таблица 3.1

Кадры канального уровня

Поле AC кадра	Код	Назначение кадра
I	0000 1000	Информационный кадр
SNRM	1100 1001	Установить режим нормальных ответов
UA	1100 1110	Ненумерованное подтверждение
RR	1000 1000	Готов к приему
RNR	1010 1000	Неготов к приему
DISC	1100 1010	Разъединить канал

Программное обеспечение станции ЛВС состоит из модуля основной части Main и обработчика прерываний от сетевого порта NET_Handler (рис. 3.2.).

Схема алгоритма основного модуля программы представлена на рис. 3.3. В основной части программы пока не будет нажата клавиша Esc, происходит вызов следующих модулей:

Clear_flags () - установка в исходное положение флагов и указателей;

Initalize () - инициализация узлов сетевого порта (таймеров, DMA, PC) и установка вектора сетевого прерывания;

Menu () - программный модуль из первой части курсовой работы (устанавливает режим работы станции, номер сокета, адрес взаимодействующей станции);

Window () - отображает на экране монитора два окна для приемного и передающего блоков станции, в которых можно наблюдать последовательности переданных и поступивших кадров;

Schedule () - модуль планирования. С помощью флага SP этот модуль активизирует отдельные фазы работы сетевой станции. Эти фазы могут также активизироваться непосредственно прерываниями сетевого порта.

