

**МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ГРАЖДАНСКОЙ АВИАЦИИ**

Ю.С.Гладышев

ПЕРИФЕРИЙНЫЕ УСТРОЙСТВА

ПОСОБИЕ
для выполнения
лабораторных работ №2,3,4

*для студентов IV курса
специальности 220100
дневного обучения*

Москва-2005

Данные пособия для выполнения лабораторных работ по курсу "Периферийные устройства" издаются в соответствии с учебной программой для студентов 4 курса дневного обучения специальности 220100.

Рассмотрены и одобрены на заседаниях кафедры __ мая 2005г. и методического совета факультета __ мая 2005г.

Лабораторная работа №

Интерфейс последовательной связи

1. Цель работы:

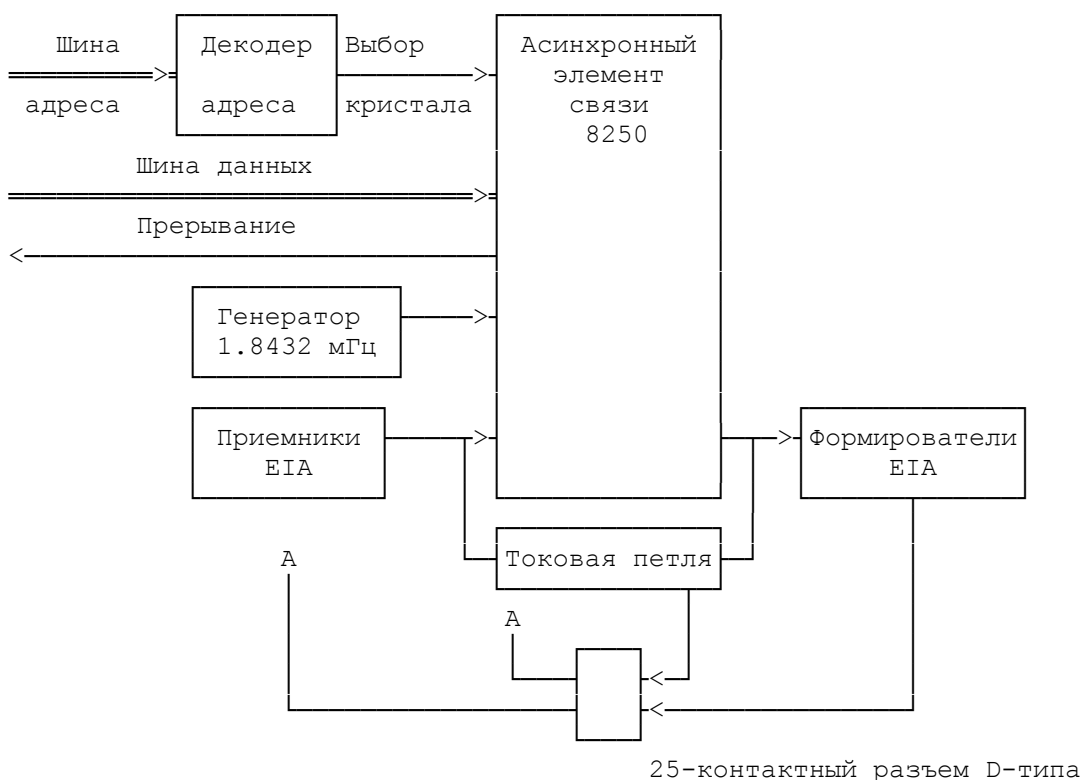
- 1.1. Научиться программировать последовательный порт IBM PC
- 1.2. Изучить временные диаграммы последовательных посылок кодов ASCII литер.
- 1.3. Изучить структуру коммуникационной процедуры

2. Теоретические сведения

Последовательная связь предусматривает последовательную (по битам) передачу данных. Такая связь может осуществляться как между компьютерами, так и между компьютером и другими устройствами. Этот вид связи широко используется для передачи данных между устройствами расположенными в одном помещении, но, если воспользоваться, например, телефонными линиями, такую связь можно осуществлять на любом расстоянии. ЭВМ к телефонной линии подключается с помощью модема. ЭВМ при этом играет роль **ООД** (Оконечного Оборудования Данных) или **DTE** (Data Terminal Equipment), а модем роль **АПД** (Аппаратуры Передачи Данных) или **DCE** (*Data Terminal Equipment*). Подключение АПД к ООД стандартизовано. В настоящее время широко используется американский стандарт RS-232C, которому соответствует наш "СТЫК2" (C2). Максимальная скорость передачи данных согласно стандарту RS-232C составляет 20 КБит/С.

Для формирования и синхронизации строк битов, составляющих последовательные данные, разработаны специальные микросхемы, которые называются «Универсальный Асинхронный Приемник-Передачик» («Universal Asynchronous Receiver Transmitter»). IBM PC использует UART 8250 фирмы Intel.

2.1. Блок схема асинхронного адаптера связи. Архитектура UART 8250. Описание регистров.



Операционная система поддерживает два порта коммуникации COM1 и COM2, базовые адреса которых 3F8H и 2F8H соответственно.

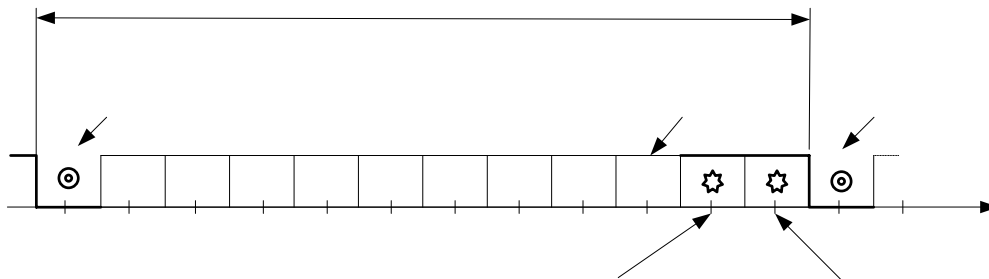
Микросхема UART 8250 имеет 10 программируемых однобайтных регистров, адреса которых даны в табл.1.

2.1.1. Адресация регистров.

Таблица 1

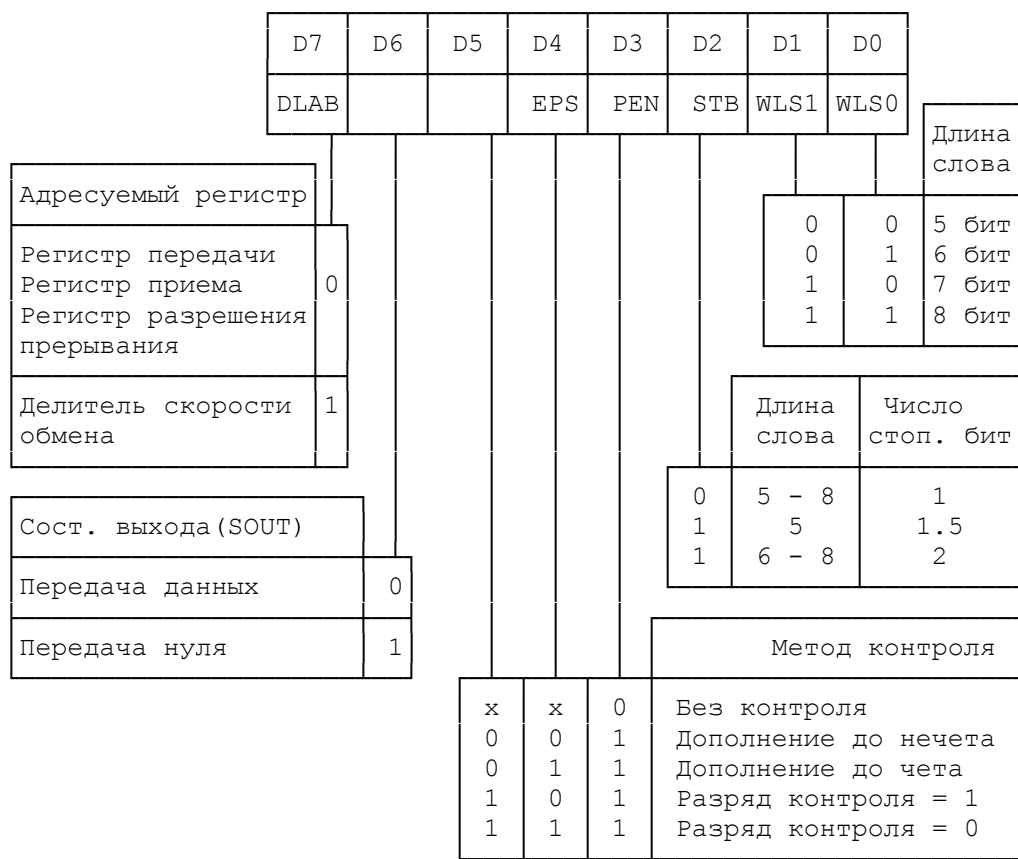
Код адреса регистра		Наименование регистра	DLA В	Сигнал	
COM1	COM2			Чтение IOR	Запись IOW
3F8h	2F8h	Регистр хранения передатчика			
3F8h	2F8h	Регистр данных приемника			
3F8h	2F8h	Делитель скорости обмена (мл.байт)			
3F9h	2F9h	Делитель скорости обмена (ст.байт)			
3F9h	2F9h	Регистр разрешения прерываний "IER"			
3FAh	2FAh	Регистр идентификации прерываний "IIR"			
3FBh	2FBh	Регистр управления линией "LCR"			
3FCh	2FCh	Регистр управления модемом "MCR"			
3FDh	2FDh	Регистр состояния линии LSR			
3FEh	2FEh	Регистр состояния модема MSR			

2.1.2. Формат данных

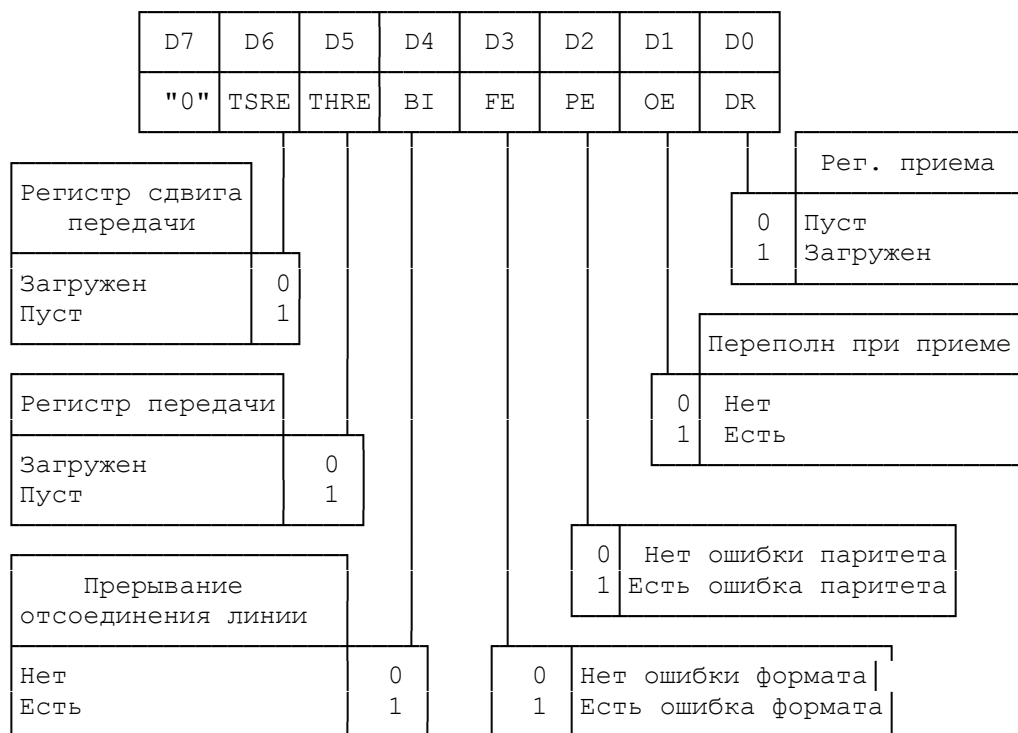


Биты данных D7-D0 . Адаптер автоматически вставляет стартовый бит, бит четности (если это запрограммировано), и стоповые биты(1, 1.5 или 2, в зависимости от команды в регистре управления линией).

2.1.3. Регистр управления линией (LCR).



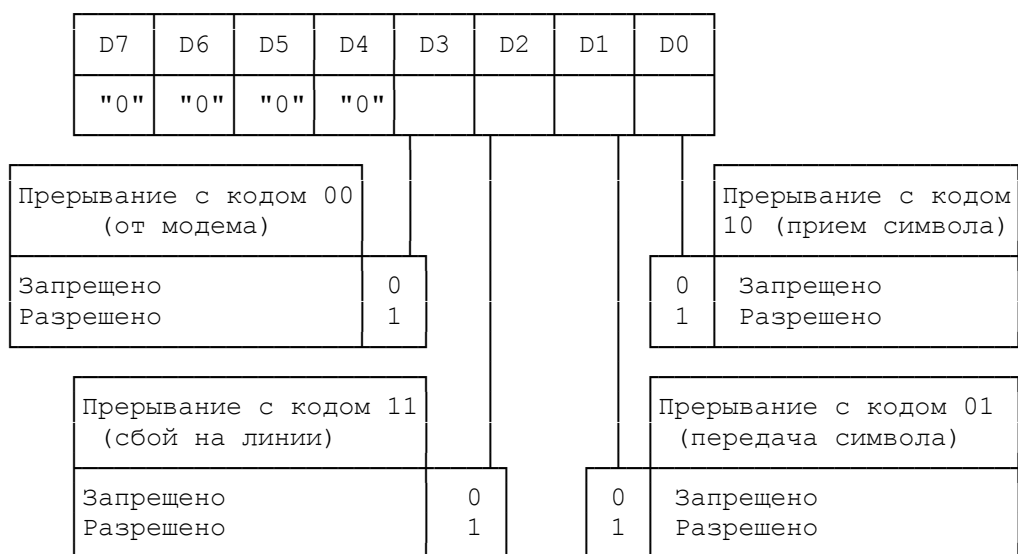
2.1.4. Регистр состояния линии (LSR).



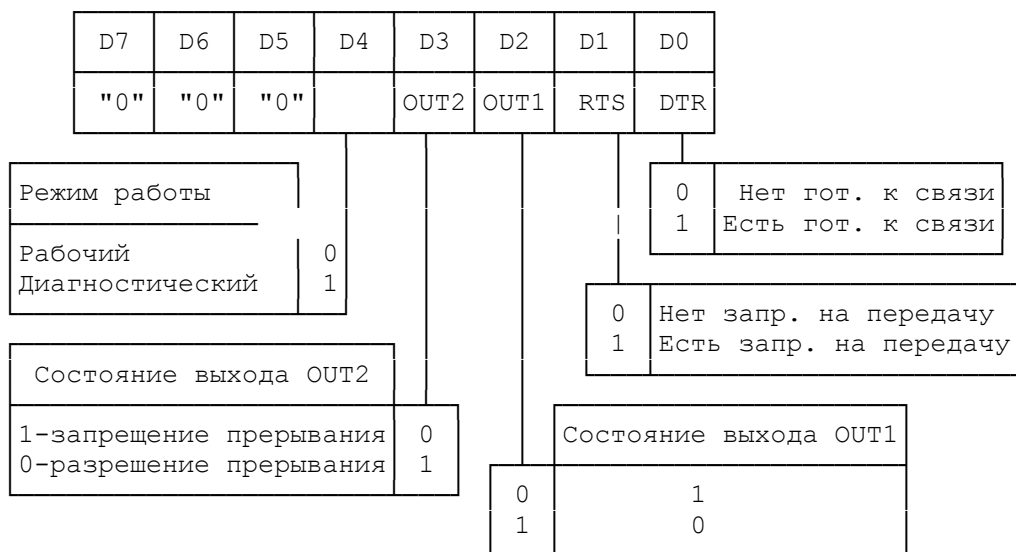
2.1.5. Регистр идентификации прерываний (IRR).



2.1.6. Регистр разрешения прерываний (IER).



2.1.7. Регистр управления модемом (MCR).



Сигналы от компьютера к модему:

DTR (Data Terminal Ready) - готовность компьютера,

RTS (Request To Send) - запрос на посылку.

2.1.8. Регистр состояния модема (MSR).

D7	D6	D5	D4	D3	D2	D1	D0
DCD	RI	DSR	CTS	DCD	RI	DSR	CTS
Состояние сигналов (рабочий режим)				Изменение сигналов			
0-низкий уровень				1-есть изменение			
1-высокий уровень				0-нет			

Сигналы от модема к компьютеру:

DCD (Data Carrier Detect) - обнаружен носитель данных,

RI (Ring Indicator) - индикатор звонка,

DSR (Data Set Ready) - готовность модема,

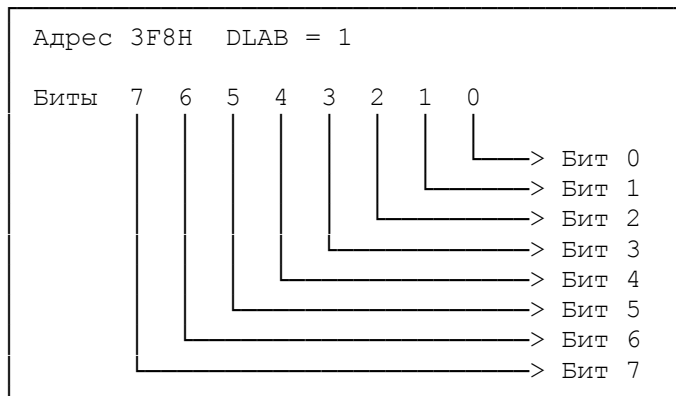
CTS (Clear To Send) - готовность к посылке.