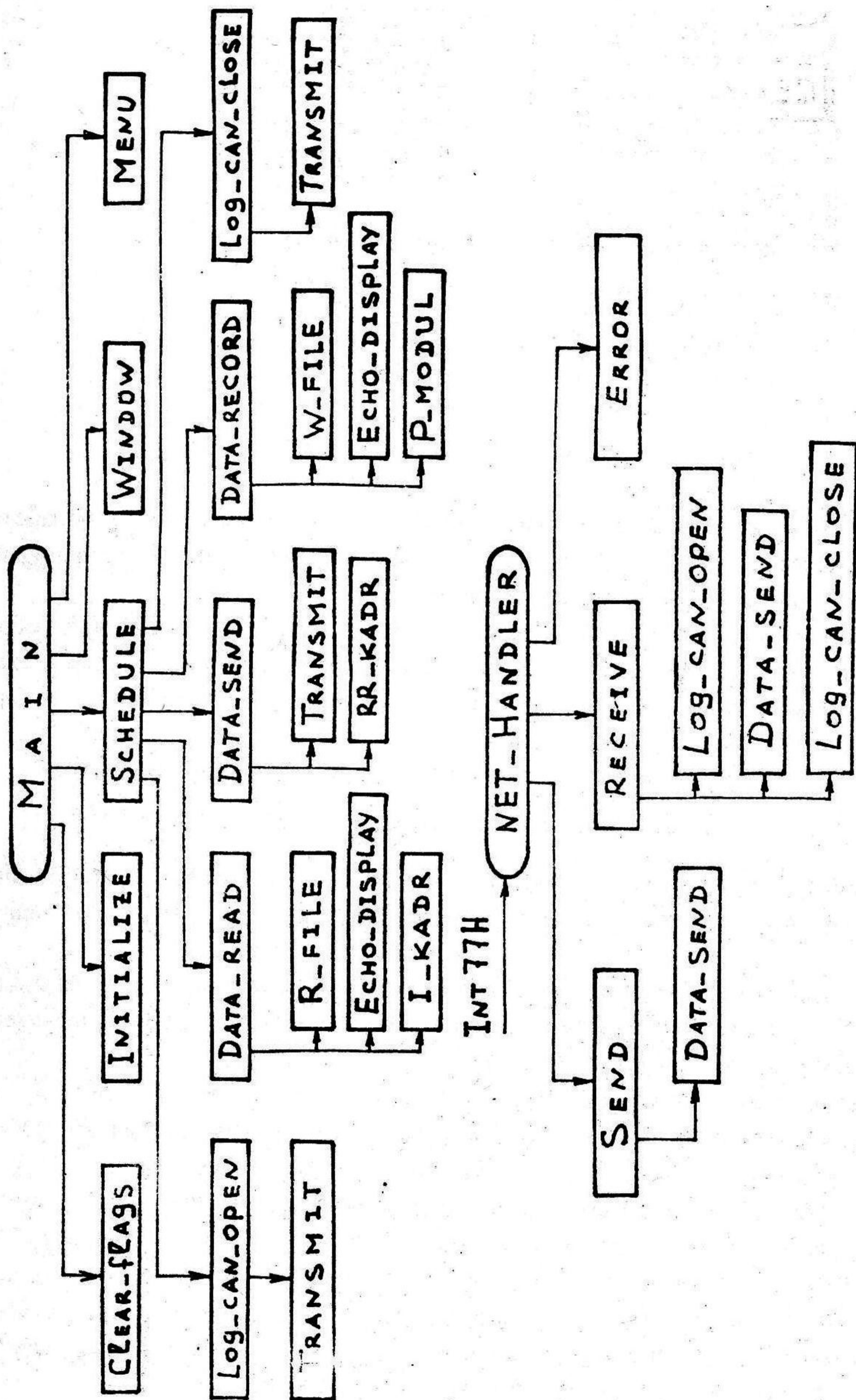


Рис. 3.2. Структура программы

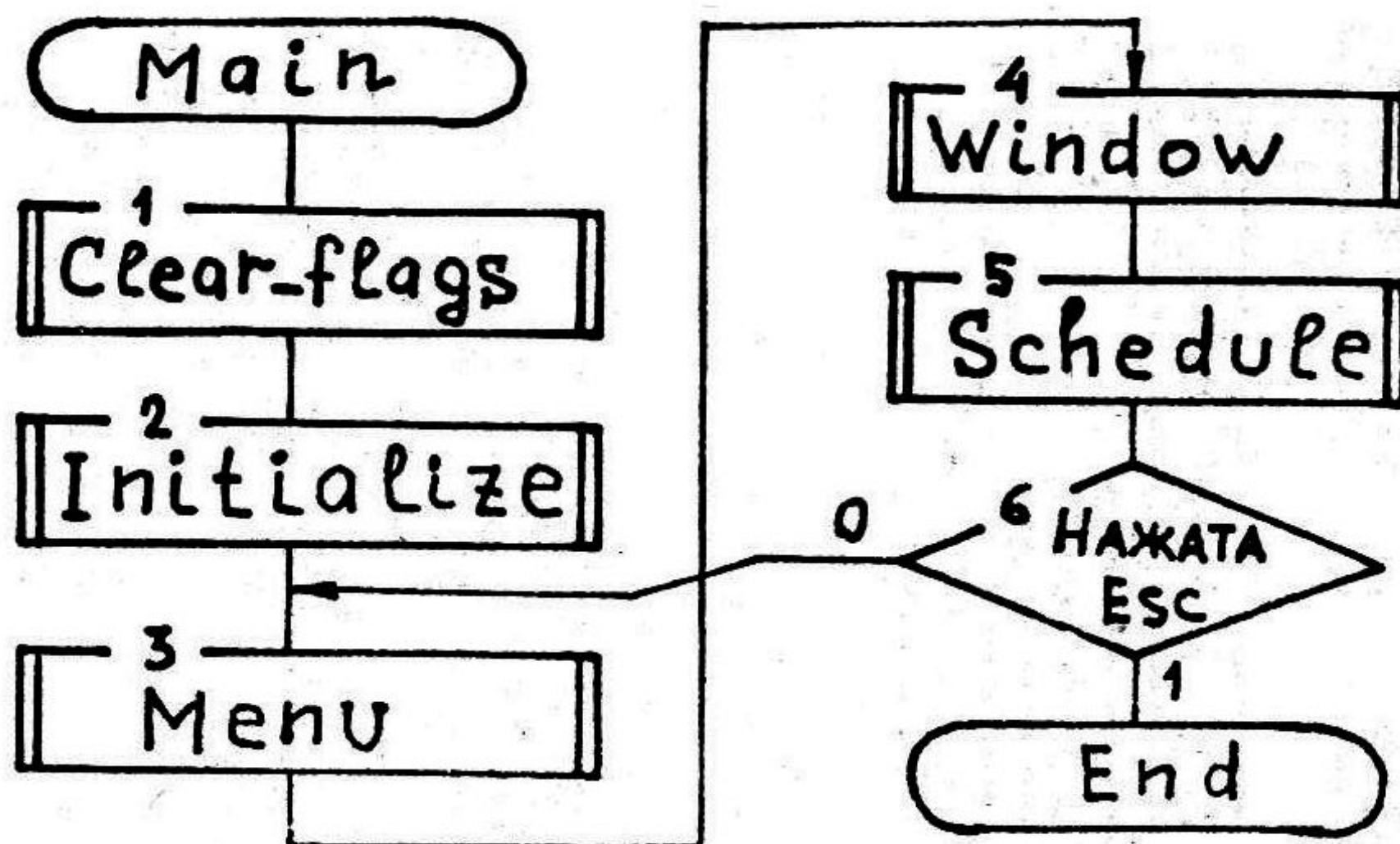


Рис. 3.3. Схема алгоритма модуля Main

Обработчик прерывания `NET_Handler()` настраивается на обработку сетевого прерывания `IRQ 15`. Три возможных причины возникновения сетевого прерывания были рассмотрены выше.

В процессе обработки того или иного прерывания, а также при работе планировщика `Schedule()`, происходит вызов отдельных программных модулей. Рассмотрим назначение и состав основных программных модулей станции.

3.2. Разработка программных модулей станции

3.2.1. Модуль планирования `Schedule()`

Работу сетевой станции можно разделить на три фазы: установление логического канала связи, передача кадров и закрытие логического канала связи.

Управление этими фазами, а также взаимодействием между сетевыми уровнями осуществляет программа планирования. Для этой цели используется двухбитовый флаг `SP` (табл. 3.2).

Схема алгоритма модуля `Schedule()` показана на рис. 3.4. Вспомогательные переменные `n1 - n3` позволяют идентифицировать работу сетевого узла в каждой фазе в режимах передачи или приема.

Инициатором создания логического канала связи является станция, работающая в режиме "передача" кадров. Для этого в первой фазе работы (`SP = 01`) вызывается модуль `Log_cap_open()`. Этот модуль адресует кадр `SNRM` той станции, с которой будет установлен логический канал связи. Получив ответный кадр подтверждения `UA`, модуль устанавливает признак логического канала `log_cap` и сбрасывает флаг `SP`.

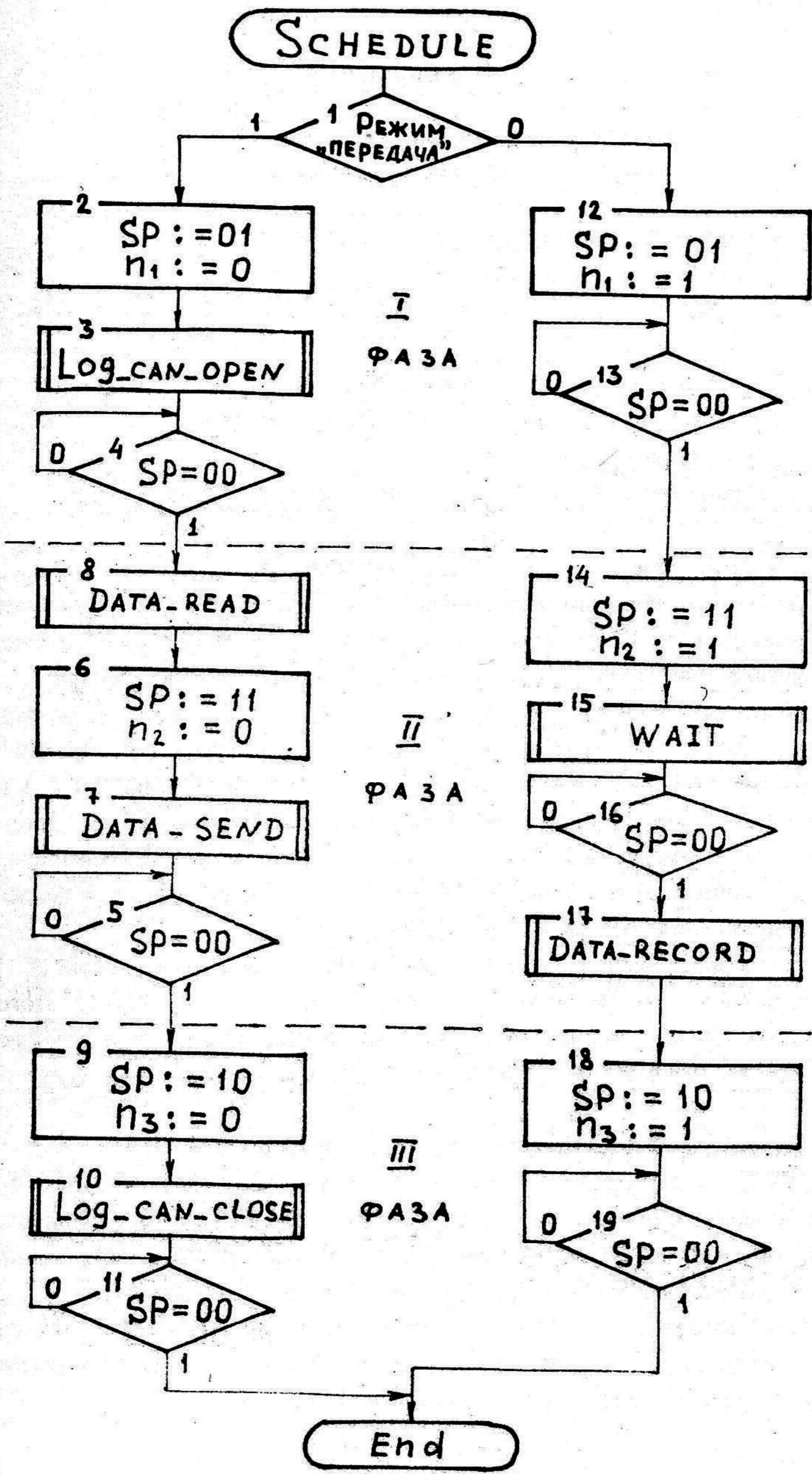


Рис. 3.4. Схема алгоритма модуля Schedule ()

Назначение разрядов флага SP

Флаг SP	Назначение
0 1	Запрос канальному уровню установить логический канал связи для передачи кадров
1 0	Запрос канальному уровню закрыть логический канал связи
1 1	Запрос канальному уровню передать кадры из TBUF в сеть или переслать принятые кадры из RBUF прикладному уровню
0 0	Ответ канального уровня после выполненного запроса

Во второй фазе работы ($SP = 11$) вызывается модуль прикладного уровня `Data_read ()`, который читает файл `R_FILE` с диска, формирует из него информационные I - кадры, пересылает их в буфер передатчика TBUF и отображает эти кадры на выделенной половине экрана дисплея.

Далее вызывается модуль канального уровня `Data_send`, который осуществляет пересылку I - кадров из буфера TBUF через сеть станции назначения. На приемной станции с помощью процедур `NET_Handler ()` и `Wait ()` поступившие кадры размещаются в приемном буфере RBUF. Затем вызывается модуль прикладного уровня `Data_record ()`, с помощью которого кадры пересылаются в файл `W_FILE` на диске и передаются прикладному модулю программы пользователя `P_modul`.

На третьей фазе работы с помощью модуля `Log_cap_close ()` станции обмениваются между собой кадрами DISC и UA. После этого логический канал связи закрывается и управление передается модулю `Menu ()` для задания параметров нового сеанса связи.

3.2.2. Модуль `Log_cap_open ()`

Модуль `Log_cap_open ()` входит в состав канального уровня. Он позволяет установить логический канал связи между двумя сетевыми станциями для обмена информацией (рис. 3.5). Адреса взаимодействующих станций DA и SA определены в процессе выполнения модуля `Menu ()`.

В режиме "передача" ($n1 = 0$) обращение к модулю происходит от планировщика `Schedule ()`, в режиме "прием" ($n1 = 1$) - от процедуры `Receive ()` при обработке сетевого аппаратного прерывания.

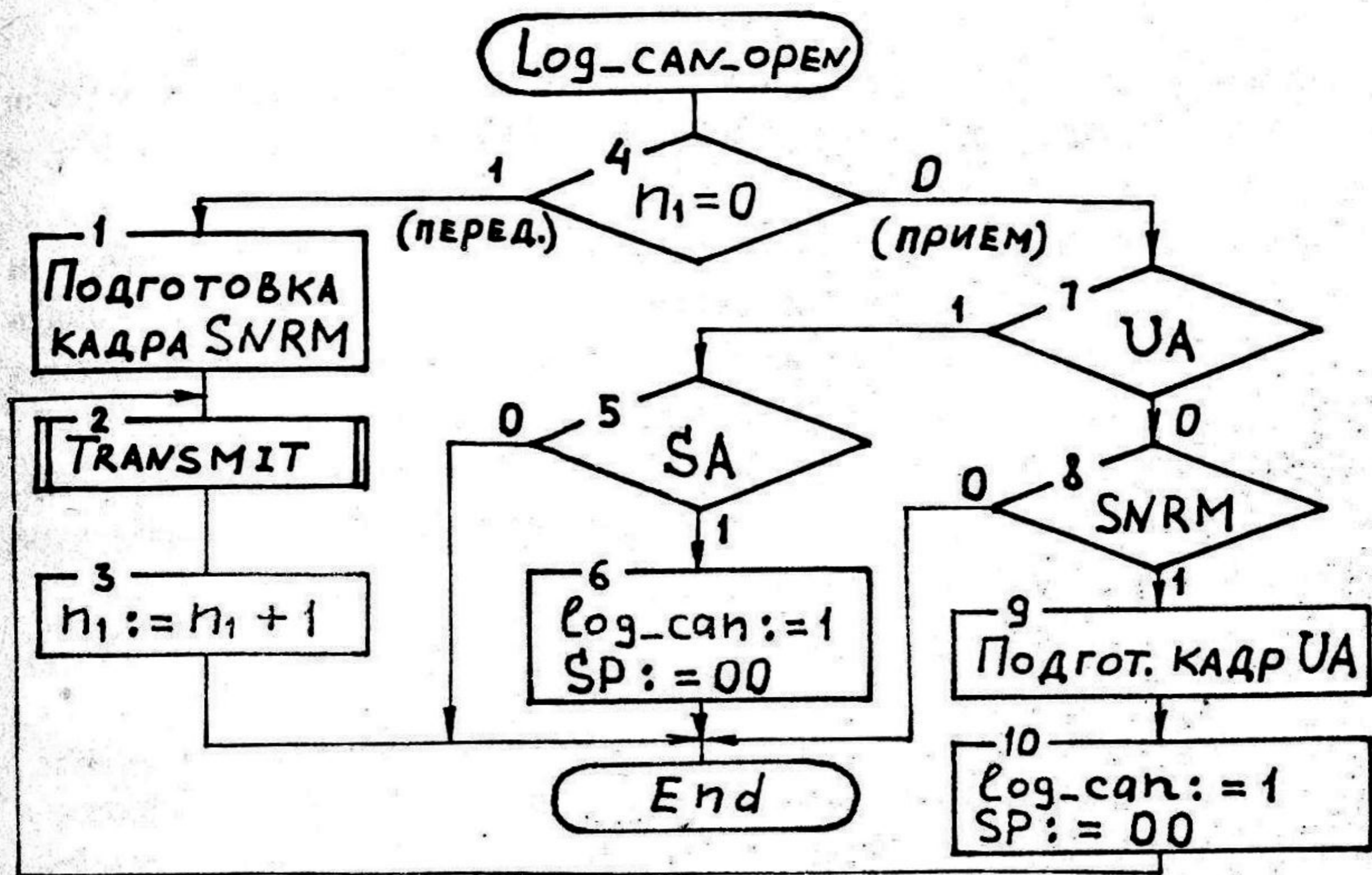


Рис. 3.5. Схема алгоритма модуля Log_can_open ()

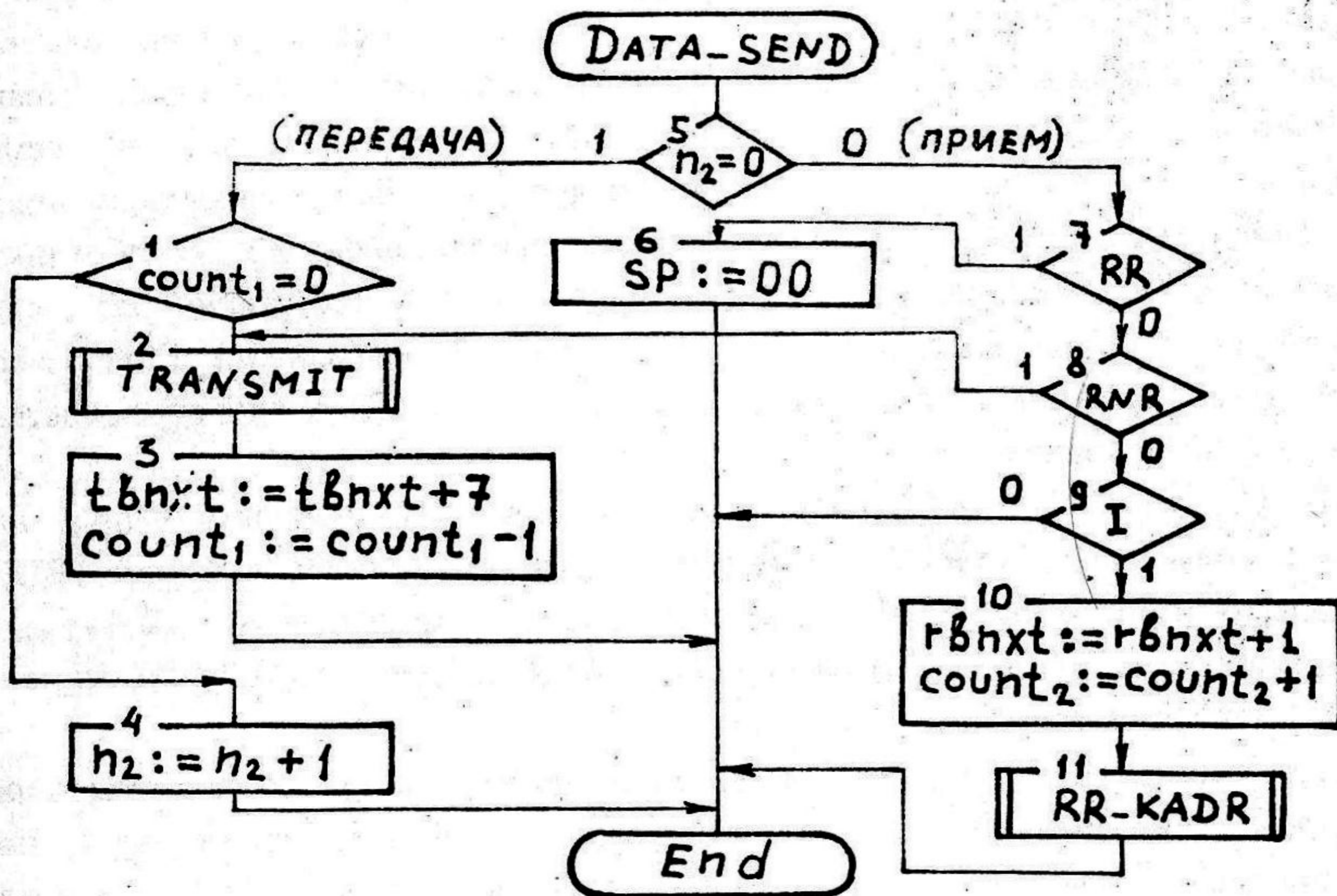


Рис. 3.6. Схема алгоритма модуля Data_send ()

Сформированный на передающей станции кадр с помощью модуля Transmit () передается из буфера TBUF в FIFO1 и далее по сети в приемный блок станции назначения.