2.1.9. Программируемый генератор скорости передачи.

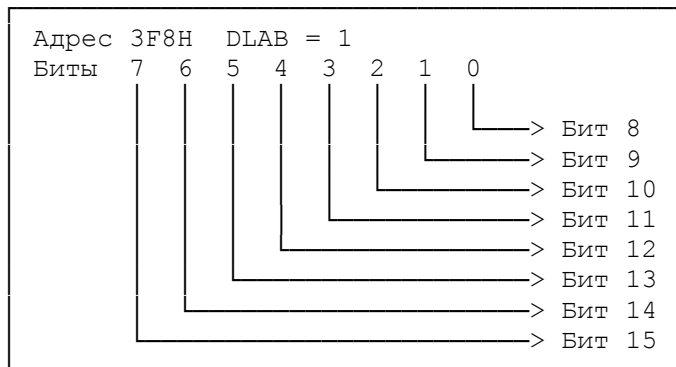
UART 8250 содержит программируемый генератор скорости передачи, который способен делить входной сигнал (1,8432 МГц) на любое число от 1 до (2**16-1). Выходная частота представляет собой скорость передачи умноженную на 16 [делитель = (вхоная частота)/(скорость передачи * 16)]. Делитель хранится в двух 8-битных регистрах в 16-битном двойном формате. Эти регистры делителя должны быть

загружены при инициализации. При загрузке любого регистра из регистров делителя загружается 16-битный счетчик скорости. Это предотвращает долгий пересчет при первоначальной загрузке.

Младшие значащие биты регистра делителя (DLL).



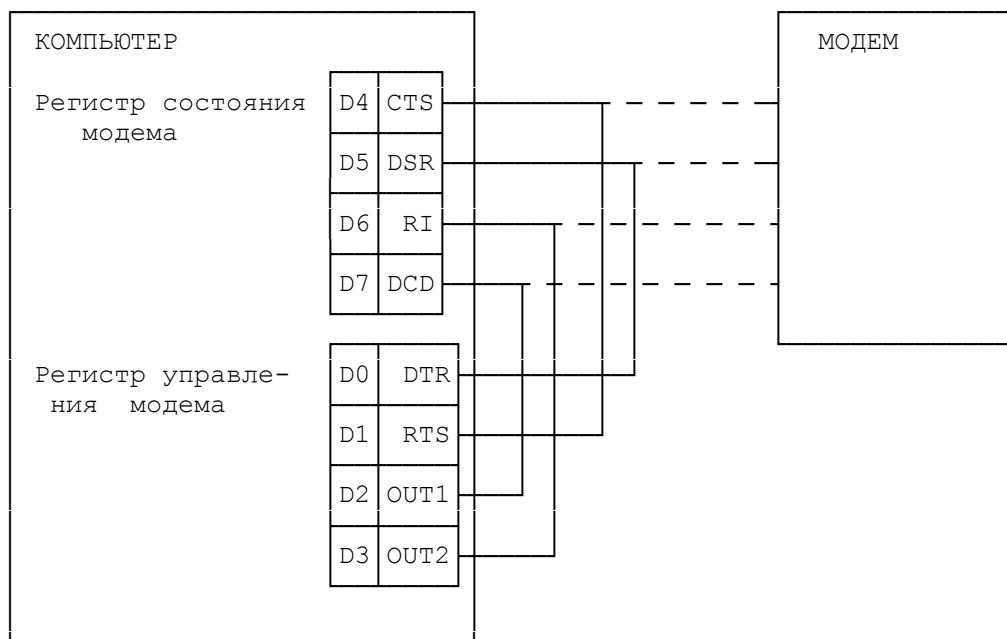
Старшие значащие биты регистра делителя (DLM)



Ниже проиллюстрировано использование генератора скорости передачи с тактовой частотой 1,8432 МГц. Полученная ошибка для скоростей 19200 бод и ниже является минимальной.

Скорость передачи	Делитель используемый для генерации 16*тактовую частоту (десятиричное) (шестнадцатиричное)		Погрешность частоты в %
110	1047	417	0,026
134,5	857	359	0,058
150	768	300	—
300	384	180	—
600	192	0C0	—
1200	96	060	—
1800	64	040	—
2000	58	03A	0,69
2400	48	030	—
3600	32	020	—
4800	24	018	—
7200	16	010	—
9600	12	00C	—
19200	6	006	—

2.1.10. Работа последовательного порта в диагностическом режиме



3. Порядок подготовки к лабораторной работе

- 3.1. Изучить блок-схему программы UART (см. приложение 1) и саму программу на языке Ассемблера (см. приложение 2)
- 3.2. Изучить разъем последовательного порта, назначение его контактов (см. приложение 3.).
- 3.3. Нарисовать последовательную посылку литеры "а", код которой 61h, количество стоповых бит 2, четный паритет.

4. Порядок выполнения лабораторной работы

- 4.1. Соединить контакты разъема и подключить осциллограф так, как показано в приложении 3
- 4.2. Включить ПЭВМ IBM PC и загрузить ОС.
- 4.3. Вызвать редактор текста и с его помощью создать текстовый файл, в который записать программу UART из приложения 2.
- 4.4. Затем выполнить обработку исходного файла:– трансляцию, редактирование связей и запустить полученный модуль на решение.
- 4.5. В результате диалога с ЭВМ ввести в программу параметры регистра режима.
- 4.6. Затем на экране ЭЛТ осциллографа получить устойчивое изображение временной диаграммы последовательной посылки литеры "а".

- 4.7. По известной величине скорости передачи интерфейса и установленной длительности развертки осциллографа рассчитать длину в миллиметрах одного бита посылки.
- 4.8. Нажмите клавишу, соответствующую другой литере и определите ее код. Зарисуйте временную диаграмму и т.д.
- 4.9. Изменить количество стоповых бит и повторите пункт 4.8.
- 4.10. Отменить бит паритета и повторите пункт 4.8.

5. Содержание отчета

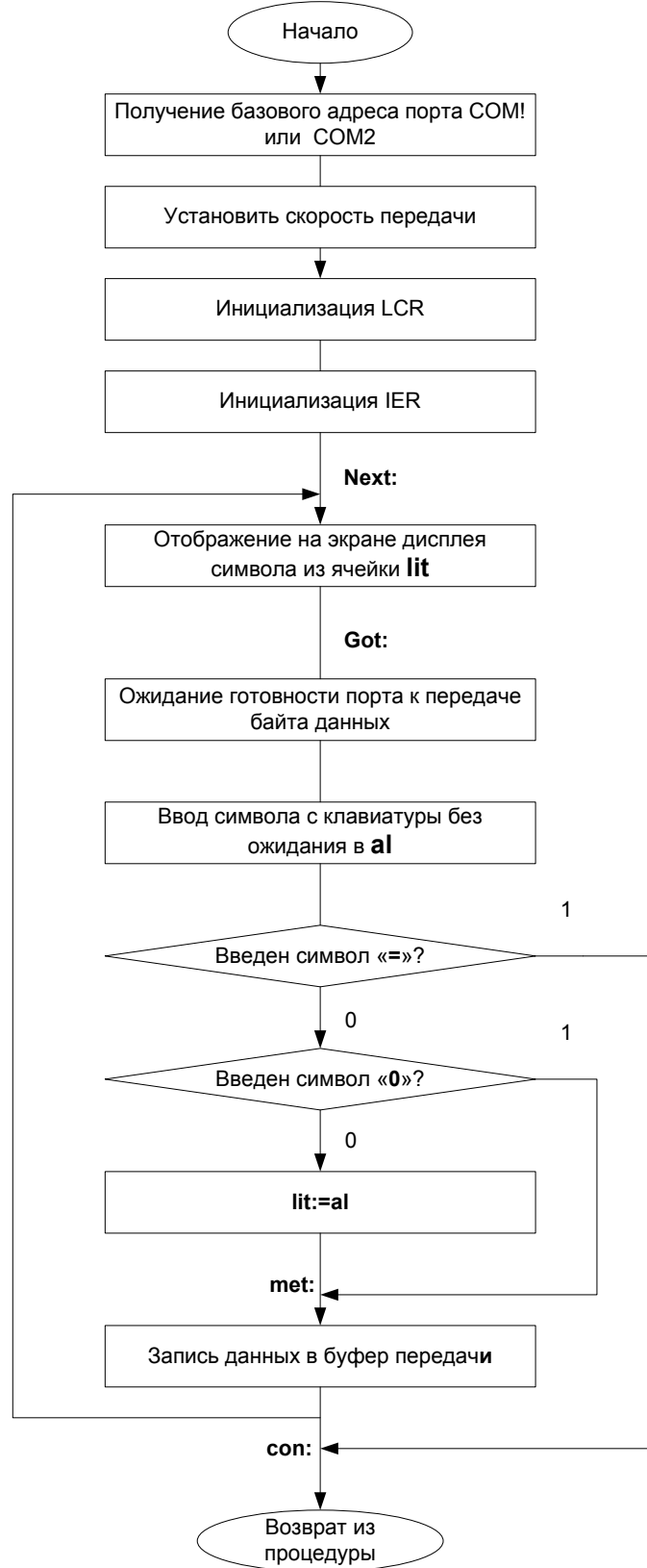
- 5.1. Наименование и цель работы.
- 5.2. Задание, выполненное при подготовке к лабораторной работе.
- 5.3. Задание, выполненное при проведении лабораторной работы.
- 5.4. Выводы по лабораторной работе

6. Контрольные вопросы

- 6.1. Привести временную диаграмму передачи по асинхронной последовательной линии буквы C_{10} , если применяется 7-битный код ASCII, четный паритет и 2 стоповых бита.
- 6.2. Предположим, что промежутков между символами нет, символы содержат 7 информационных бит, имеются бит паритета и 2 стоповых бита, а скорость передачи составляет 600 бит/с. Сколько символов передается в секунду? Сравните ее со скоростью синхронной линии с той же двоичной скоростью 600 бит/с, при отсутствии холостых символов, если сообщение состоит из двух символов синхронизации и последующих 200 7-битных информационных символов.
- 6.3. Каков будет формат передаваемого символа, если регистр LCR содержит код **00011111**?
- 6.4. Определите форматы символа и сообщения последовательной передачи, если в регистре LCR содержится код **00111111**?

7. Приложение 1

7.1. Блок-схема программы UART 8250



8. Приложение 2

8.1. Программа ввода данных с клавиатуры и вывода в последовательный порт: UART 8250

title uart8250

```
stack SEGMENT para stack           ;определение
    DB 8 dup('stack ')             ;сегмента
stack ENDS                          ;стека

data SEGMENT para                  ;определение сегмента данных
    lit DB 'a'
    per DB 13,10,'$'
data ENDS

code SEGMENT
    ASSUME cs:code,ss:stack,ds:data
port PROC far
    push ds                         ;занесение в стек
    push ax                         ;начальных значений
    mov bx,data                     ;для возврата управления
    mov ds,bx

    mov ax,40h                      ;получение базового
    mov es,ax                       ;адреса
    mov dx,es:[2]                   ;порта COM2

;установка скорости обмена
    add dx,3                         ;адресация к LCR
    mov al,80h                       ;DLAB=1,уст-ка ск-ти обмена
    out dx,al
    dec dx                           ;адресуемся к старшему байту
    dec dx                           ;регистра делит. ск-ти обмена
    mov al,01h                       ;скорость обмена=300 бод
    out dx,al
    dec dx                           ;адресуемся к младшему байту
    mov al,80h                       ;устанавливаем младший байт
    out dx,al                       ;регистра делит. для ск-ти 300 бод

;инициализация регистра управления линией
    mov al,3Bh                       ;уст-ка знач.(8 инф,1 ст.бит,
    ;проверка на четность)
    add dx,3                         ;адресуемся к LCR
    out dx,al                       ;посылаем значение в порт

;инициализация регистра разрешения прерываний
```

```

dec dx                ;адресуемся к регистру
dec dx                ;разрешения прерывания IER
mov al,0              ;запрещение прерываний
out dx,al             ;посылаем в порт

mov ah,9              ;перевод строки
lea dx,per
int 21h

next:mov dl,lit       ;вывод на экран содержимого
mov ah,2              ;ячейки lit
int 21h

;установка связи с модемом
mov ax,40h            ;определение базового адреса
mov es,ax             ;порта COM2
mov dx,es:[2]

add dx,4              ;адресация к регистру MCR
mov al,3h             ;назнач. 0-ой,1-ый разряды=1
out dx,al             ;посылаем в порт

add dx,1              ;адресуемся к регистру LSR
got: in al,dx         ;читаем из порта
test al,20h          ;проверяем готовность порта
jz got                ;на метку, если не готов к
;передаче
mov ah,7              ;ввод с клавиатуры с ожиданием
int 21h               ;без отображения

cmp al,'='           ;выход по клавише '='
je con
cmp al,0
je met
mov lit,al

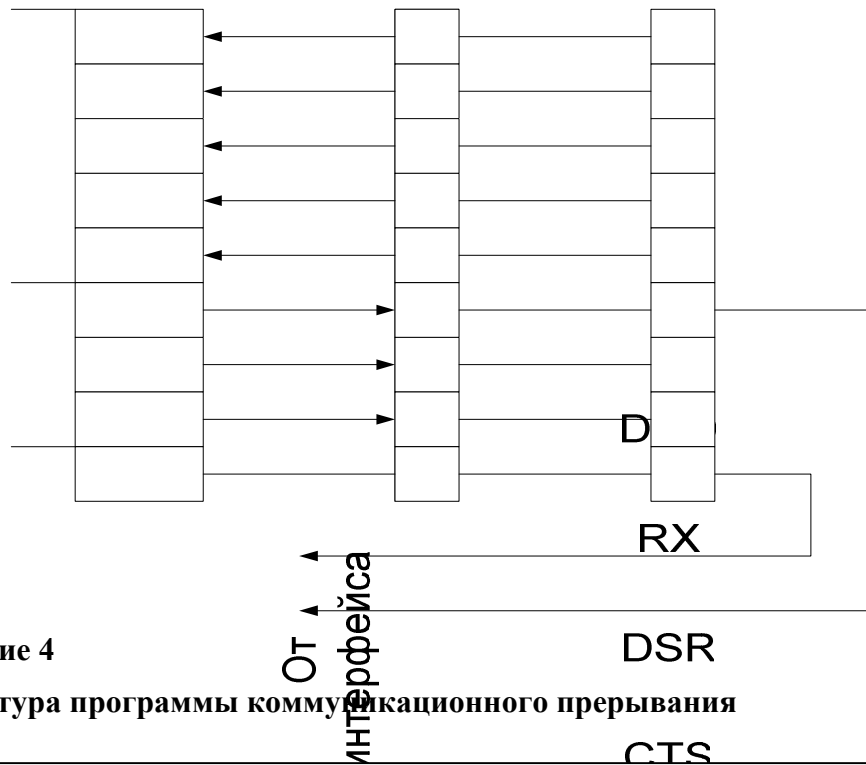
met :mov ax,40h       ;определение базового адреса
mov es,ax             ;порта COM2
mov dx,es:[2]
mov al,lit            ;вывод в порт содержимого
out dx,al             ;ячейки lit
jmp next

con: ret
port ENDP
code ENDS
END port

```

Приложение 3

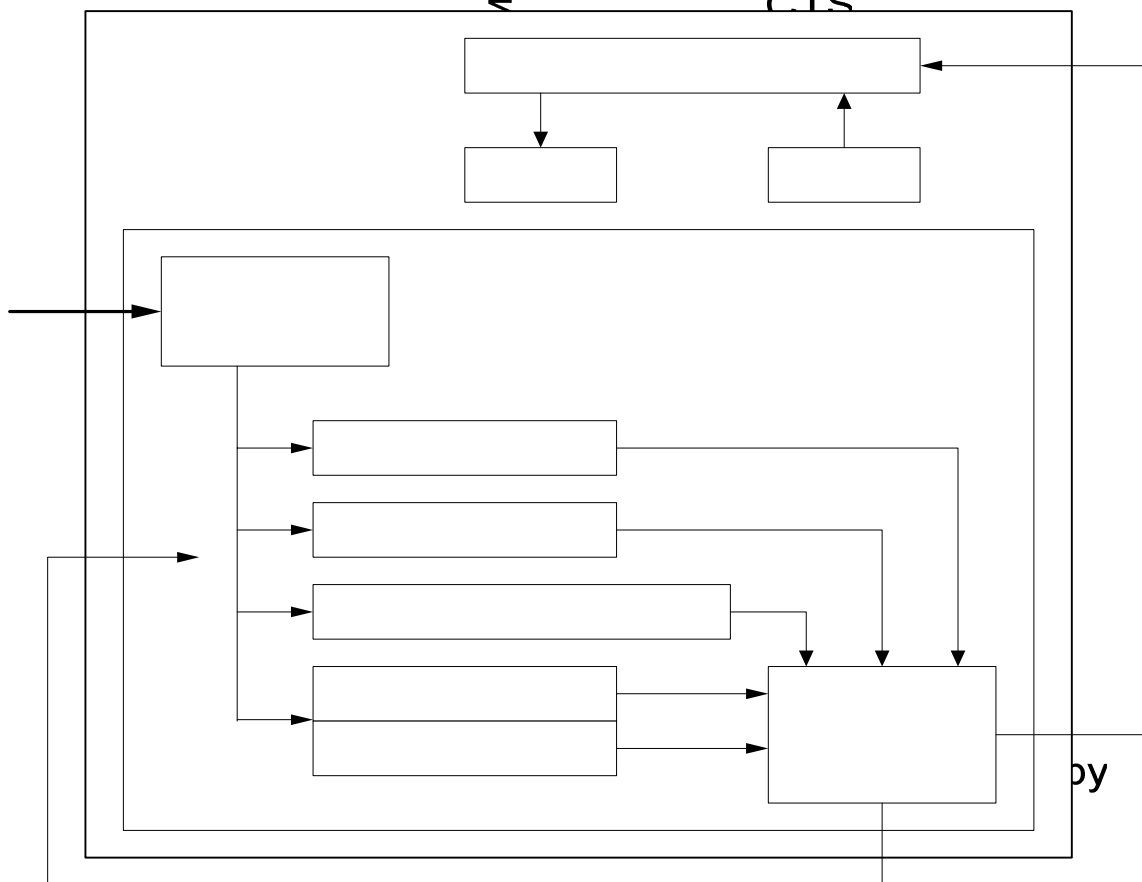
8.2. Обозначение контактов разъема на задней стенке ПЭВМ IBM PC



25-ко
р

9. Приложение 4

9.1. Структура программы коммуникационного прерывания



Литература:

- 1 Техническое описание IBM PC,XT.
- 2 Ю-ЧженЛю, Г.Гибсон "МИКРОПРОЦЕССОРЫ СЕМЕЙСТВА 8086/8088"
М.,Радио и связь, 1987

Лабораторная работа № 3

Устройство графического ввода информации в ЭВМ. планшет (tablet).