Получив кадр SNRM, станция назначения подготавливает ответ и выдает кадр UA, а также устанавливает признак логического канала log_cap в единицу. Исходная станция после получения кадра подтверждения UA и проверки адреса SA отправителя кадра, в свою очередь, устанавливает признак log_cap и сбрасывает флаг запроса SP. На этом завершается работа первой фазы. Аналогичным образом работает процедура закрытия канала связи Log_cap_close ().

3.2.3. Модуль Data_send ()

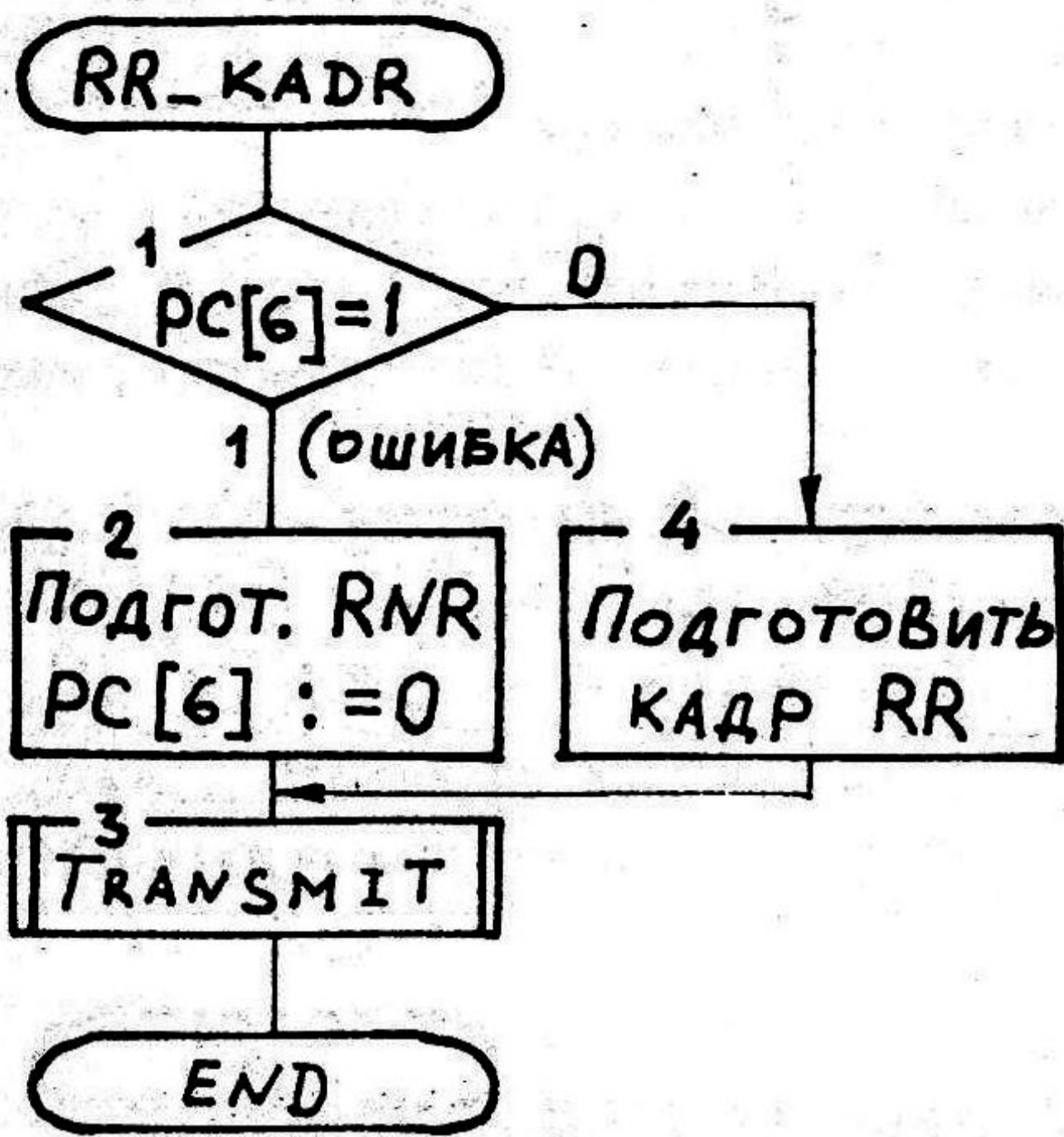
Модуль Data_send () входит в состав программ прикладного уровня. Модуль позволяет организовать передачу и прием информационных кадров, а также супервизорных кадров подтверждения приема (RR) или неприема (RNR) I - кадров. Схема алгоритма модуля Data_send () приведена на рис. 3.6.

В режиме "передача" ($n_2 = 0$) происходит выдача в сеть I - кадров, предварительно сформированных в буфере TBUF. После передачи одного I - кадра с помощью процедуры Transmit (), счетчик числа кадров count уменьшается на единицу, а указатель tbnxt наращивается на семь (указывает на начало следующего кадра). Передача следующих информационных кадров происходит с использованием обработчика прерываний NET_Handler (), который получает управление всякий раз, как только буфер передатчика FIFO1 окажется пустым. Вызов модуля Data_send в этом случае происходит из процедуры Send (), которая входит в состав обработчика прерывания.

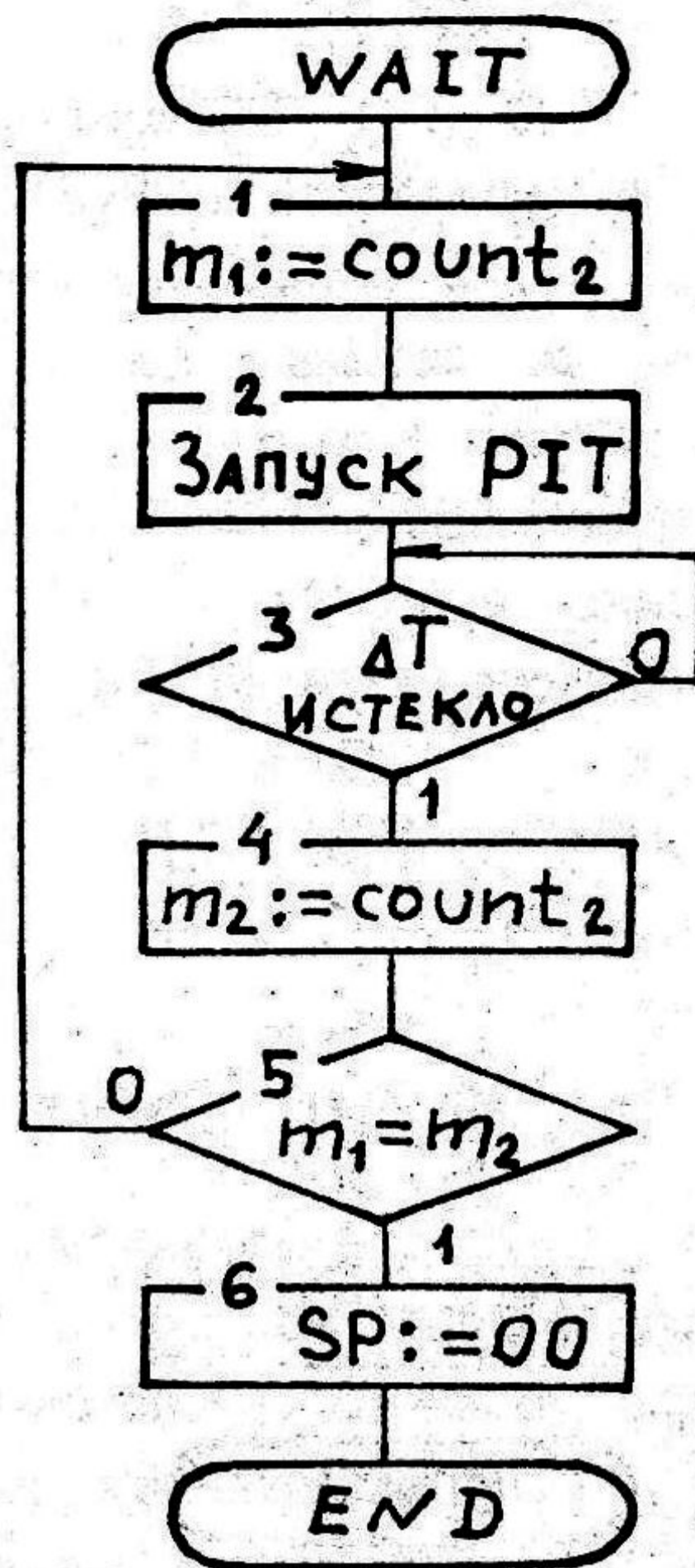
В режиме "прием" происходит проверка поступившего кадра на ошибки (модуль RR_kadr), размещение кадра без ошибок в буфер TBUF и подготовка параметров tbnxt и count к приему следующего I - кадра. Если кадр принят с ошибкой, выдается ответ RNR с тем, чтобы станция-источник повторила передачу I - кадра.