1. Цель работы:

- 1.1. Изучить технические характеристики устройства и систему команд.
- 1.2. С помощью команд планшета составить программу, моделирующую работу мыши.
- 1.3. Научиться “связывать” рабочее окно планшета с рабочим окном экрана дисплея.
- 1.4. Научиться вводить в ЭВМ чертежи, размер которых превышает размер рабочего поля планшета.

2. Теоретические сведения

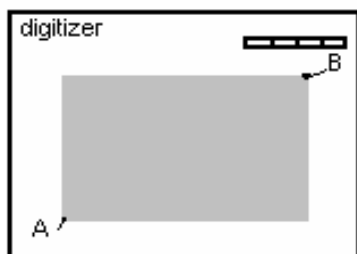
Для ввода в ЭВМ графической информации, например, чертежей, рисунков и т. д. широко используются планшеты (tablets). Поскольку при вводе информации (координат точки) осуществляется их оцифровка, то эти устройства относят к классу дигитайзеров (digitizer), а иногда их так и называют. К классу дигитайзеров, например, относятся широко распространенные в настоящее время мыши, причем мышь можно рассматривать как частный случай дигитайзера. В работе изучается “DIGITIZER KD 4600”, приводятся его технические характеристики, форматы передаваемых данных, система команд GRAPHTEC, отмечается универсальность этого планшета, поскольку он может эмулировать системы команд других фирм. При вводе чертежей, размер которых превышает размеры рабочего поля планшета, устройство следует рассматривать как “окно” на вводимый чертеж, для привязки которого к чертежу в системе команд планшета имеются специальные команды смещения начала координат и поворота осей координат рабочего поля планшета.

2.1. Описание и технические характеристики планшета

2.2. Модель: KD 4600.

- 2.3. Метод кодирования аналоговой информации: магнитная индукция, абсолютные координаты.
- 2.4. Эффективное кодируемое пространство: 460 * 310 мм.
- 2.5. Эффективная кодируемая высота: в пределах 5 мм.
- 2.6. Разрешение: 0.1 мм.
- 2.7. Точность данных: +/- 0.5 мм.
- 2.8. Скорость кодирования информации:
- 2.8.1. мах 60 позиций в сек. в ASCII формате;
- 2.8.2. мах 150 позиций в сек. в двоичном формате.

DIGITIZER KD4600



A - начало координат (0,0) ;
 B - значение координат (4600,3100)
 Эффективное пространство -
 460 на 310
 Разрешение -0.1 мм/0.005 дюйм

ПЕРЕКЛЮЧАТЕЛИ

SW1



SW2



SW3



Указанное положение переключателей устанавливает: параметры передачи **9600,7,N,2** с ограничителем **CR+CF**, формат передаваемых данных **ASCII**, единицы измерения **MM**, включенный динамик, систему команд **GRAPHTEC**.

Для съема информации используются курсор (cursor) или перо (stylus).

Курсор содержит:

1. ПРИЦЕЛ (crosshair)

Планшет считывает координаты точки, находящейся в центре прицела.

2. КНОПКИ ФЛАГОВ (flag batton)

Четыре кнопки курсора (Z, 1, 2, 3) обеспечивают установку флагов . В POINT (точечный) и SWITCH STREAM (переключения потоков) режимах координаты считываются во время нажатия одной из этих кнопок .

Перо содержит:

1. КОНЧИК ПЕРА (stylus tip)

Планшет считывает координаты точки на планшете с которой контактирует кончик пера . Легким нажатием устанавливается внутренний переключатель , использующийся для ввода координат в режимах POINT и SWITCH STREAM . Этот переключатель выполняет те же самые функции , что и кнопка Z курсора .

2. ПЕРЕКЛЮЧАТЕЛЬ

Кнопка переключателя соответствует кнопке курсора 1. При нажатии этой кнопки точка, отображенная посредством пера , считывается как установка координат .

2.9. Структурная схема планшета

Дигитайзер типа KD 4600, используемый в ПАК в качестве полуавтоматического устройства ввода графической информации, обладает всеми типичными для такого рода устройств признаками. Но наряду с этим он имеет несколько особенностей, главная из которых заключается в том, что источником пульсирующего электромагнитного поля являются не возбуждённые координатные шины X и Y, а катушка индуктивности курсора, в которую подаётся пульсирующий ток от тактового генератора. Токи, возникающие в координатных шинах за счёт взаимодействия проводника шины с магнитным полем, создаваемым катушкой курсора, пропорциональны расстоянию от центра курсора до проводника координатной шины. Это справедливо как в направлении координаты X, так и в направлении координаты Y. Фиксируя, таким образом, амплитуды сигналов, считанных с координатных шин и анализируя распределение этих амплитуд вдоль каждой их координат, можно с высокой точностью определять положение центра курсора на рабочем поле дигитайзера.

Структурная схема дигитайзера типа KD 4600 приведена на рисунке 3. В схеме опущены некоторые элементы, способствующие быстрой и точной регистрации координат, но не влияющие на понимание принципов работы дигитайзера.

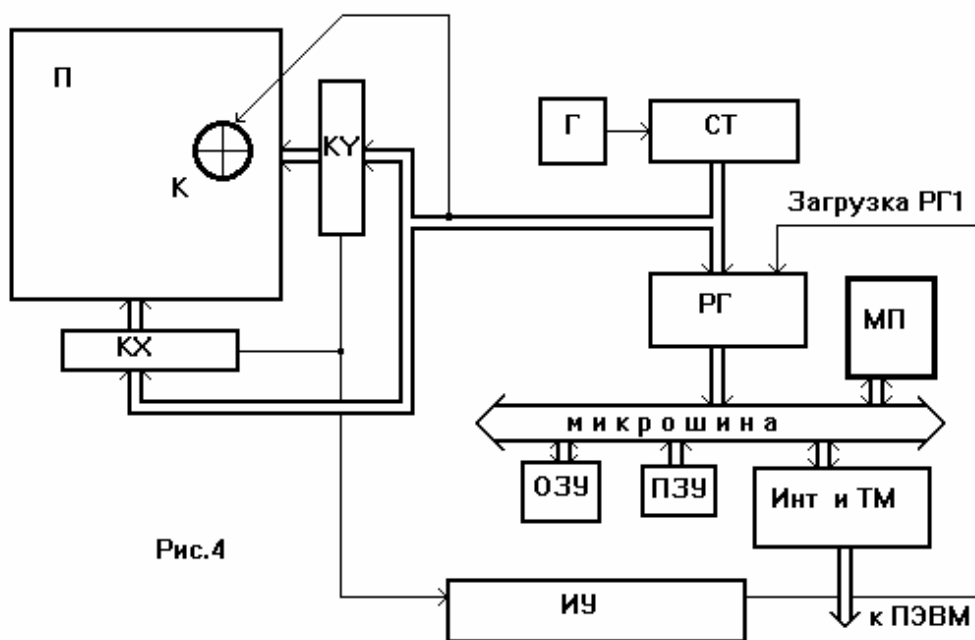


Рис.4

Рисунок 3. Структурная схема планшета.

В состав дигитайзера входят следующие основные узлы и устройства:

- планшет (П) - квадратная матрица с вмонтированной в неё системой координатных шин;
- курсор (К) - катушка индуктивности с визирным перекрестием в геометрическом центре катушки;
- генератор (Г) сигналов тактовой частоты 9 мГц;
- счётчик (СТ), работающий в пересчётном режиме;
- коммутаторы координатных шин (КХ и КУ) по осям X и Y;
- микропроцессор (МП), оперативное запоминающее устройств (ОЗУ) и постоянное запоминающее устройство (ПЗУ), подключённые к аппаратуре дигитайзера с помощью микрошины;
- измерительное устройство (ИУ) для измерения и фиксации амплитуд сигналов, считанных с координатных шин;
- интерфейсный блок (Инт) и таймер (ТМ).

Принцип работы дигитайзера кратко заключается в следующем. При установке визира на кодируемую точку чертежа (рисунка), неподвижно закреплённого на рабочем поле дигитайзера, и нажатии кнопки на корпусе визира сигналы с генератора (Г) начинают поступать на счётчик (СТ), управляющий очередностью возбуждения координатных шин через коммутатор (КХ). Сигнал со счётчика поступает также на катушку индуктивности курсора. Таким образом, координатные шины

возбуждаются коммутатором КХ синхронно с поступлением сигнала в катушку индуктивности курсора. Токи, возникающие в координатных шинах по законам взаимной индукции, по величине и направлению пропорциональны расстоянию координатной шины от центра курсора и взаимному расположению шины и центра катушки курсора (слева или справа). С выхода коммутатора КХ напряжения, пропорциональные токам, протекающим по шинам, поступают в измерительное устройство (ИУ), в котором эти напряжения последовательно запоминаются на схемах выборки-хранения, а затем фильтруются (сглаживаются) и подаются на компаратор (схему сравнения). Сигнал, достигший максимума, переключает компаратор, который, в свою очередь, ставит RS-триггер в измерительном устройстве в “1”, а следующий за максимальным нулевой сигнал в “0”. Тем самым фиксируется координата X на поле планшета, над которой находится центр курсора. Сигнал с выхода RS-триггера измерительного устройства поступает на регистр (РГ), разрешая запись содержимого счётчика (СТ) в регистр. В регистр, таким образом, заносится код координаты (X) центра курсора. После этого микропроцессор (МП) обрабатывает содержимое регистра, заносит его в ОЗУ и через интерфейс передаёт код координаты X в ПЭВМ. На этом заканчивается фаза определения координаты X центра курсора.

Определение координаты Y производится аналогично в следующей фазе работы дигитайзера после сброса всех узлов схемы в исходное состояние.

2.10. Схема алгоритма главной программы устройства

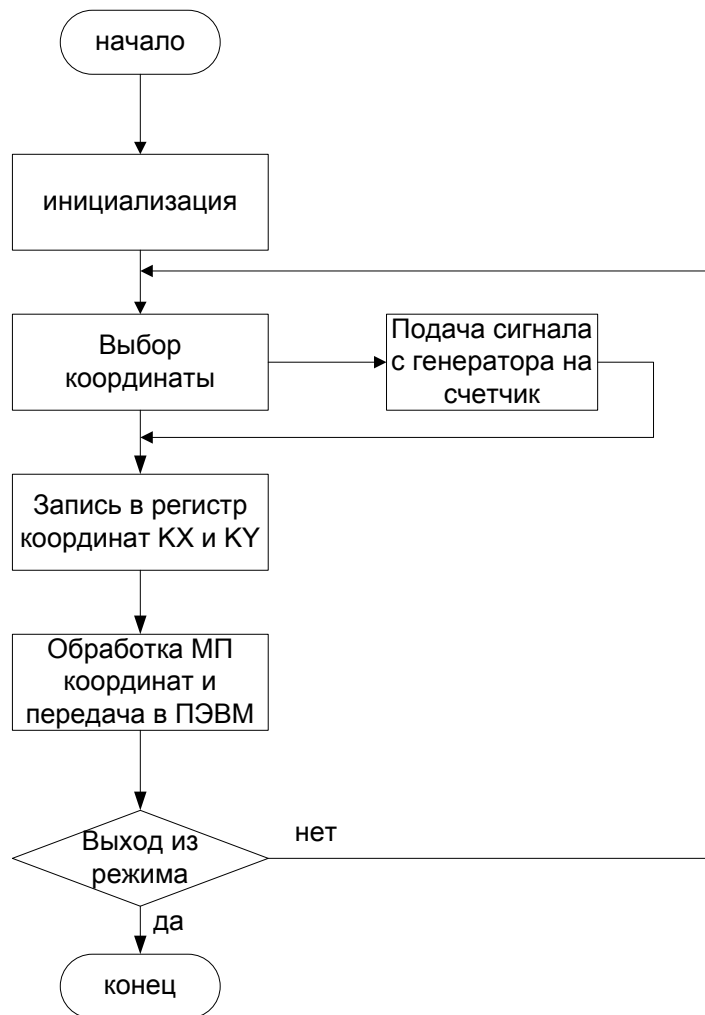


Рисунок 4. Алгоритм главной программы устройства.

2.11. Формат выводимой информации

Выбор формата вывода устанавливается позицией 1 на SW1.

ASCII ФОРМАТ.

ASCII формат 1:

XXXX,YYYY,F CR LF

↙ ↓ ↘ →

значения по значения по идентификатор терминатор
горизонтальной вертикальной переключения
оси оси

X и Y символы ASCII (0-9). Когда значение координаты меньше, чем 4 цифры, выводится 0. F - флаг для идентификатора переключателя. Значения флага, как показано ниже, выводятся согласно нажатому переключателю :

для курсора:

0 - никакой переключатель не нажат

1 - нажата кнопка Z

2 - нажата кнопка 1

4 - нажата кнопка 2

8 - нажата кнопка 3

для пера :

1 - нажата кнопка Z

2 - нажата кнопка 1 ;

(CR) (LF) - ограничители строки. Вместе CR(возврат каретки) и LF(перевод строки) или только LF могут быть выбраны как терминаторы , в зависимости от позиции 2 SW1 ;

ASCII формат 2 :

В режимах изменения и относительном используется следующий формат вывода :

± XXXX, ± YYYYY,F CR LF

Знак плюса или минуса прикладывается к горизонтальному или вертикальному значениям координат.

ДВОИЧНЫЙ ФОРМАТ.

Двоичный формат 1 :

В двоичном формате данные выводятся в пятибайтовом режиме, как показано ниже:

бит байт	7	6	5	4	3	2	1	0
1	0	1	F3	F2	F1	F0	XA	YA
2	0	0	X5	X4	X3	X2	X1	X0
3	0	0	X11	X10	X9	X8	X7	X6
4	0	0	Y5	Y4	Y3	Y2	Y1	Y0
5	0	0	Y11	Y10	Y9	Y8	Y7	Y6

0: всегда 0;

1: всегда 1;

X0-X11, Y0-Y11 : выражают координаты вдоль осей X и Y в двоичной системе счисления ;

F0-F3: соответствуют кнопкам Z,1,2 и 3 на курсоре. Все равны 1, если соответствующая кнопка нажата и 0 - если отключена. При работе с пером используются только кнопки F0,F1 (Z= F0, 1=F1, 2=F2, 3=F3);

XA, YA: всегда 0, за исключением режимов изменения и относительном:

XA: 0, если X - положительное или 0;

1, если X - отрицательное;

YA: 0, если Y - положительное или 0;

1, если Y - отрицательное;

X0-X11 и Y0-Y11 выводятся как абсолютные значения.

Двоичный формат 2.

В двоичном формате 2 данные выводятся в восьмибайтовом формате, как показано ниже :

бит байт	7	6	5	4	3	2	1	0
1	0	PH	0	0	0	0	T	PR
2	0	0	0	F4	F3	F2	F1	F0
3	0	0	X5	X4	X3	X2	X1	X0
4	0	0	X11	X10	X9	X8	X7	X6
5	0	0	0	X16	X15	X14	X13	Y6
6	0	0	Y5	Y4	Y3	Y2	Y1	Y0
7	0	0	Y11	Y10	Y9	Y8	Y7	Y6
8	0	0	0	Y16	Y15	Y14	Y13	Y12

PH-фазовый бит, всегда 1. T-флаг дигитайзера, всегда 1. PR-ближайший бит, 0 - показывает связь внутри эффективного пространства, 1 - показывает связь за пределами эффективного пространства .

F4-F0-биты флагов. F4 всегда 0. F0-F3 соответствуют кнопкам Z,1,2 и 3 на курсоре. Все равны 1, если соответствующая кнопка нажата и 0 - если отключена. При работе с пером используются только кнопки F0,F1 (Z= F0, 1=F1, 2=F2, 3=F3);

X0-X16, Y0-Y16 - значения координат X и Y .

2.12. Совместимость с дигитайзерами других фирм

DIP - переключатель SW3 может быть использован для эмуляции планшетов, изготавливаемых другими фирмами производителями. Различия прежде всего касаются команд (приказов) которые поддерживаются соответствующими планшетами.

2.12.1. Изучаемый планшет помимо “своих” команд поддерживает команды:

- 2.12.1.1. DT 4100 Commands
- 2.12.1.2. HDG Commands
- 2.12.1.3. Bit Pad Two Commands
- 2.12.1.4. MM Series Commands

При использовании данного планшета с программой Автокад, его нужно настроить на MM Series Commands, на которые сам Автокад может быть настроен.

2.13. Система команд graphtec

T COMMAND (TEST)

Функция : После получения команды T для каждого байта , полученного планшетом , идентичный байт возвращается в компьютер .

Код команды : T (54H)

Пример : PRINT #1,"T";

Описание : Эта команда используется для проверки данных , которые были посланы и получены интерфейсом . Режим проверки длится от момента приема команды T до того момента , когда планшет выключают или нажимают кнопку RESET . Для реализации этой проверки планшет должен находиться в режиме POINT . Перед выполнением команды T необходимо нажать кнопку RESET .

Пример программы :

```
50 CLS
100 OPEN "COM1:9600,N,7,2,CS,DS" FOR RANDOM AS #1
110 PRINT #1, "T";
120 PRINT "*** HIT KEY ***"
130 PRINT "E=====> END"
140 A$ = INKEY$
150 IF A$ = "" THEN 140
160 IF A$ = "E" THEN 220
170 PRINT "OUTPUT CHAR= "; A$
```



```

180 PRINT #1, A$
190 LINE INPUT #1, B$
200 PRINT "INPUT CHAR= "; B$
210 IF A$ = B$ THEN 120 ELSE PRINT "TEST *** NG ***"
220 CLOSE #1
230 END

```

U/V COMMANDS (STATUS LAMP ON/OFF CONTROL)

Функция : после получения команды U STATUS LAMP загорается , после получения команды V , STATUS LAMP гаснет .

Код команды : U (55H) ,
V (56H) .

Пример : PRINT #1,"U";
PRINT #1,"V";

Описание : эта команда контролирует STATUS LAMP на планшете . Она используется для проверки лампочки .
Пример программы :

```

100 OPEN "COM1:9600,N,7,2,CS,DS" AS#1
110 PRINT "U KEY -----> STATUS LAMP ON"
120 PRINT "V KEY -----> STATUS LAMP OFF"
130 PRINT "E KEY -----> PROGRAM END"
150 A$=INKEY$
160 IF A$="" THEN 150
170 IF A$="E" THEN 210
180 IF A$<>"U" AND A$<>"V" THEN 150
190 PRINT #1,A$
200 GOTO 150
210 CLOSE #1
220 END

```

Z COMMAND (BEEPER CONTROL)

Функция : Команда озвучивает гудок планшета .

Код команды : Z(5AH)

Пример : PRINT #1,"Z";

Описание : Команда озвучивает гудок планшета в течении 40 мс. Она обычно используется для тестирования гудка или быстрого ввода оператора.

Пример программы :
100 CLS

```

110 OPEN "COM1:9600,N,7,2,CS,DS" FOR RANDOM AS #1
120 PRINT "B KEY -----> BUZZER ON"
130 PRINT "E KEY -----> PROGRAM END"
140 A$ = INKEY$
150 IF A$ = "E" THEN 190
160 IF A$ = "" AND A$ <> "B" THEN 140
170 IF A$ = "B" THEN PRINT #1, "Z";
180 GOTO 140
190 CLOSE #1
200 END

```

S COMMAND (DATA OUTPUT INHIBIT)

Функция : Эта команда запрещает вывод данных с планшета .

Код команды : S(53H)

Пример : PRINT #1,"S";

Описание : Команда используется в режиме STREAM для остановки потока выводимых из планшета данных . В режиме POINT и др . ,в которых данные выводятся при нажатом курсоре , эта команда может быть использована для прекращения вывода , если кнопка нажата невовремя .

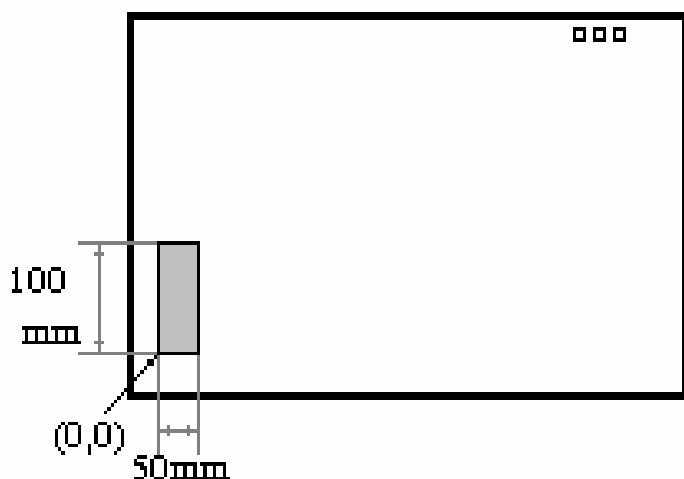
После использования команды S вывод данных разрешается через 0.5 секунд , затем посылаются команды режимов POINT , STREAM, SWITCH STREAM .

Пример программы :

```

100 CLS
110 OPEN "COM1:9600,N,7,2,CS,DS" FOR RANDOM AS #1
120 PRINT #1, "J";
130 INPUT #1, X, Y, F
140 PRINT X; Y; F
150 IF F <> 8 THEN 160 ELSE 200
160 IF X <= 500 AND Y <= 1000 THEN PRINT #1,"S"; : ELSE 130
170 PRINT #1, "Z";
180 A$ = INKEY$
190 IF A$ <> "R" THEN 180 ELSE 120
200 CLOSE #1
210 END

```



P COMMAND (POINT MODE)

Функция : выбирает режим POINT .

Код команды : P(50H)

Пример : PRINT #1,"P" ;

Описание : когда эта команда послана планшету , возможен режим POINT . В POINT режиме , когда кнопка курсора или пера нажата , значение координат X,Y и номер флага выводятся из планшета на компьютер .Режим POINT эффективен для введения диаграмм , включающих большое число прямых линий , или для вызова подпрограмм номером флага .

Пример программы :

```
100 CLS
```

```
110 OPEN "COM1:9600,N,7,2,CS,DS" AS#1
```

```
120 PRINT #1,"P";
```

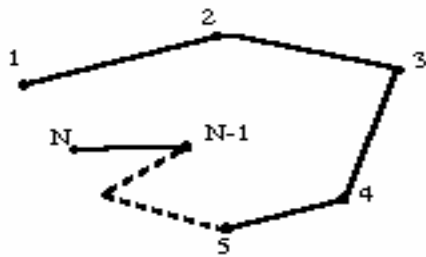
```
130 INPUT #1,X,Y,F
```

```
140 PRINT X;Y;F
```

```
150 IF F<>8 THEN 130
```

```
200 CLOSE #1
```

```
210 END
```



Y COMMAND (TRIGGER MODE)

Функция : Вывод одного установленного значения данных в беззнаковом формате.

Код команды : Y(59H)

Пример : PRINT #1,"Y";

Описание : При получении команды Y выводятся текущие координаты и значение флага, независимо от положения курсора, STATUS LAMP и пера.

Если курсор или перо вышли за пределы эффективного пространства планшета, выводится значение 9999, 9999 в ASCII формате или значение 4095, 4095 в двоичном формате 1. При двоичном формате 2 данные выводятся соответственно установленному ранее эффективному пространству планшета.

Пример программы :

```

100 CLS
110 OPEN "COM1:9600,N,7,2,CS,DS" FOR RANDOM AS #1
120 PRINT #1, "Y";
130 INPUT #1,X,Y,F
140 PRINT X; Y; F
150 IF F <> 8 THEN 120
200 CLOSE #1
210 END

```

A~G COMMAND (SWITCH STREAM MODE)

Функция : команды выбирают режим SWITCH STREAM .

Коды команд : @(40H), A(41H), B(42H), C(43H),
D(44H), E(45H), F(46H), G(47H)

Пример : PRINT #1,"@";

Описание : В данном режиме при отжатой кнопке курсора значения координат X и Y и флага выводятся непрерывно. При использовании пера, как и в режимах STREAM и PIONT, значение флага равно 1 или 2. Скорость вывода данных зависит от команд.

Команда	<i>@</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>
Скорость (данн./с)	2	4	10	20	40	60	60	60

Пример программы :

```

100 CLS
110 OPEN "COM1:9600,N,7,2,CS,DS" FOR RANDOM AS #1
120 PRINT "COMMAND (@,A-G) "
130 PRINT "*** HIT KEY ***"
140 A$ = INKEY$
150 IF A$ = "" THEN 140
160 PRINT #1, A$
170 INPUT #1, X, Y, F
180 PRINT X; Y; F
190 IF F <> 8 THEN 170
200 CLOSE #1
210 END

```

H~O COMMAND (STREAM MODE)

Функция : Выбор режима STREAM.

Коды команд : H(48H), I(49H), J(4AH), K(4BH),
L(4CH), M(4DH), N(4EH), O(4FH)

Пример : PRINT #1,"H";

Описание : При получении одной из этих команд планшетом начинают непрерывно выводиться значения координат X и Y и флага, независимо от курсора и стайлоса (on/off). Когда кнопка курсора не нажата, значение флага - 0, в остальных случаях значение флага соответствует нажатой кнопке, если используется стайлос, значение флага равно 1 или 2. Скорость вывода данных зависит от команд.

Команда	<i>H</i>	<i>I</i>	<i>J</i>	<i>K</i>	<i>L</i>	<i>M</i>	<i>N</i>	<i>O</i>
Скорость (данн./с)	2	4	10	20	40	60	60	60

Пример программы :

```

100 CLS
110 OPEN "COM1:9600,N,7,2,CS,DS" FOR RANDOM AS #1
120 PRINT "COMMAND (H-O) "
130 PRINT "*** HIT KEY ***"

```

```

140 A$ = INKEY$
150 IF A$ = "" THEN 140
160 PRINT #1, A$
170 INPUT #1, X, Y, F
180 PRINT X; Y; F
190 IF F <> 8 THEN 170
200 CLOSE #1
210 END

```

a COMMAND (RELATIV MODE)
b COMMAND (RELATIV MODE RESET)

Функция : Установка и переустановка относительного координатного режима.

Код команды : a(61H)
b(62H)

Пример : PRINT #1,"a";
PRINT #1,"b";

Описание : При получении команды "a" планшет входит в режим RELATIV (относительный). Координаты, приобретенные в режимах POINT, STREAM, SWICH STREAM и TRIGGER выводятся как относительные значения до тех пор, пока планшет не выключен, не нажата кнопка RESET или не получена команда "b".

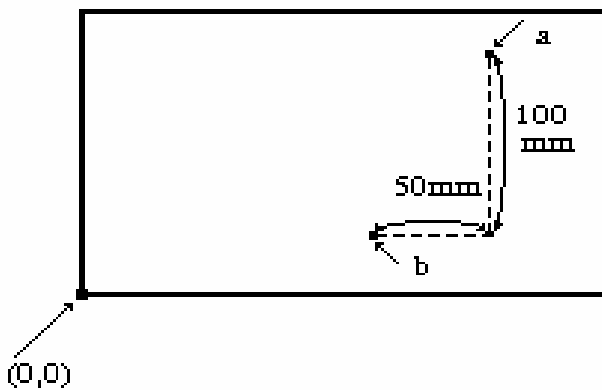
В режиме вывода относительных координат данные, полученные из планшета, отображают расстояние от предыдущей точки. Если текущая точка левее и ниже предыдущей, то координаты X и Y - отрицательные.

Пример программы :

```

100 CLS
110 OPEN "COM1:9600,N,7,2,CS,DS" FOR RANDOM AS #1
120 PRINT #1, "P";
130 PRINT #1, "a";
140 INPUT #1, X, Y, F
150 PRINT X; Y; F
160 IF F <> 8 THEN 140
170 PRINT #1, "b";
200 CLOSE #1
210 END

```



c **COMMAND (ORIGIN CHANGE MODE)**
d **COMMAND (ORIGIN RESET)**

Функция : Команды изменяют и переустанавливают начало координат .

Код команды : cX, Y CR LF
d(64H)

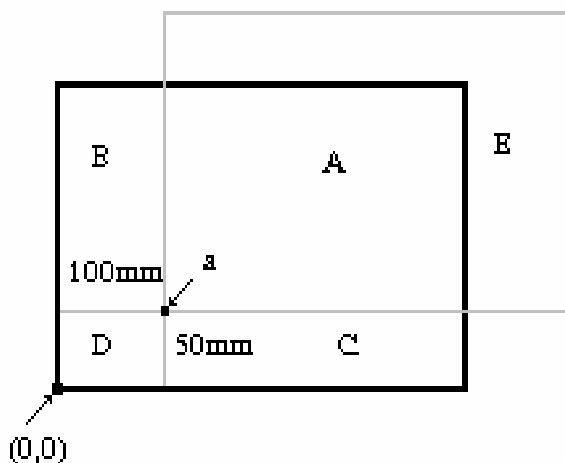
Пример : PRINT #1, "cX,Y"; CHR\$(&HA); CHR\$(&HD);
PRINT #1, "d";

Описание : Начало координат может быть установлено в любой точке эффективного пространства. Команды cX, Y CR LF устанавливают начало координат в X и Y, где X и Y абсолютные координаты.

Начало координат остается в X и Y, пока не получена команда d, прибор не выключен или не нажата кнопка RESET.

Пример программы :

```
100 CLS
110 OPEN "COM1:9600,N,7,2,CS,DS" AS#1
120 PRINT #1,"P";
130 PRINT #1,"c1000,500"; CHR$(&HA); CHR$(&HD);
150 INPUT #1,X,Y,F
160 PRINT X;Y;F
170 IF F<>8 THEN 150
180 PRINT #1,"d";
190 CLOSE #1
200 END
```



R COMMAND (All Mode)

Wd COMMAND (Increment Mode)

Функция : R выбирает все режимы (All Mode).

Wd устанавливает режим Increment Mode

Код команды : R(52H)

Wd(57H)

Пример : PRINT #1, "R";

PRINT #1, "W50"; CHR\$(&HA); CHR\$(&HD);

Описание : Эти команды выбирают условия при выводе данных.

В режиме ALL все координатные значения поступают на выход с разрешением 0.1 мм или 0.005 дюйма. Дигитайзер в этом режиме работает при включении питания или после RESET. В INCREMENT MODE значения координат с выхода дигитайзера поступают только в том случае, если текущая точка находится на расстоянии d , от предыдущей координаты. Если при этом будет включен режим POINT MODE, то в этом случае координаты будут выдаваться в момент нажатия любой клавиши независимо от расстояния текущей точки от предыдущей.

Пример программы INC.BAS:

```
100 CLS
```

```
110 OPEN "COM1:9600,N,7,2,CS,DS" FOR RANDOM AS #1
```

```
120 PRINT #1, "K";
```

```
130 PRINT #1, "W50"; CHR$(&HD); CHR$(&HA)
```

```
140 INPUT #1, X, Y, F
```

```
150 PRINT X; Y; F
```

```
160 IF F <> 8 THEN 140
```

```
170 PRINT #1, "R";
```

```
180 CLOSE #1
```


190 END

e **COMMAND** (*Angle Compensation Mode*)

f **COMMAND** (*Angle Compensation Reset*)

Функция : Эти команды выполняют угловую компенсацию осей координат (поворот осей координат рабочего поля планшета).