Так как заранее на приемной станции неизвестно число поступающих I - кадров, на прием очередного I - кадра отводится задержка T_3 , по истечении которой в модуле Wait () сбрасывается флаг SP и выдается кадр подтверждения RR полученных кадров. На этом вторая фаза работы станции завершается.

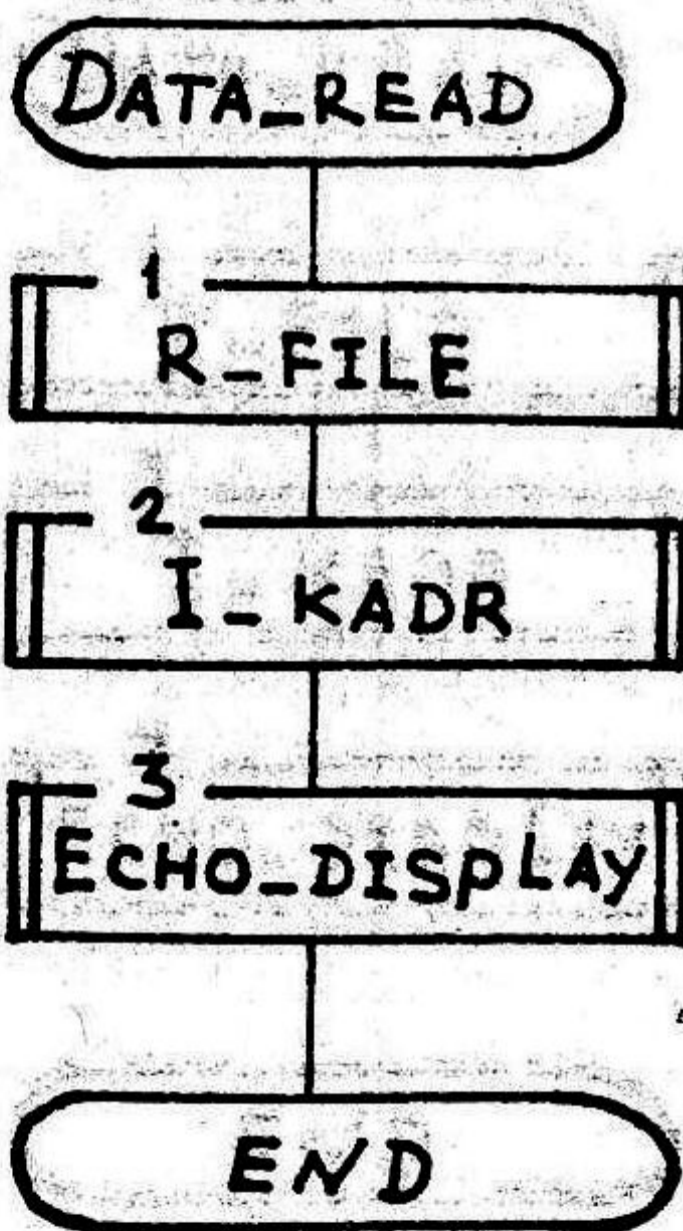
Схемы алгоритмов вспомогательных модулей системы приведены на рис. 3.7.



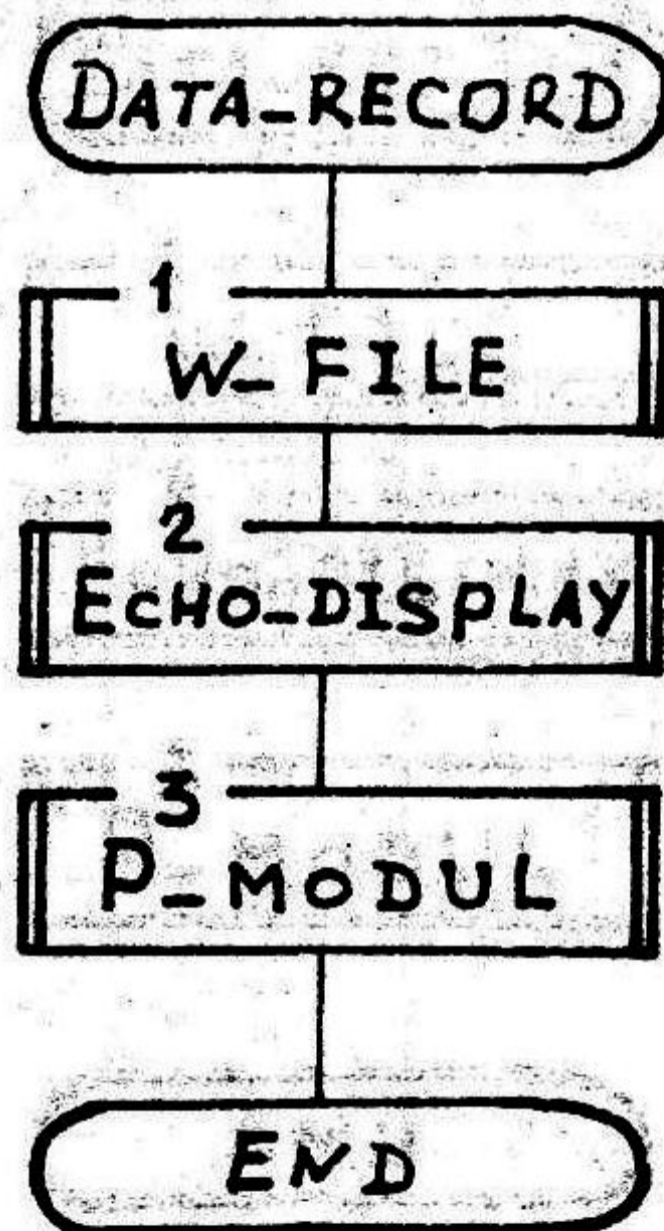
а)



б)



в)



г)

Рис. 3.7. Схемы алгоритмов вспомогательных модулей

3.2.4. Обработчик сетевого прерывания

Вызов обработчика NET_Handler () происходит по вектору прерывания 77h , установленному функцией Initialize () , если разрешены прерывания от сетевого порта . Схема алгоритма обработчика прерываний приведена на рис . 3 . 8 , а .