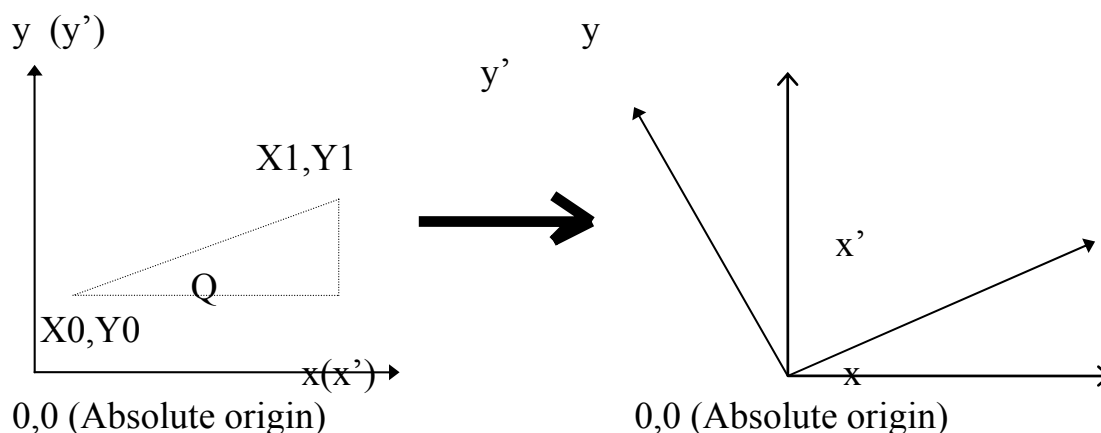
Код команды : e X0,Y0,X1,Y1,CR,LF

f(66H)

Пример : PRINT #1, "e X0,Y0,X1,Y1"; CHR\$(&HA); CHR\$(&HD)
PRINT #1, "f"

Действие команды демонстрирует следующий рисунок

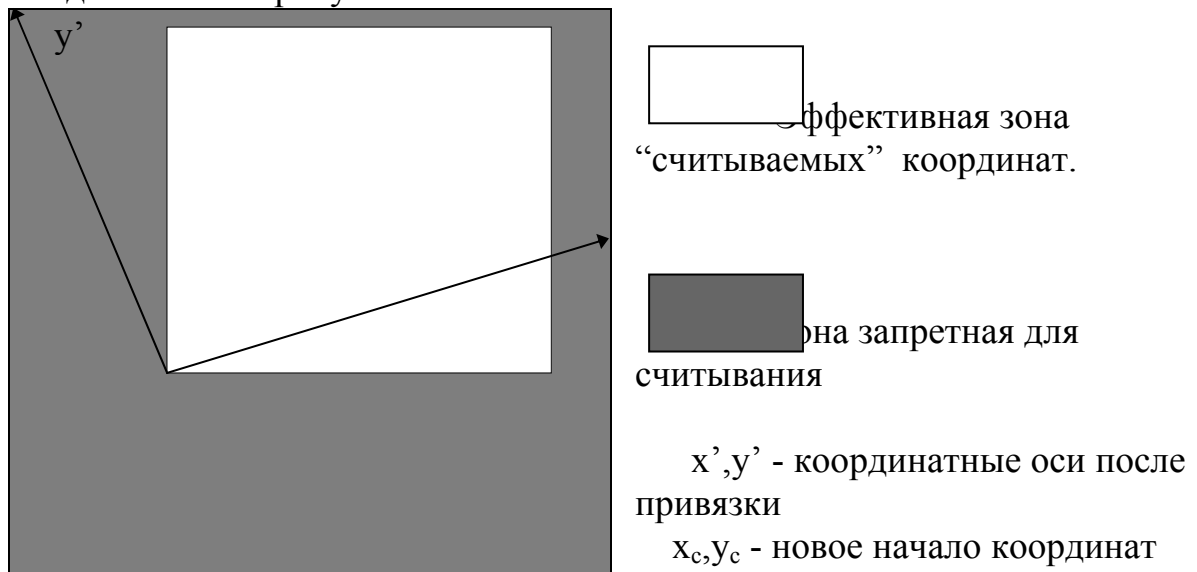
**Соотношение координатных осей перед и после угловой компенсации
(поворот осей x',y' относительно осей x,y)**



где $X0, Y0, X1, Y1$ - десятичные числа задающие вектор, параллельно которому после выполнения команды e X0,Y0,X1,Y1,CR,LF будет проходить абсцисса рабочего поля планшета Ox' , причем допустимый угол поворота осей (Q) лежит в пределах 0 - 90 градусов.

Режим угловой компенсации (Angle Compensation Mode) может быть использован в комбинации с режимом установки начала координат планшета (Origin Change Mode) для осуществления привязки системы координат планшета к реперной точке, заданной в системе координат чертежа. Указанные преобразования координат имеют место при вводе в ЭВМ больших чертежей размер которых превышает рабочее поле планшета.

Следует иметь в виду, что такие преобразования уменьшают эффективную зону “снимаемых” координат. Это эффект отражает приводимый ниже рисунок.



Пример программы anglcomp.bas:

```

100 CLS : SCREEN 1, 0
110 MXT = 4600: MYT = 3100: MXS = 319: MYS = 199
120 A = MXS / MXT: B = MYS / MYT
130 IF A > B THEN MTS = B ELSE MTS = A
140 LINE (0, 199)-(319, 0), 7, B
150 OPEN "COM1:9600,N,7,2,CS,DS" FOR RANDOM AS #1
155 C = 0
160 PRINT #1, "P"
170 PRINT "SELECT 2 POINTS"
180 INPUT #1, X0, Y0, F
190 INPUT #1, X1, Y1, F
195 COL = F
200 LINE (X0*MTS,199-Y0*MTS)-(X1*MTS,199-Y1*MTS), COL
210 IF C = 0 THEN PRINT #1,"e";X0;" ";Y0;" ";X1;" ";Y1: C=C+1
220 IF F <> 8 THEN 160
230 PRINT #1, "f"
240 CLOSE #1
250 END
  
```

3. Порядок подготовки к лабораторной работе

1. Изучить систему команд (приказы) планшета.
2. Изучить формат вводимых в ЭВМ данных.
3. Разработать программу для планшета моделирующую работу мыши.
4. Разработать программу для ввода в ЭВМ графических примитивов: линии, прямоугольника, окружности и дуги с отображением рабочего окна планшета на экране дисплея.

4. Порядок выполнения лабораторной работы

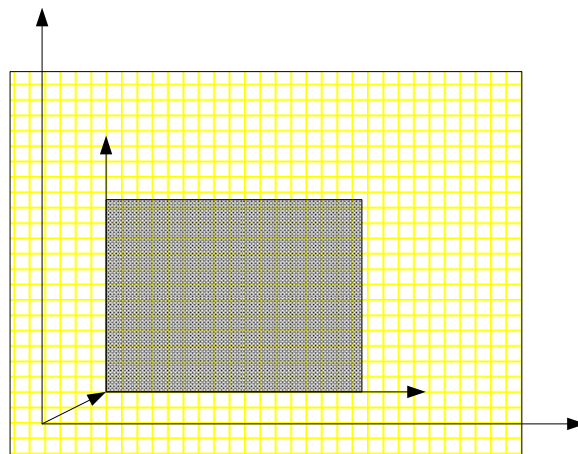
- 4.1. Подсоединить планшет к последовательному порту COM1 и включить.
- 4.2. Включить ЭВМ и загрузить ОС.
- 4.3. Запустить интерпретатор Бэйсик, а с его помощью из каталога LR_TABL запускать на решение поочередно соответствующие файлы, каждый из которых содержит программу, демонстрирующую действие одной из команд планшета.
- 4.4. С помощью команд планшета составить программу на Бэйсике, моделирующую работу мыши.
- 4.5. Разработать для ввода в ЭВМ графических примитивов: линия, прямоугольник, окружность, дуга с отображением рабочего окна планшета на экране дисплея.
- 4.6. По заданию преподавателя модифицировать программу, связывающую рабочее окно планшета с рабочим окном экрана дисплея.

5. Содержание отчета

- 5.1. Наименование и цель работы.
- 5.2. Привести блок-схему алгоритма процесса вычисления координат положения курсора в рабочем поле планшета.
- 5.3. Программу, моделирующую работу мыши.
- 5.4. Программу для ввода в ЭВМ графических примитивов: линии, прямоугольника, окружности и дуги с отображением рабочего окна планшета на экране дисплея.
- 5.5. Методику ввода в ЭВМ чертежей больших размеров.

5.6. Выводы

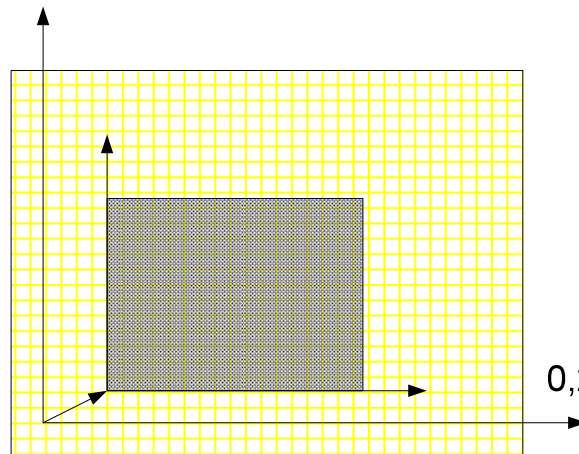
6. Приложение 1



X_T, Y_T -система координат планшета; M_{XT}, M_{YT} -максимальные размеры рабочего поля планшета; X_{TR}, Y_{TR} -система координат рабочего окна планшета; X_{TR0}, Y_{TR0} -начало координат рабочего окна планшета; M_{XTR}, M_{YTR} - максимальные размеры рабочего окна планшета.

Y_T Система координат
0,3100 Tablet

Y_{TR}
 M_{XTR}



Система координат
Screen

XS,YS-система координат экрана; MXS,MYS максимальные размеры рабочего поля экрана; XSR,YSR-система координат рабочего окна экрана; XSR0,YSR0-начало координат рабочего окна экрана; MXSR,MYSR-максимальные размеры рабочего окна экрана.

YSR

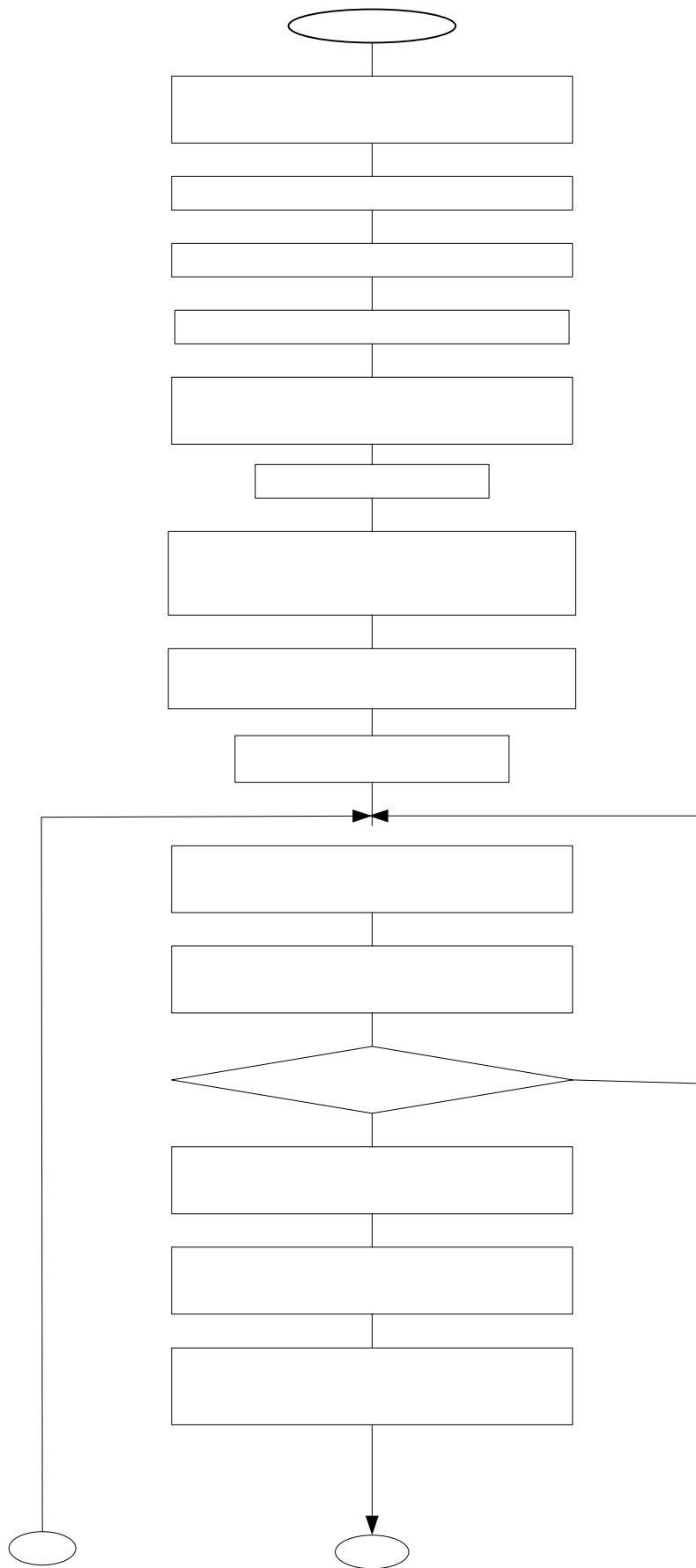
MXSR

Рабочее окно экра

XSR0,YSR0

0,0

7. Схема алгоритма программы Tbsc.bas



НА

Очищаем экран
графиче

Задаем рабочие ок

Резервируем ме

Определяем фо

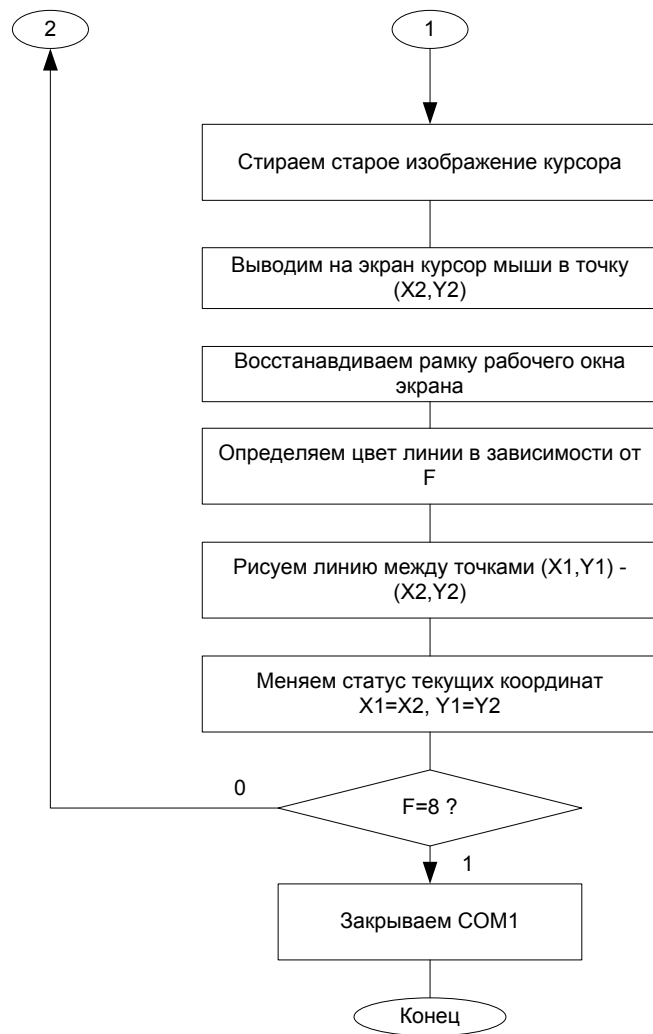
Изображение мыш
G%

Очища

Задаем начальны
Рисуем курсор в

Рисуем рамки раб
окна

Открывае



7.1.ПРИЛОЖЕНИЕ 1 (продолжение)

Программа **Tbsc.bas**

```
100 CLS : SCREEN 1, 0
110 MXT = 4600: MYT = 3100: MXS = 319: MYS = 199
120 XTR0 = 1000: YTR0 = 500: MXTR = 500: MYTR = 500
125 XSR0 = 100: YSR0 = 50: MXSR = 100: MYSR = 100
130 DIM G%(25)
140 LINE (0, 0)-(3, 0), 7: LINE -(2, 1), 7
150 LINE -(4, 3), 7: LINE -(3, 4), 7
160 LINE -(1, 2), 7: LINE -(0, 3), 7
170 LINE -(0, 0), 7
180 PAINT (1, 1), 7
190 GET (0, 0)-(4, 4), G%
191 CLS
192 X1=160:Y1=100
193 PUT (X1, Y1), G%, XOR
200 LINE (0, 0)-(MXS, MYS), 7, B
205 LINE (XSR0, 199 - YSR0)-(XSR0 + MXSR, 199 - (YSR0 + MYSR)), 7, B
210 OPEN "COM1:9600,N,7,2,CS,DS" FOR RANDOM AS #1
220 PRINT #1, "P"
REM 230 PRINT #1, "c1000,500"; CHR$(&HA); CHR$(&HD)
240 PRINT #1, "Y"
250 INPUT #1, X, Y, F
255 IF X = 9999 THEN 250
260 IF X < XTR0 THEN XTR = 0
270 IF X > XTR0 + MXTR THEN XTR = MXTR
280 IF X > XTR0 AND X < XTR0 + MXTR THEN XTR = X - XTR0
290 IF Y < YTR0 THEN YTR = 0
300 IF Y > YTR0 + MYTR THEN YTR = MYTR
310 IF Y > YTR0 AND Y < YTR0 + MYTR THEN YTR = Y - YTR0
320 A = MXSR / MXTR: B = MYSR / MYTR
330 IF A > B THEN MTSR = B ELSE MTSR = A
350 X2 = INT(XTR * MTSR) + XSR0
360 Y2 = INT(200 - YTR * MTSR) - YSR0
361 PUT (X1, Y1), G%, XOR
370 PUT (X2, Y2), G%, XOR
382 LINE (XSR0, 199 - YSR0)-((XSR0 + MXSR), (199 - (YSR0 + MYSR))),
7, B
390 IF F = 0 THEN COL = 0
400 IF F = 1 THEN COL = 1
410 IF F = 2 THEN COL = 2
420 IF F = 4 THEN COL = 3
```



```
430 LINE (X1, Y1)-(X2, Y2), COL
440 X1 = X2: Y1 = Y2
460 IF F = 8 THEN 470 ELSE 220
470 CLOSE #1
480 END
```

8. ПРИЛОЖЕНИЕ 2

Операторы BASIC, используемые в программе tbsc.bas.