Получив управление , обработчик NET_Handler () уточняет причину возникновения прерывания чтением содержимого регистра состояния PC сетевого адаптера . При PC [0] = 1 выполняется вызов процедуры Send () (рис . 3 . 8 , б) . Если PC [7] = 1 , вызывается процедура Receive () (рис . 3 . 8 , в) . При PC [5] = 1 происходит вызов процедуры обработки ошибок Error () .

4. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Цель экспериментальных исследований состоит в проверке работоспособности алгоритмов и программ прикладного и канального уровней сетевого программного обеспечения . Эксперимент проводится на двух сетевых компьютерах с использованием последовательного интерфейса RS - 232 [19] . Все ПЭВМ , выпускаемые в настоящее время , поддерживают этот интерфейс при помощи последовательных COM - портов .

При выполнении экспериментальных исследований использовано подключение рабочих станций ЛВС к интерфейсу RS - 232 с использованием токовой петли 20 мА (рис . 4 . 1) .

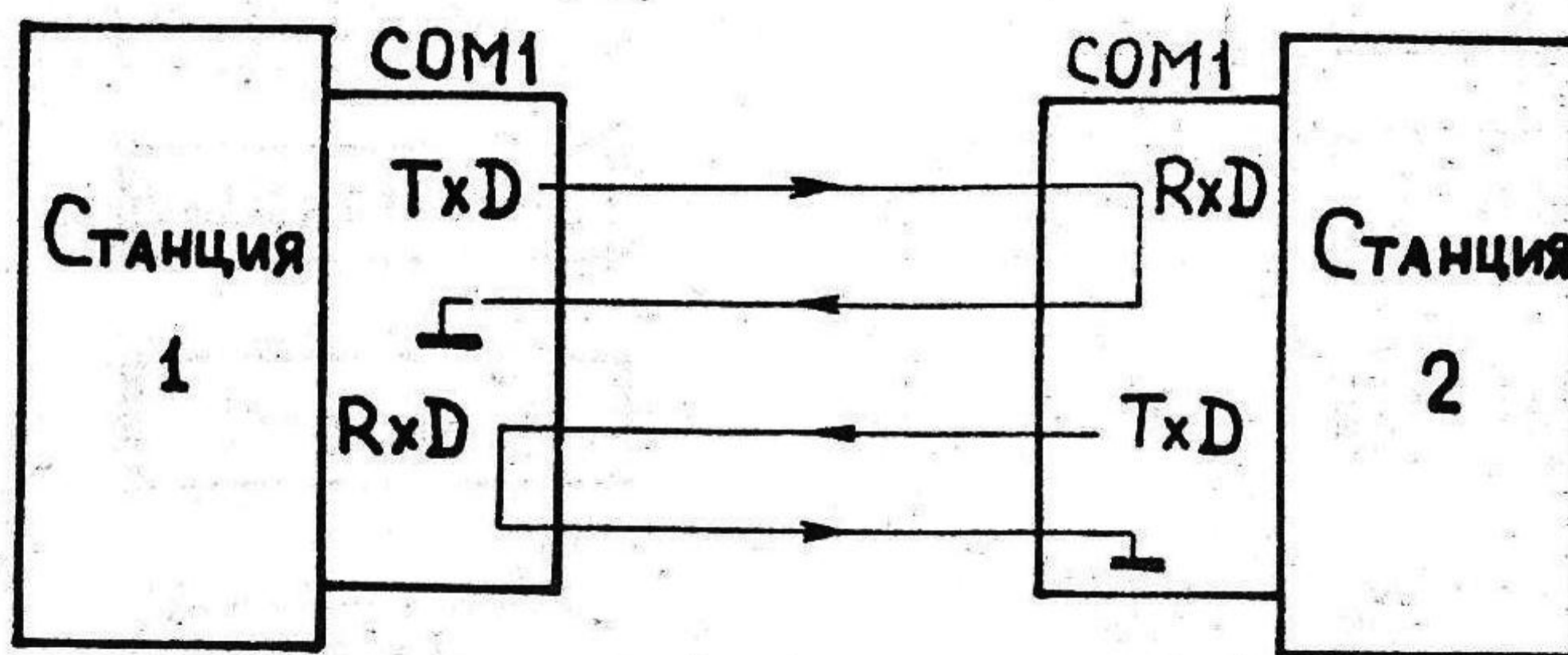
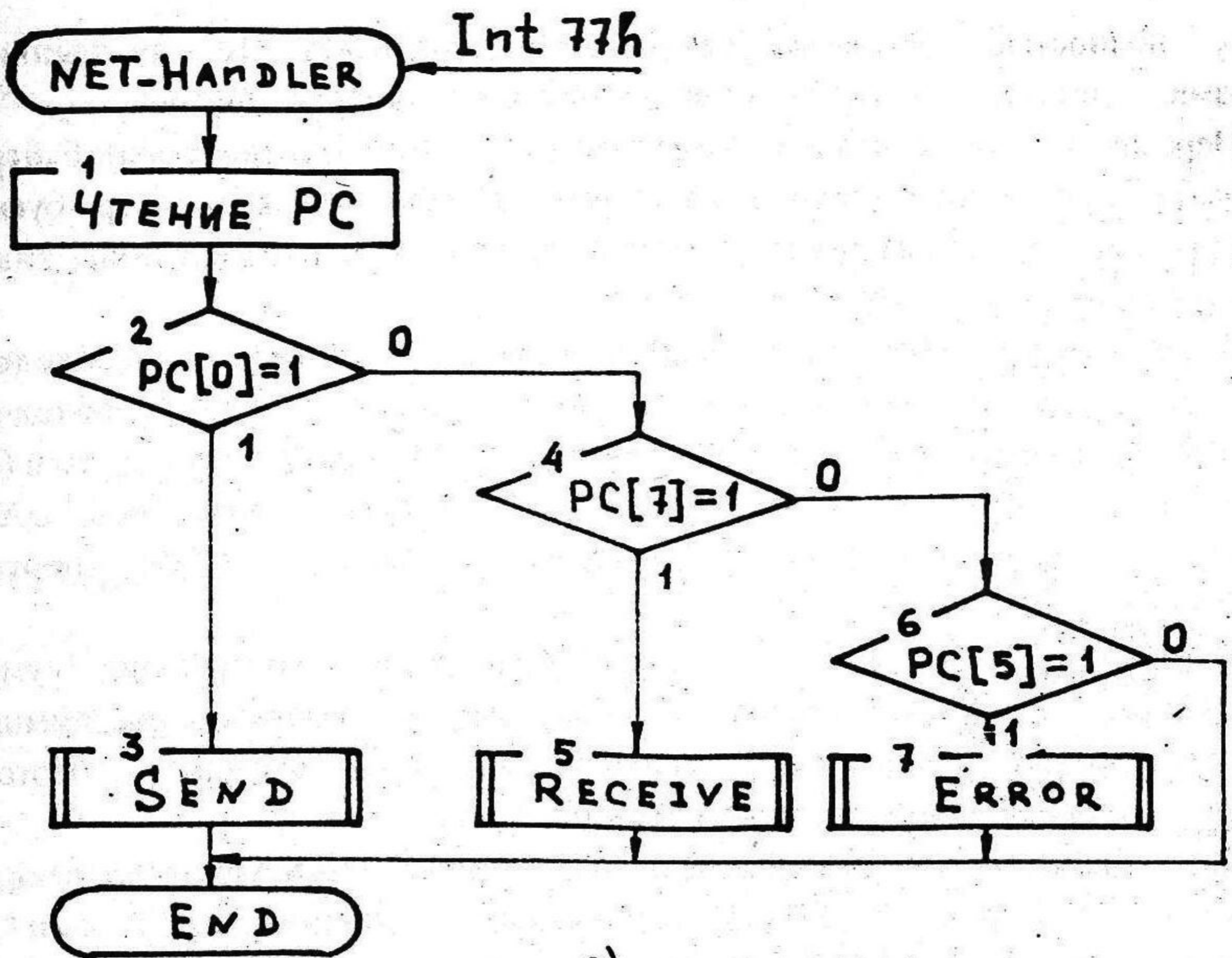
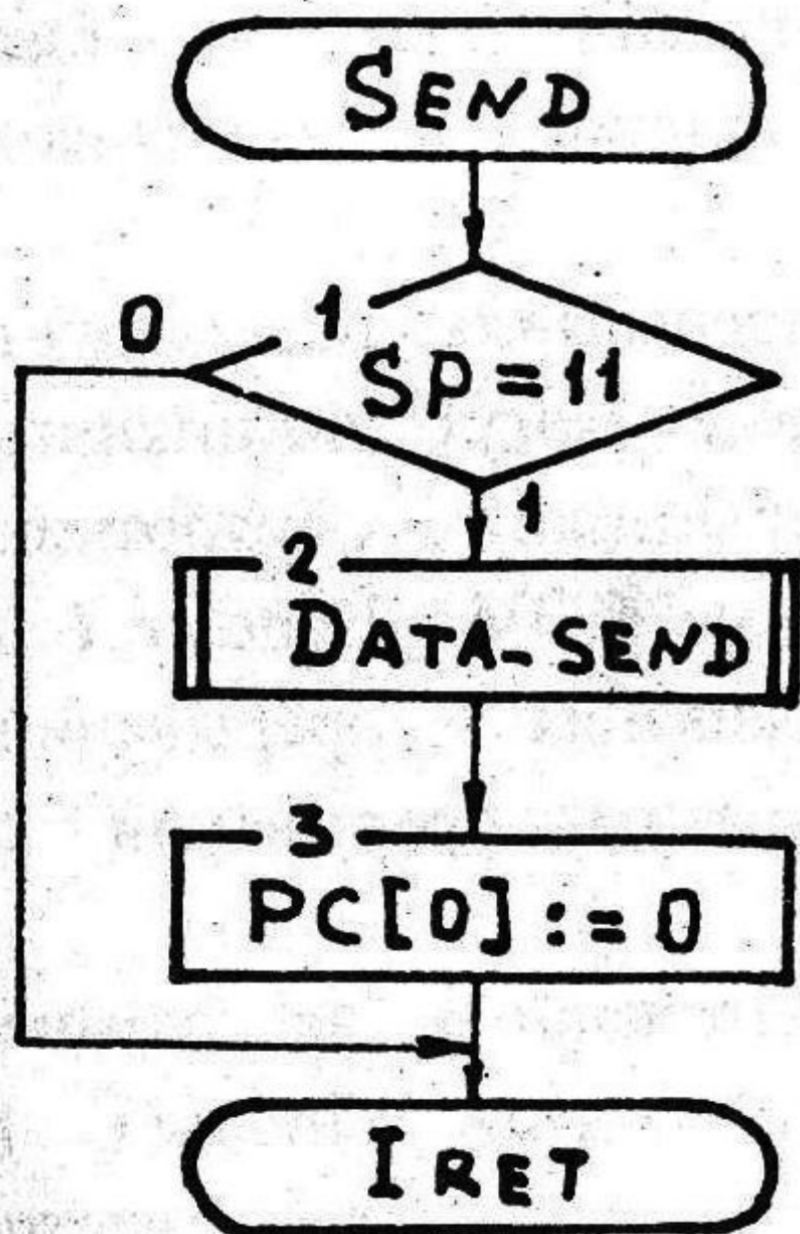