LINE - графический оператор, предназначенный для вывода отрезка или прямоугольника на экран.

Формат : *LINE [(x1,y1)] - (x2,y2)[,[цвет]][,B[F]]*

Примечание : здесь и далее параметры, указанные в квадратных скобках, являются необязательными.

Параметры :

$(x1,y1)$, $(x2,y2)$ - пары координат, определяющие крайние точки отрезка или противоположные вершины прямоугольника. При незадаваемых $(x1,y1)$ позиция первой крайней точки определяется координатами текущего положения курсора;

цвет - код цвета [0,3];

B - вывод контура прямоугольника;

BF - вывод закрашенного прямоугольника.

GET(графический) - запись сформированного образа изображения с экрана в ОП персонального компьютера.

Формат : *GET (x1,y1) - (x2,y2), имя_массива*

Параметры :

$(x1,y1)$ - координаты верхнего левого угла;

$\{x2,y2\}$ - координаты нижнего правого угла;

имя_массива - имя массива, в который записывается образ.

PUT(графический) - вывод образа изображения из массива памяти на экран.

Формат : *PUT (x,y), имя_массива, операция*

Параметры :

(x,y) - координаты верхнего левого угла прямоугольной области экрана, ограничивающей воспроизводимый фрагмент;

имя массива - имя массива, в котором хранится образ изображения;

операция - тип операции, выполняемой над хранящимися в массиве кодами цветов пикселей прежде, чем они будут преобразованы в видимое изображение: *PSET* - код пикселя не меняется;

PRESET - инверсия кода пикселя;

XOR - ИСКЛЮЧАЮЩЕЕ ИЛИ кода из массива и кода из памяти экрана;

OR - ИЛИ кода из массива и кода из памяти экрана;

AND - И кода из массива и кода из памяти экрана.

PAINT - заполнить область экрана заданным цветом вплоть до заданной границы.

Формат : *PAINT* (x,y) , *цвет_окраски*, *цвет_границы*

Параметры :

(x,y) - координаты произвольной точки, находящейся внутри закрашиваемой области;

цвет_окраски - код цвета [0,3], определяющий цвет закрашиваемой области;

цвет_границы - код цвета [0,3], идентифицирующий контур, который должен служить границей закрашиваемой области.

OPEN "COM..." - открывает файл телепередачи.

Формат : *OPEN "COM n: [скорость] [,четность] [,биты] [,stop] [,RS] [,CS[m]] [,DS[m]] [,CD[m]] [,LF]" AS [#]номер_файла [LEN=количество]*

Параметры :

n - константа, равная 1 или 2, определяющая номер асинхронной схемы телепередачи;

скорость - скорость передачи данных (число битов в секунду). Допустимые значения параметра: 75, 110, 50, 300, 600, 1200, 1800, 2400, 4800, 9600. По умолчанию 300 битов в секунду;

четность - способ контроля четности во время передачи :

S - передача и прием нулевого бита четности;

O - передается нечетное число нулевых битов , при приеме выполняется проверка нечетности;

M - передача и прием бита четности, равного 1 ;

E - передается четное число ненулевых битов, при приеме - контроль по четности;

N - не передается бит четности, не выполняется контроль по четности.

По умолчанию - *E* .

биты - число передаваемых битов байта. Допустимые значения 4, 5, 6, 7 и 8. По умолчанию 7.

stop - число стоповых битов. Допустимые значения 1 и 2. По умолчанию 2 для скоростей передачи 75 и 110 битов в секунду, для остальных 1. Если параметр *биты* равен 4 или 5, то *stop* принимает значение 1.5 стоповых бита;

RS - устанавливает низкий уровень линии RTS (Request To Send). По умолчанию линия RTS устанавливается на высоком уровне;

CS - проверяет состояние линии CTS (Clear To Send);

DS - проверяет состояние линии DSR (Data Set Ready);

CD - проверяет состояние линии CD (Carrier Detect);

LF - передает код <LF> после каждого кода CR;

[m] - время ожидания сигналов CTS, DSR и CD [0,65535]мс, если указанный сигнал не появится за заданное время, то сигнализируется ошибка " Device Timeout ". Если параметр *m* отсутствует или равен нулю, то состояния указанной линии не проверяются;

номер_файла - допустимый номер файла, который приписан к файлу пока он открыт, его можно использовать другими операторами телепередачи, связанными с этим файлом;

количество - максимальное число читаемых или записываемых в буфер телепередачи байтов. По умолчанию 128 байтов.

CLOSE - завершение операции ввода/вывода для устройства или файла.

Формат : **CLOSE** [[#]*номер_файла* [, [#]*номер_файла*]...]

Параметры :

номер_файла - номер открытого оператором **OPEN** файла.

Примечание : оператор **CLOSE** без параметров (аналогично оператору **END**) закрывает все открытые файлы и устройства. Закрытые файлы или устройства могут быть вновь открыты с такими же или другими номерами. Освободившийся номер может быть использован при открытии другого файла или устройства.

PRINT # - запись данных в последовательный файл.

Формат : **PRINT #***номер_файла*, [**USING** *v\$*;] *список_выражений*

Параметры :

номер_файла - номер, с которым выходной файл был открыт ;

v\$ - текстовое выражение, определяющее поля и формат выводимых текстов и чисел ;

список_выражений - список выводимых в файл числовых или текстовых выражений, разделенных запятой или точкой с запятой.

INPUT # - чтение данных из файла или устройства с последовательным доступом и присвоение их значений переменным в программе.

Формат : *INPUT* #номер_файла, переменная[, переменная]...

Параметры :

номер_файла - номер, с которым файл был открыт ;

переменная - числовая или текстовая переменная или элемент массива, которым будет присвоено прочитанное значение.

Примечание : типы данных должны соответствовать типам переменных.

ЛИТЕРАТУРА

1. DIGITIZER KD 4600. USER'S Manual. Graphtec corporation. TOKIO, JAPAN.

Лабораторная работа № 4

Устройство вывода графической информации из ЭВМ (плоттер-plotter)

1. Цель работы:

- 1.1. Изучить технические характеристики планшетного плоттера и языки его программирования.
- 1.2. Изучить программно-аппаратные средства подключения плоттера к ЭВМ.
- 1.3. Получить навык в программировании чертежей и выводе их на плоттер.

2. Теоретические сведения

Данный плоттер разработан на базе многолетнего накопленного опыта Graphtec в производстве X - Y плоттеров и включает передовые технологии в этой области.

Данный плоттер подходит для широкого диапазона применений: включает рисование всех типов графики, формирование статистических чертежей, инженерные разработки (CAD), построение NC контролирующих мониторов, компьютерное искусство.

Внедрение современного 5- фазного шагового двигателя позволило добиться разрешения 25 мкм и максимальной скорости 40 см/с впервые в плоттере этого типа.

2.1. Описание и функциональные части плоттера.

Непосредственно на самом устройстве можно определить:

1. *Панель карандашей* с набором карандашей готовых для рисования.
2. *Держатель карандаша или каретку*, содержащую карандаш используемый для рисования.
3. *Y- стойку*, перемещающую каретку карандаша влево-вправо.
4. *Управляющую панель*, содержащую клавиши, используемые для управления операциями плоттера и лампочки, которые указывают состояние плоттера.

5. *Панель для рисования*, содержащую бумагу. В изучаемой модели MP 3100 это - магнитный лист. В моделях MP 3200, MP 3300, MP 3400 это - панель с электростатическим притяжением

6. *Рычаг сжатия карандаша* - (MP 3100, MP 3200, MP 3300). В зависимости от положения рычага используются разные типы карандашей.

7. *Выключатель питания*.

8. *Разъем (DC LINE)* для подключения АС адаптера.

9. *DIP переключатели (SW1, SW2)*, устанавливающие конфигурацию компьютера и параметры интерфейса.

10. *2 разъема* для интерфейсов RS-232C и CENTRONICS.

Управляющая панель содержит лампы дисплея и управляющие клавиши.

Лампы дисплея:

POWER - загорается при включении

FAST (зеленый) - горит при быстром режиме (только для MP 3100

)

PAUSE - горит при режиме останова

ALARM / PROMPT (красный) - загорается в случае ошибки

FAST / SLOW (только для MP 3100) - связан с клавишей FAST / SLOW. Горит при быстром (40 см/ в сек.) режиме.

Управляющие клавиши:

PEN UP / DOWN - поднимает и опускает карандаш

P1, P2 - определяют масштабные точки.

Λ ∨ < > - служат для перемещения карандаша в соответствующих направлениях.

ENTER -служит для установки различных функций

PAUSE - нажимать для приостановки рисования.(для продолжения нажмите еще раз)

BUFER CLEAR - когда эта клавиша нажата вместе с клавишей ENTER, содержимое буфера очищается и рисование приостанавливается (осуществляется только в режиме останова).

FAST / SLOW - изменяет скорость рисования (только для MP 3100) (быстрая скорость - 40 см/сек, медленная 10 см/сек.)

Переключатели на задней панели

определяют конфигурацию и параметры интерфейса плоттера. С их помощью можно выполнить:

1. *Установку размеров бумаги.*

Поле плоттера может быть выбрано в соответствии с используемым форматом бумаги. Выбор происходит установкой битов 4 и 5 DIP switch2 на задней панели плоттера.

Размеры бумаги и плоттерные поля.

SW2		Формат страницы	Размер бумаги (мм)	Плоттерное поле (мм)
5	4	Size name	Paper size	(мм)
OFF	OFF	ISO A3	420 x 297	404 x 285
ON	OFF	ANSI B	432 x 280	416 x 268
OFF	ON	ISO A4	297 x 210	281 x 198
ON	ON	ANSI A	280 x 216	264 x 204

2. *Выбор команд (команды плоттера).*

Установка плоттера для набора команд используется в применяющихся программах. Это может быть сделано установкой 6 и 7 битов DIP switch2 на задней панели.

Набор команд и установка.

SW2		Набор команд	
7	6		
OFF	ON	GP- GL	MP
ON	OFF		Digiplot
OFF	OFF	HP - GL	FP- Personal

3. *Установку размера шага (для GP - GL).*

Программируемый размер шага плоттера может быть установлен либо 0.1 мм, либо 0.025 мм. Установка бита 8 DIP switch2 выбирает размер шага комплекта программ на компьютере.

Установки размера шага.

SW2(8)	Step size
ON	0.025 мм
OFF	0.1 мм

4. Установку скорости плоттера.

Изменение скорости рисования осуществляется нажатием < FAST / SLOW > клавиши. Скорость 40см/сек, когда <FAST / SLOW> включены, иначе скорость 10 см/сек.

5. Метод очистки (сброса).

Если клавиша <ENTER> нажата вместе с клавишами ∨ или ∧, то плоттер инициализируется и все установки возвращаются к своим начальным значениям, которые были в момент включения питания.

6. Очистку буфера.

Все данные, хранящиеся в буфере плоттера, стираются следующими операциями. Плоттер переводится в режим < PAUSE >. Далее держа < ENTER > нажать <BUFFER CLEAR> клавишу.

ЗНАЧЕНИЯ DIP SWITCH 1

GP-GL система команд

bit N	8	7	6	5	4	3	2	1
Значение	не испол-зуется	стоповый бит	паритет		длина символов	RS-управление	HANDSHAKE	интерфейсн. режим
ON		2-бита	четный	бит паритета	8-бит	всегда ON	Xon/ /Xoff	RS-232C
OFF	всегда	1-бит	нечет.	нет	7-бит	в течении и передачи	HARDWARE	параллельный

HP-GL система команд

bit N	3	2
ON	esc-команды ввода в буфер	игнорируется
OFF	.обычно	обычно

Бит1 - Всегда установлен в ON.

Бит2 - Когда используется MP-GL команды:

OFF - Обычная позиция.

ON - плоттерные команды <ESC> и <ESC Z> игнорируются.

Бит3 - Когда используется MP - GL команды :

OFF - Обычная позиция.

ON- Интерфейс управления командой < ESC command >, также вводит в буфер.

Бит4 - Выбор длины символов для передачи (7 или 8 бит).

OFF - 7 бит;

ON - 8 бит.

Бит5 - Выбор наличия или отсутствия бита паритета

OFF - нет бита паритета;

ON - бит паритета.

Бит6 - Выбор четного или нечетного паритета

OFF - нечетный;

ON - четный.

Бит7 - Выбор числа стоповых битов

OFF - 1

ON - 2

Бит8 - Для MP 3100 всегда установлен в OFF.

ЗНАЧЕНИЯ DIP SWITCH 2

bit N	10	9	8	7	6	5	4	3	2	1
значение	карандаш	вид карандаша	размер шага	система команд		размер бумаги		baud rate		
ON	активный	активный	0.025							
OFF	неактивный	неактивный	0.1							

5	4	формат страниц
OFF	OFF	ISO A3
ON	OFF	ANSI B
OFF	ON	ISO A4
ON	ON	ANSI A

	3	2	1
9600	OFF	OFF	ON
4800	OFF	ON	OFF
2400	OFF	ON	ON
1200	ON	OFF	OFF
600	ON	OFF	ON
300	ON	ON	OFF

7	6	тип команд
OFF	ON	MP Digiplot
ON	OFF	FP Personal
OFF	OFF	MP - GL

2.2.

2.3.

2.4. Архитектура плоттера

Плоттер представляет собой специализированную ЭВМ архитектура которой представлена на рис 1. Исполнительным элементом плоттера является перо, связанное с шаговыми двигателями, перемещающими его по координатам X,Y. По координате Z перо поднимается и опускается электромагнитом. Из элементарного набора приказов:

1. переместить перо на шаг по координате X ,
2. переместить перо по координате Y ,
3. поднять перо или опустить

Разработаны подпрограммы, набор которых определяет графический язык плоттера. Коды этих подпрограмм записаны в ПЗУ.

Код любого рисунка состоит из последовательного набора имен соответствующих подпрограмм. Например, код (набор литер ASCII) для вычерчивания линии между точками X1,Y1 и X2,Y2 в системе команд HP-GL может иметь следующий вид:

PU;PA1000,2500;SP1;PD;PA2000,2500;PD; ,

где:

PU- Pen Up, **PA-**Plot Absolute, **SP1-** Select Pen 1, **PD-** Pen Down - команды из набора команд приложения 3.

X1=1000,Y1=2500; X2=2000,Y2=2500.

2.5. Структурная схема устройства управления

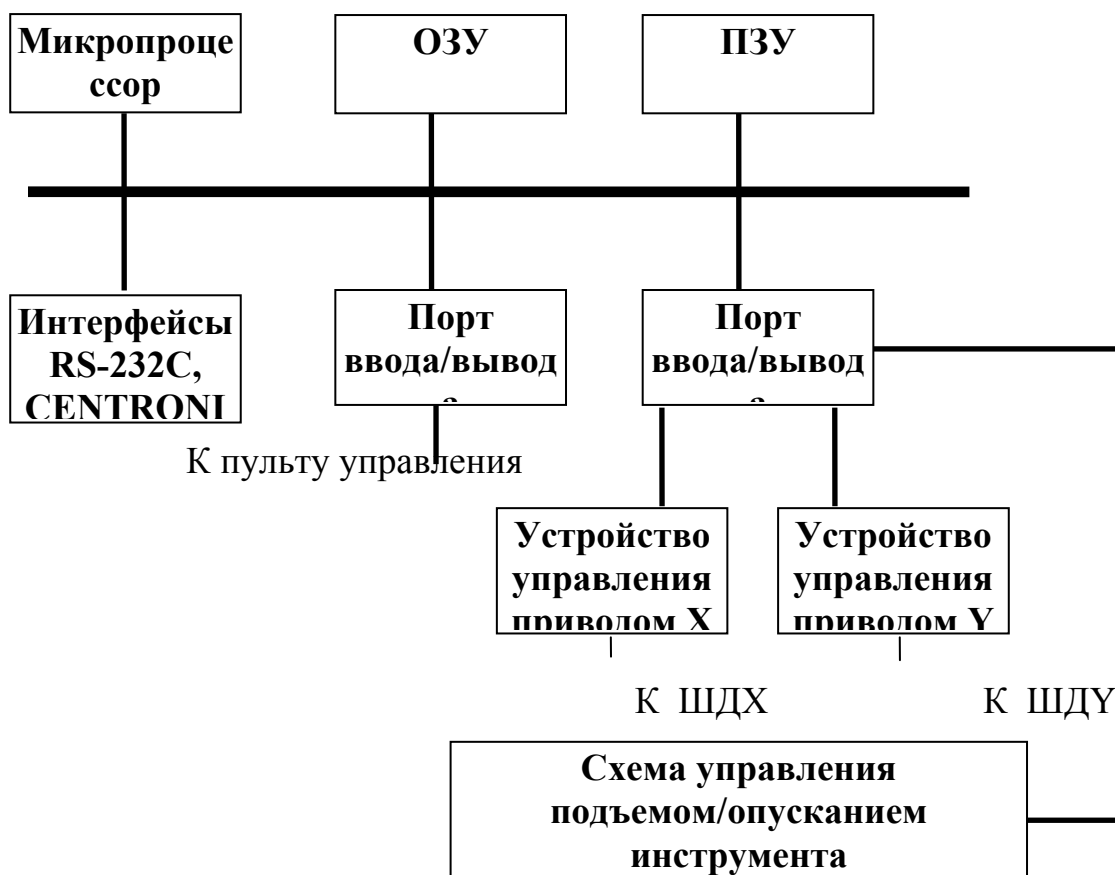


Рис.1

Литеры, описывающие рисунок, передаются на плоттер через порт RS-232C или CENTRONICS и записываются в его ОЗУ.