Рис . 4 . 1 . Схема подключения рабочих станций

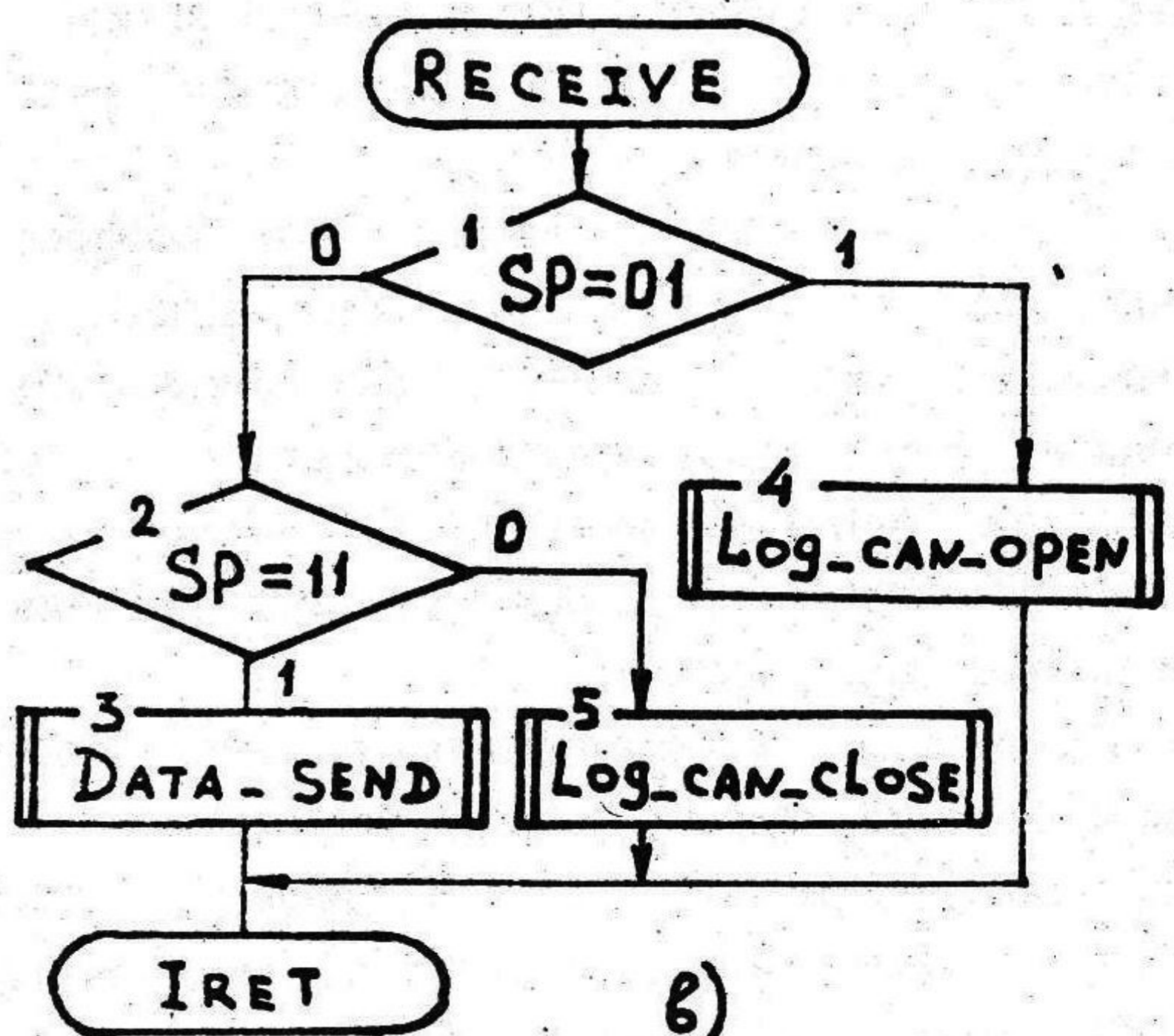
Вместо сетевого адаптера используется адаптер асинхронной связи (UART) на основе микросхемы i8250 [23] . Адаптер на основе UART



a)



б)



в)

Рис. 3.8. Процедуры сетевого обработчика прерываний

является полностью программируемым . Он управляет 10 внутренними регистрами , доступ к которым осуществляется по номерам портов .

Передача и прием кадров происходит побайтно . Для передачи байта в последовательную линию связи необходимо записать его в регистр буфера передатчика . После приема байта из линии связи в регистр буфера приемника , его можно прочитать в процессор .

В программное обеспечение для проведения экспериментов необходимо внести некоторые изменения . Вместо модуля Initialize () необходимо разработать программу инициализации адаптера COM - порта : Install_com () . Помимо задания скорости передачи байтов и других параметров UART программа выполняет установку обработчика прерываний от COM - порта - COM_Handler () .

Буферы FIFO1 и FIFO2 , а также узлы расчета контрольной суммы кадра , сборки и разборки кадров целесообразно организовать программным путем . Узлы DMA и анализатора коллизий моделировать сложно , поэтому в экспериментальной части они будут отсутствовать .

Вызов обработчика прерываний от COM - порта происходит по вектору прерываний 0Ch (адаптер COM1) , установленным функцией Install_com () . Получив управление , обработчик прерывания уточняет причину возникшего прерывания чтением содержимого регистра идентификации прерывания IIR (входит в состав UART) . Если имеется активное прерывание (младший бит регистра IIR равен нулю) обработчик выделяет причину и выполняет необходимые действия .

Перед завершением своей работы обработчик прерывания анализирует наличие еще не обработанных прерываний , которые могут возникнуть одновременно . При наличии необработанного прерывания обработчик продолжает его обработку . Завершается программа COM_Handler () инструкцией очистки внутреннего регистра обслуживания прерываний контроллера i8259 для разрешения обслуживания аппаратных прерываний с более низким приоритетом .

Результаты экспериментальных исследований отражаются на экране дисплея . Рабочая область экрана создается с помощью модуля Window () при запуске программы Main () . Экран разделяется на две части (рис . 4 . 2) . На левой половине отображаются переданные в сеть кадры , на правой - принятые из сети кадры . По результатам исследований оформляется протокол , который подписывает руководитель курсовой работы . Протокол помещается в приложение № 4 .

Передан кадр SNRM

Принят кадр UA

Между станциями с номерами 75 и 168 установлен логический канал

Передан I - кадр с номером 1

Передан I - кадр с номером 2

Передан I - кадр с номером 3

Передан I - кадр с номером 4

Передан I - кадр с номером 5

Принят кадр UA

Передан кадр DISC

Принят кадр UA

Логический канал связи между станциями 75 и 168 закрыт

Рис . 4 . 2 . Вид экрана дисплея при проведении экспериментов

Пример оформления титульного листа пояснительной записки

МИНИСТЕРСТВО ОБРАЗОВАНИЯ УКРАИНЫ

ДОНЕЦКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Кафедра “Электронных вычислительных машин”

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе по дисциплине “Сети ЭВМ”

на тему : “Разработка аппаратной части и программного обеспечения станции локальной вычислительной сети”

Разработал студент группы ВТ - 93а-8 факультета ВТИ

----- Иванов И.И.

Руководитель работы ----- Петров П.П.

Заведующий кафедрой ЭВМ

Сидоров С.С.

Донецк 1997

Пример оформления технического задания

УТВЕРЖДАЮ
Зав. кафедрой ЭВМ
проф. Святный В. А.

9 февраля 1997 г.

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

студенту группы ВТ - 93 а - 8 Иванову И. И.

Срок сдачи работы 17 мая 1997 г.

Исходные данные :

Вариант : 1 - 17

Данные для разработки оконного интерфейса :

- доступ к экрану : через функции BIOS ;
- цвет экрана меню : голубой ;
- координаты левого верхнего угла : 5 , 30 ;
- задание рамки : одинарная ;
- минимальное количество меню : 3 .

Данные для проектирования сетевой станции :

- топология сети : шина ;
- метод доступа : CSMA / CD ;
- среда передачи кадров : коаксиальный кабель ;
- размер буфера FIFO : 128 байт ;
- межкадровый интервал : 6 мкс ;
- базовый адрес станции : 0320H ;
- линия сетевого прерывания : IRQ 10 ;
- элементная база : K155 , K1810 , i8250 , i82586 , i8237 ;
- число попыток передачи кадра после коллизии : 3 ;
- реализуемые сетевые уровни : канал. , физич. , приклад ;
- формат кадра канального уровня :

1	6	6	1	1	256	2	1
F	DA	SA	CMD	L	D	CRC	F

где F - флаг (0 111 111 0) ;

DA - адрес станции назначения (6 байт) ;

SA - адрес исходной станции (6 байт) ;

CMD - тип кадра (1 байт) ;

L - поле длины информационной области (9 бит) ;

D - информационная область кадра (512 байт) ;

CRC - поле проверочной последовательности (2 байта) ;

Вид порождающего полинома :

$$P(x) = x^{16} + x^{12} + x^5 + 1.$$

- формат кадра прикладного уровня :

1 6 1 1 256

0000 0001	DA	C	L	D
-----------	----	---	---	---

где DA - адрес станции назначения (6 байт) ;

C - сокет (номер прикладной задачи) ;

L - длина информационной области (9 бит) ;

D - информационная область (512 байт).

$$DA = 100 ; SA = 56 ; C = 1.$$

Задача , решаемая на прикладном уровне :

$$y = \prod_{i=1}^{17} a_i + b , a_1 = b = 8 ; a_{i+1} = a_i + 1 , i = 2 \div 17$$

Разделы пояснительной записки , подлежащие разработке :

- оконный интерфейс для задания режимов работы станции ;
- функциональные электрические схемы станции и ее узлов ;
- алгоритмы работы сетевой станции ;
- программное обеспечение станции ;
- результаты экспериментальных исследований ;
- расчет основных технических характеристик станции .

График выполнения курсовой работы :

1 неделя - составление и анализ технического задания ,
уяснение полученной задачи .

2 - 4 недели - разработка оконного интерфейса для задания
режимов работы 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 .

Дата выдачи задания : 9 февраля 1997 г.

Задание принял к исполнению : ----- Иванов И.И.

Руководитель работы : ----- Сидоров С.С.

Пример оформления протокола № 1

ПРОТОКОЛ № 1

Исследование оконного интерфейса для задания режимов работы сетевой станции

Выполнил : студент группы ВТ - 93 а Иванов И. И.

Вариант : 1 - 17

Исходные данные :

- доступ к экрану через BIOS ;
- координаты левого верхнего угла : 5 , 30 ;
- цвет меню : красный ;
- рамка под меню : одинарная ;
- количество меню : 3 .

В результате работы программы получены результаты :

(Приводится вид окон меню в различных режимах работы станции)

Руководитель курсовой работы ----- Сидоров С. С.

Пример оформления протокола № 2

ПРОТОКОЛ № 2

Исследование программного обеспечения сетевой станции

Выполнил : студент группы ВТ - 93 а Иванов И.И.

Вариант № 1 - 17

Исходные данные :

- реализуемые сетевые уровни : физич. , канал. , прикладной ;
- протокол : HDLC ;
- топология сети : шина ;
- метод доступа : CSMA / CD ;
- тип кадров канального уровня : RR , RNR , UA , DISC .
- задача прикладного уровня : $y = \prod_{i=1}^{17} a_i + b$;
- интерфейс физического уровня : RS - 232 .

В результате проведенных экспериментов были получены следующие результаты :

1)

2)

.

Руководитель курсовой работы ----- Сидоров С.С.

УЧЕБНОЕ ИЗДАНИЕ

Методические указания и задания к курсовой работе
по дисциплине «Сети ЭВМ»
(для студентов специальности 7.09.15.01 дневной и вечерней форм обучения)

Составитель: ГУБАРЬ ЮРИЙ ВЛАДИМИРОВИЧ

Подп. к печати 13.12.96 Формат 60x84 1/16 Бумага тип. № 2 Печать офсетная
Усл. печ. л. 3,49 Усл. кр.-отт. 3,71 Уч.-изд, л, 66 Тираж 250 экз, Заказ 965

Типография «Новый мир»
340050, Донецк, ул. Артема, 96