Последовательность литер можно формировать с помощью ЭВМ. При этом в работе рассматриваются 2 варианта:

1. *Непосредственный* - при котором литеры набираются на клавиатуре ЭВМ и (по мере набора) сразу передаются в плоттер через соответствующий интерфейс.

2. *Файловый* - когда код чертежа записывается в файл с помощью редактора текста, с последующей его “распечаткой” на плоттере.

Структурная схема лабораторной установки для изучения плоттера имеет следующий вид (рисунок 2).

2.6.Схема лабораторной установки

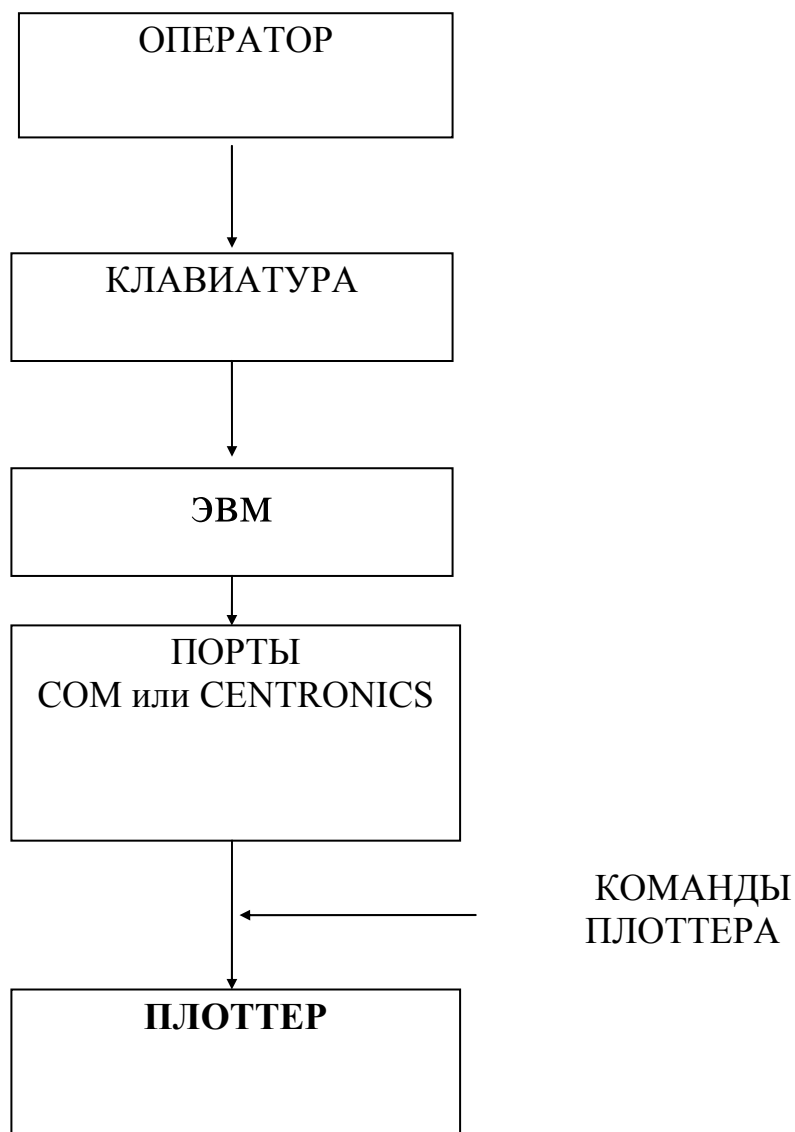


Рисунок 2

Для обеспечения ее работоспособности необходимо решить задачу подключения плоттера к ЭВМ, для чего необходимы

1. аппаратный компонент (кабели) и
2. программный компонент
 - а. программы связи клавиатуры с портом RS-232C или с портом CENTRONICS для пультового режима.
 - б. программы вывода текстового файла (“распечатки файла”) в порты RS-232C или CENTRONICS для файлового режима.

Ниже рассмотрены случаи решения задачи подключения плоттера к ЭВМ через интерфейсы RS-232C и CENTRONICS для пультового режима работы.

2.7. Подключение плоттера к эвм посредством интерфейса RS-232C

Схема кабеля имеет следующий вид:

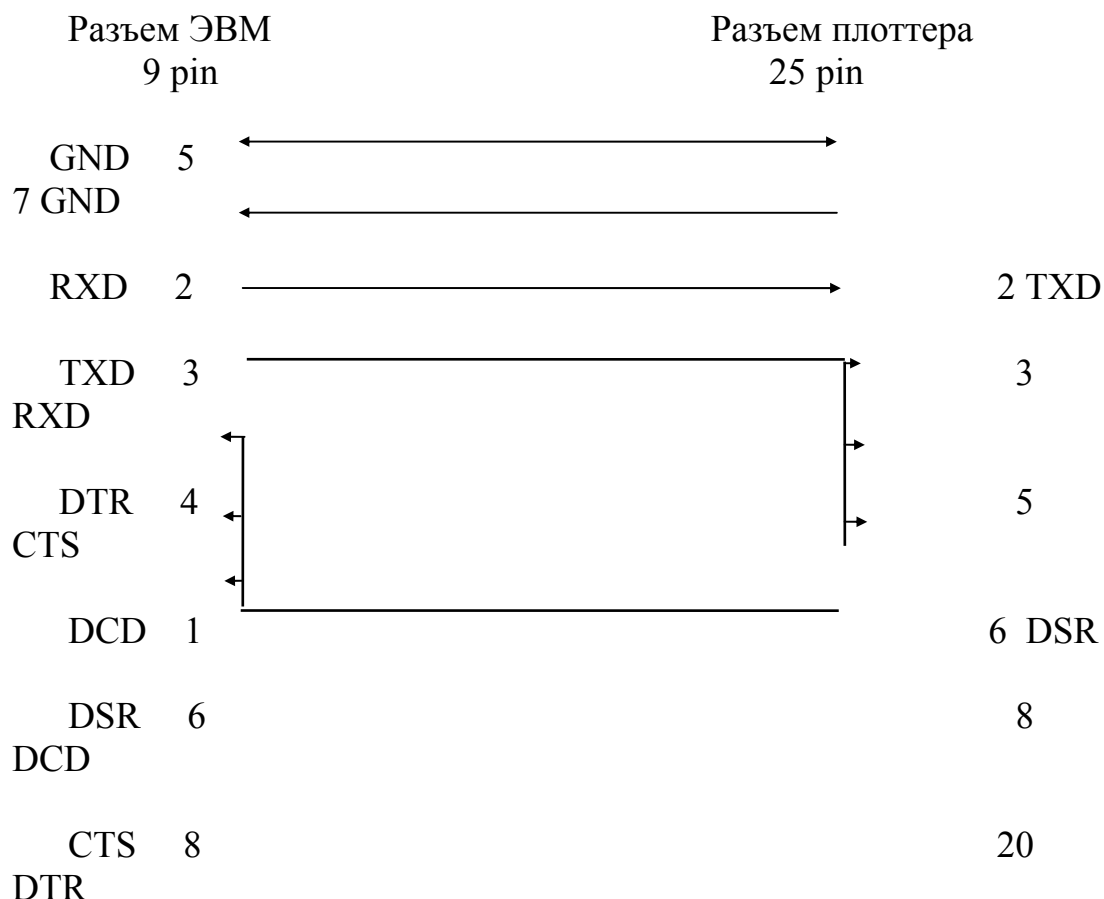


Рис.3

Схема положения переключателей конфигурации плоттера SW1 и SW2

Переключатели расположены на задней стенке плоттера и позволяют устанавливать аппаратно его конфигурацию. Для примера на рис. Показано положение переключателей для следующих параметров конфигурации:

1. Интерфейс RS 232C

Параметры кадра: 8 - информационных бит, 1- стоповый, четный паритет, скорость передачи 300 бит/с

2. Система команд HP-GL

SW1

8	7	6	5	4	3	2	1	
								ON
								OFF

SW2

8	7	6	5	4	3	2	1	
								ON
								OFF

В соответствии с кабелем и установленными переключателями SW1, SW2 (аппаратная часть), программа для подключения плоттера к ЭВМ для пультового режима имеет вид приведенный в приложении 1. а для файлового режима в приложении 2.

2.8. Подключение плоттера к ЭВМ посредством интерфейса CENTRONICS

Для этого случая применяется стандартный кабель такой же, как и для подключения принтера.

Положение переключателей SW1, SW2 может иметь следующий вид:

SW1

8	7	6	5	4	3	2	1	
								ON
								OFF

SW2

8	7	6	5	4	3	2	1	
								ON
								OFF

Программа для подключения плоттера к ЭВМ с помощью интерфейса CENTRONICS приведена в приложении 2.

3. Порядок подготовки к лабораторной работе:

- 3.1. Изучить язык программирования плоттера HP-GL (Hewlett Packard Graphic Language), пользуясь приложением №3.
- 3.2. Разработать программу для чертежа, заданного в приложении №1, используя язык программирования HP-GL.
- 3.3. Повторить команды Ассемблера.
- 3.4. Изучить программу PRINTER.ASM, которая приведена в приложении №2.

4. Порядок выполнения лабораторной работы:

- 4.1. Вставить в планшет пишущие элементы.
- 4.2. Подключить планшет к параллельному порту CENTRONICS и включить.
- 4.3. Включить ЭВМ с загрузкой ОС.
- 4.4. Запустить программу PRINTER.EXE
- 4.5. Для пробы на клавиатуре большими буквами набрать команду SP1; (Set Pen 1 -взять перо №1) и убедиться, что она выполнится. Символ «;» является как разделителем команд, так и инициатором их выполнения.
- 4.6. По заданию преподавателя ввести в плоттер команды и проследить их выполнение. Если при наборе команды допущена ошибка, то на пульте плоттера загорается сигнальная лампочка ALARM/PROMPT. В этом случае плоттер лучше инициализировать командой IN;
- 4.7. По нажатию клавиши «=>» вернуться в NC. С помощью редактора NC сформировать файл **c:\plot.drw** с программой заданного Вам чертежа.
- 4.8. Выполнить команду DOS **COPY c:\plot.drw PRN** и убедиться в правильности прорисовки вашего чертежа.
- 4.9. Запрограммировать символ пользователя по заданию преподавателя и прорисовать его с помощью плоттера
- 4.10. Пользуясь командами символьной группы, получить твердую копию вашей программы **c:\plot.drw**

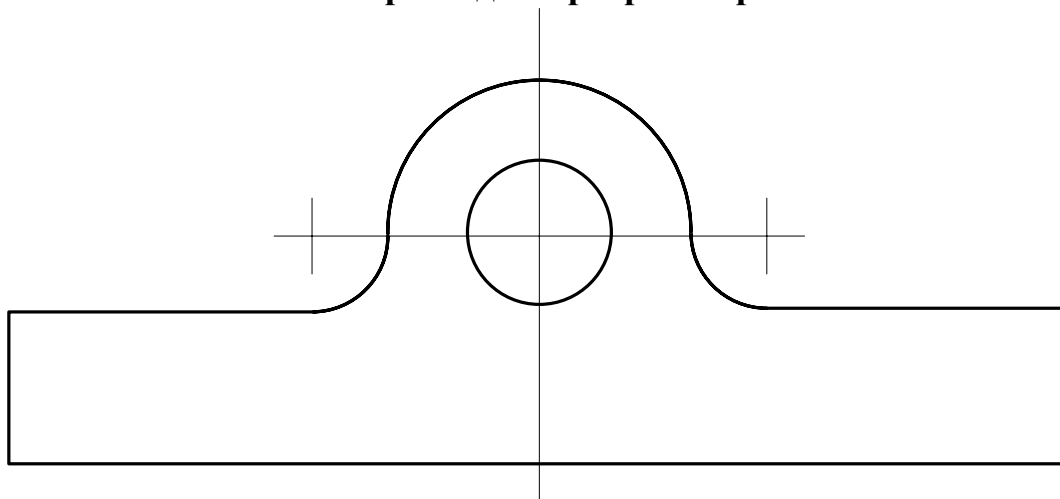
5. Отчет должен содержать:

- 5.1. Основные характеристики плоттера;

- 5.2. Архитектуру плоттера как специализированной ЭВМ;
- 5.3. Описание изученных команд;
- 5.4. Схему подключения плоттера к ЭВМ;
- 5.5. Программную часть, обеспечивающую это подключение;
- 5.6. Заданный чертеж и его программу.

6. Приложение 1

Чертеж для программирования



7. Приложение 2

```
title printer.asm ;
stack segment para stack ;
db 8 dup('stack ')
stack ends
```

```
cseg segment para ;
assume cs:cseg,ss:stack;
```

```
main proc far
```

```
push ds
sub ax,ax
push ax
```

```
go: mov ah,8
int 21h
```

```
mov ah,6
mov dl,al
int 21h
```

```

cmp al,'='
je go1

mov ah,5
int 21h

jmp go

go1: ret

main endp
cseg ends
end main

```

8. Приложение 3

Система команд HP-GL

Векторная группа

AA : (ARC ABSOLUTE) Absolute - coordinate arc command

Назначение: *Рисует дугу определенную в абсолютных координатах ;*

Функция: AA x, y, Q(, f) ;

Пример: AA 1000, 1000, 60 (,5) ;

Ограничение: - 32767 <= x, y, Q <= 32767
 - 32767 <= f < 32767

Примечание: x, y - абсолютные координаты центра дуги ;
 Q - центральный угол дуги ;
 f - разрешение ;



AR : (ARC RELATIVE) Relative - coordinate arc command

Назначение: *Рисует дугу определенную в относительных координатах*

Функция: AR dx, dy, Q (,f) ;

Пример: AR 1000, 1000, 60 (,5) ;

Ограничение: - 32767 <= dx,dy,Q <= 32767 ;
- 32767 <= f < 32767 ;

Примечание: x, y - относительные координаты центра дуги ;



CI : (CIRCLE) Circle plotting command

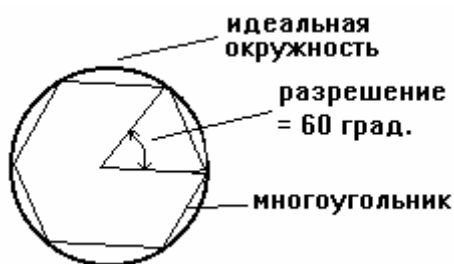
Назначение: *Рисует окружность и различного угла многоугольники*

Функция: CI r, f ;

Пример: CI 2000, 5 ;

Ограничение: - 32767 <= r, f <= 32767 ;

Примечание: r - радиус ; f - разрешение ;



PA : (PLOT ABSOLUTE) Absolute - coordinate movement command

Назначение: *Передвигает карандаш в заданные координаты (в абсолютных координатах) ;*

Функция: PA x1, y1(, ..., xn, yn) ;

Пример: PA 2000, 1500 ;

Ограничение: - 32767 <= x, y <= 32767 ;

Примечание: x, y - абсолютные координаты, в которых

передвигается карандаш из текущего положения ;

PU/PP : (PEN UP / PEN POWN) Pen control command

Назначение: *Поднимает / опускает карандаш в текущей позиции ;*

Функция: PU ; PU x, y ;

PP ; PP x, y ;

Пример: PU 0, 1500 ;

PP 0, 1500 ;

Ограничение: - 32767 <= x, y <= 32767 ;

PR : (PLOT RELATIVE) Relative - coordinate movement command

Назначение: *Передвигает карандаш в точку, которая определяется относительными координатами ;*

Функция: PR dx, dy (, ..., dxn, dyn) ;

Пример: PR 0, 1500 ;

Ограничение: - 32767 <= x, y <= 32767 ;

Примечание: x, y - относительные координаты, в которых передвигается карандаш из текущего положения ;

FT : (FILL TYPE) Shading selection command (filling and hatching)

Назначение: *Выбор типа промежутка между линиями закрашки и угла наклона закрашки ;*

Функция: FT n, d, Q ;

Пример: FT 4, 100, 45 ;

Ограничение: 1 <= n <= 4 ;


0 <= d <= 32767 ;


- 45 <= Q <= 45 ;

Примечание: n - тип закрашки :

n = 1 - с одним проходом

n = 2 - с двумя проходами

n= 3 

n= 4 

d - расстояние между линиями закрашки ;

Q - угол наклона закрашки ;

RA : (SHADE RECTANGLE ABSOLUTE) Absolute - coordinate rectangle shading

Назначение: *Определяет вид прямоугольника в абсолютных координатах и закрашивает, заранее выбранным видом закрашки, определенную область ;*

Функция: RA x, y ;

Пример: RA 4000, 3000 ;

Ограничение: $- 32767 \leq x, y \leq 32767$;

Примечание: x, y - абсолютные координаты противоположного угла прямоугольника относительно текущего положения карандаша ;



EA : (EDGE RECTANGLE ABSOLUTE) Absolute - coordinate rectangle plotting

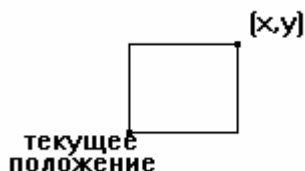
Назначение: *Рисует прямоугольник в абсолютных координатах;*

Функция: EA x, y ;

Пример: EA 4000, 3000 ;

Ограничение: $- 32767 \leq x, y \leq 32767$;

Примечание: x, y - абсолютные координаты противоположного угла прямоугольника относительно текущего положения карандаша :



RR : (SHADE RECTANGLE RELATIVE) Relative - coordinate rectangle shading

Назначение: *определяет вид прямоугольника в относительных координатах и закрашивает заранее выбранным типом закрашки, определенную область ;*

Функция: RR dx, dy ;

Пример: RR 1000, 1000 ;

Ограничение: - 32767 <= dx, dy <= 32767 ;

Примечание: x, y - относительные координаты противоположного угла прямоугольника относительно текущего положения карандаша;



ER : (EDGE RECTANGLE RELATIVE) Relative - coordinate rectangle plotting

Назначение: *Рисует прямоугольник в относительных координатах ;*

Функция: ER dx, dy ;

Пример: ER 1000, 1000 ;

Ограничение: - 32767 <= dx, dy <= 32767 ;

Примечание: dx, dy - относительные координаты противоположного угла прямоугольника относительно текущего положения карандаша



WG : (SHADE WEDGE) Segment shading

Назначение: *Используется вместе с FT и PT командами, определяет сегмент окружности и закрашивает его ;*

Функция: WG r, l, Q, f ;

Пример: WG 1000, 90, 180, 5 ;

Ограничение: - 32767 <= r, Q <= 32767 ;

$$1 \leq f \leq 120;$$

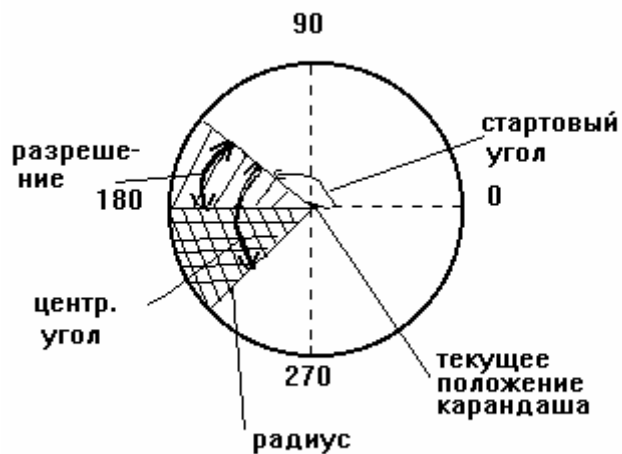
$$-360 \leq l \leq 360;$$

Примечание: r - радиус сегмента ;

l - стартовый угол ;

Q - центральный угол ;

f - разрешение ;



EW : (EDGE WEDGE) Segment drawing

Назначение: *Рисует сегмент окружности ;*

Функция: EW r, l, Q, f ;

Пример: EW 1000, 90, 180, 5;

Ограничение: $-32767 \leq r, Q \leq 32767$;

$$1 \leq f \leq 120 ;$$

$$-360 \leq l \leq 360 ;$$

Примечание: r - радиус окружности; l - стартовый угол; Q - центральный угол; f - разрешение ;

Символьная группа

CP : (CHARACTER PLOT) Character movement command

Назначение: *Перемещает карандаш на расстояние, определяемое количеством символов ;*

Функция: CP n, m;

Пример: CP 5, 35;

Ограничение: $-127,999 \leq n, m \leq 127,999$;

Примечание: n - количество символов в горизонтальном, направлении, m - в вертикальном направлении ;

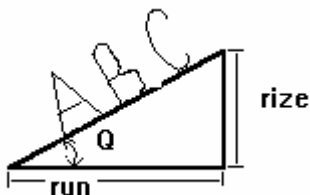
DI : (ABSOLUTE DIRECTION) Absolute character - direction specification command

Назначение: *Определяет наклон строки символов в абсолютных координатах ;*

Функция: DI run, rise ;

Пример: DI 10, 10 ;

Ограничение: $-127,999 \leq \text{run}, \text{rise} \leq 127,999$; Примечание: run - длина наклона ; rise - высота наклона ;



DR : (RELATIVE DIRECTION) Relative character - direction specification command

Назначение: *Определяет наклон строки символов в относительных координатах ;*

Функция: PR run, rise ;

Пример: PR 2, 1 ;

Ограничение: $-127,999 \leq \text{run}, \text{rise} \leq 127,999$;

Примечание: run - процент от $|P2x - P1x|$, rise - процент от $|P2y - P1y|$. Употребляется с IP ;

DT : (DEFINE TERMINATOR) Character string terminator specification command

Назначение: *Определяет терминатор для символьной строки (для печати) ;*

Функция: DT t ;

Пример: DT 01H ;

Примечание: t - ASCII управляющий символ, который может быть определен как параметр;

LB : (LABEL) Character string writing command

Назначение: *Печатает символьную строку с заданными символьными установками ;*

Функция: LB c1...cn t ;

Пример: LB ABCDE ;

Ограничение: c - 01H до 7FH ;

Примечание: c1...cn - символьная строка ; t - ASCII-код символа определяемый параметром DT команды ;

SI : (ABSOLUTE CHARACTER SIZE) Absolute character size specification command

Назначение: *Определяет размер символа в абсолютных значениях ;*

Функция: SI w, h ;

Пример: SI 0.5, 0.75 ;

Ограничение: $- 127.999 \leq w, h \leq 127.999$;

Примечание: w - длина символа ; h - высота символа ; Размер определяется в сантиметрах.

SL : (CHARACTER SLANT) Character string inclination specification command

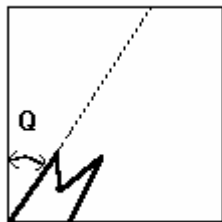
Назначение: *Определяют наклон символов ;*

Функция: SL tan Q ;

Пример: SL 1 ;

Ограничение: $0.05 \leq \tan Q \leq 2$ - для стандартного размера ; $0.05 \leq \tan Q \leq 3.5$ - для длинного размера символов ;

Примечание: tan Q - тангенс угла ;



SR : (RELATIVE CHARACTER SIZE) Relative character size specification command

Назначение: *Определяет размер символов в относительных значениях*

Функция: SR w, h ;

Пример: SR 0.5, 0.75 ;

Ограничение: $-127.999 \leq w, h \leq 127.999$ ($< > 0$);

Примечание: w - длина символа / $P2x - P1x$ /, h - высота символа / $P2y - P1y$ /. Используется с IP ;

SS : (SELECT STANDART SET) Standart character set specification

Назначение: *Возможность печати символов в стандартном символьном наборе;*

Функция : SS;

Пример: SS;

UC : (USER DEFINED CHARACTER) User-defined character command

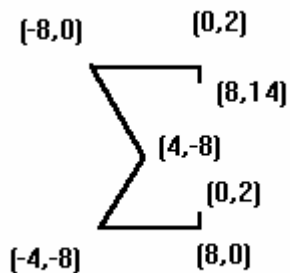
Назначение: *Печатает любые символы;*

Функция : UC(i,) x1, y1,(i,) xn,yn;

Пример: UC 8, 14, 99;

Ограничение: $i \geq 99$ - карандаш опущен; $i \leq -99$ - карандаш поднят; $-98 \leq x, y \leq 98$;

Примечание: i - управляющее значение карандаша; x,y - расстояние, на которое смещается карандаш в вертикальном и горизонтальном направлении; (карандаш смещается вправо, если положительное значение; влево - если отрицательное значение); UC 8, 14, 99, 0, 2, -8, 0, 4, -8, -4, -8, 8, 0, 0, 2;



ГРУППА ТИПА ЛИНИИ

LT : (LINE TYPE) Broken - line mode selection command

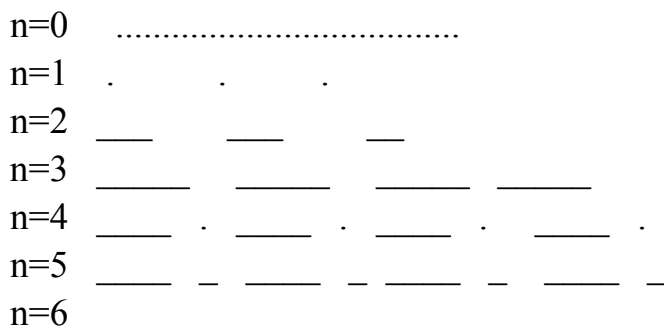
Назначение: *Определяет длину линии и 7 типов ее прерываний*

Функция: LT n,l ;

Пример: LT 1,20;

Ограничение: $0 \leq n < 6$; $0 \leq l \leq 127,999$;

Примечание: n - тип линий ; l - длина линии без прерывания ;



SM : (SYMBOL MODE) Symbol specification command

Назначение: *Рисует символ в указанной точке плоттера*

Функция: SMc ;

Пример: SM* ;

Ограничение: c - 01H до 7FH

Примечание: c - символ

SM * ; PA 3000 , 3000 ; PD ; PR 3000 , 0 ;
 * _____ *

SP : (PEN SELECT) Pen selection command

Назначение: *Исполняет программу выбора карандаша*

Функция: SP n ;
Пример: SP 4 ;
Ограничение: $0 \leq n \leq 8$;

Примечание: n - номер карандаша на панели плоттера

VS : (VELOCITY SELECT) Speed specification command

Назначение: *Определение скорости плоттера ;*

Функция: VS v (, n) ;

Пример: VS 10 ;

Ограничение: $1 \leq v \leq 40$; $1 \leq n \leq 8$;

Примечание: v - скорость карандаша (см / сек)

n - карандаш (номер карандаша на панели)

ГРУППА УСТРОЙСТВ

DS : (DIGITIZE CLEAR) Digitization clear command

Назначение: *Сброс ошибки ;*

Функция: DS ;

Пример: DS ;

Примечание: Эта команда используется для восстановления режима PROMT без ввода координат и нажатия < ENTER > на панели плоттера

КООРДИНАТНАЯ ГРУППА

TL (TICK LENGTH) Axis graduation length specification command

Назначение: *Определяет длину на координатной оси*

Функция: TL tp (, tm) ;

Пример: TL100 ;

Ограничение: $-127,999 \leq tp, tm \leq 127,999$;

Примечание: tp - длина XT или YT в положительном направлении ;

$XT = (P2y - P1y) * tp \% ;$

$YT = (P2x - P1x) * tm \% ;$

XT :(X TICK) Axis graduation plotting commands

YT :(Y TICK)

Назначение: *Строит оси по координатам X и Y*

Функция: ХТ ;

УТ ;

Пример: ХТ ;

УТ ;

Примечание: Длина оси это целое значение даваемое командой TL ,
прежде чем выполнять команды ХТ и УТ ;

PU ; PA 8000, 5000 ; PD ; ХТ ; PR 2000, 0 ; ХТ ;



IP : (INPUT P1 & P2 command) P1, P2 setting command

Назначение: Программно устанавливает масштабные точки ;

Функция: IP P1x, P1y, P2x, P2y ;

Пример: IP 3000, 2000, 8000, 7000 ;

Ограничение: $-2^{15} + 1 \leq$ каждый параметр $\leq 2^{15} - 1$;

Примечание: P1x,P1y - устанавливает P1;P2x,P2y - устанавливает P2

;

IW : (INPUT WINDOW command) Plotting area setting command

Назначение: Ограничивает площадь для вывода ;

Функция: IW Xll, Yll, Xur, Yur ;

Пример: IW 4600,2000,7600,7000 ;

Ограничение: Размер бумаги Xll Yll Xur Yur ;

ANSI	A	0	0	10560	8160
------	---	---	---	-------	------

ISO	A4	0	0	11240	7920
-----	----	---	---	-------	------

ANSI	B	0	0	16640	10720
------	---	---	---	-------	-------

ISO	A3	0	0	16160	11400
-----	----	---	---	-------	-------

Примечание: Xll, Yll - устанавливает размер окна в нижнем левом
конце диагональной линии; Xur, Yur - устанавливает размер окна в правом
верхнем конце диагональной линии ;

SC : (SCALE command) Scale setting command

Назначение: Размещает блок пользователя ;

Функция: SC Xmin, Xmax, Ymin, Ymax ;

Пример: SC 0, 10, 0, 10 ;

Ограничение: $-2^{15} + 1 \leq (Xmax - Xmin) \leq 2^{15} - 1$;
 $(Ymax - Ymin)$

Примечание: Xmin, Ymin - координаты P1, а Xmax, Ymax - координаты P2 в блоке пользователя ;

ГРУППА КОМАНД КОНФИГУРАЦИИ И СОСТОЯНИЯ

DF : (DEFAULT command) Plotter function initialization command

Назначение: *Переводит плоттер в стандартный режим (состояние);*

Функция: DF ;

IM : (INPUT MASK command) Mask setting command

Назначение: *Устанавливает маски ;*

Функция: IM e ;

Пример: IM 223 ;

Ограничение:

e величина маски	номер ошибки	СМЫСЛ ОШИБКИ
1	1	использована неизвестная команда
2	2	неправильный N параметра
4	3	использован неправильный параметр
8	4	не используется
16	5	назначен неправильный набор
32	6	символов
64	7	позиция карандаша вышла за
128	8	допустимые пределы
		не используются
		не используются

Примечание: Устанавливает бит 5 в байте состояния плоттера, при условии включения лампы "ERROR" ("ОШИБКА") на передней панели ;

IN : (INITIALIZE command) Plotter initialization command

Назначение: *Определяет начальное положение плоттера*

Функция: IN ;

Примечание: Команда возвращает плоттер в то состояние, когда он был только что включен ;

RO (ROTATE COORDINATE SYSTEM command). Coordinate system rotation command

Назначение: *Вращает систему координат плоттера ;*

Функция: RO n ;

Пример: RO 90 ;

Ограничение: n = 0 или n = 90 ;

Примечание: Команда программно вращает координатную систему на 90 градусов ;

PS :(PAPER SIZE command) Paper size specification

Назначение: *Назначает размер бумаги ;*

Функция: PS n ;

Пример: PS 100 ;

Ограничение: $0 \leq n \leq 127$

Литература

1.X-Y PLOTTER, MP 3000 SERIES, USER'S MANUAL. Graftec corporation, TOKIO, JAPAN.

2.Джордейн Р, Справочник программиста персональных компьютеров типа IBM PC, XT и AT. С.-П.: Финансы и статистика,1992.

**ФЕДЕРАЛЬНАЯ АВИАЦИОННАЯ СЛУЖБА РФ
МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
ГРАЖДАНСКОЙ АВИАЦИИ**

Кафедра вычислительных машин, комплексов,
систем и сетей
Гладышев Ю.С.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
к выполнению лабораторной работы №5
**МАТРИЧНОЕ ПЕЧАТАЮЩЕЕ УСТРОЙСТВО
(PRINTER)**
по курсу
“ПЕРИФЕРИЙНЫЕ УСТРОЙСТВА ЭВМ”
*для студентов IV курса
специальности 22.01
дневного обучения*

Москва -1999

Лабораторная работа № 5

Матричное печатающее устройство (PRINTER)

Цель работы:

- 1 Изучить структуру принтера как структуру специализированной ЭВМ.
- 2 Изучить программу вывода данных в порт **LPT** и временную диаграмму интерфейса **CENTRONICS**.
- 3 Изучить схему алгоритма главной программы принтера.
- 4 Исследовать систему команд принтера.

Описание структурной схемы устройства

Печатающее устройство **LQ100+** является матричным печатающим устройством со средней скоростью печатания. Привод 24 игольчатой печатающей головки и бумаги осуществляется с помощью 2-х шаговых двигателей. Управление реализуется с помощью микропроцессора. Основные характеристики следующие:

1. Скорость печати может достигать 200 символов в секунду (200 cps – char per second) в черновом режиме для плотности печати 12 cpi (char per inch)/
2. Разрешающая способность достигает 360*360 dpi (dot per inch).
3. Реализует систему команд **EPSON ESC/P2** (Epson System Command for Printer).
4. Печатающая головка содержит 24 иглы, которые расположены в соответствии с на рис. 1. Минимальное расстояние между иглами по вертикали составляет 1/60 inch. Поэтому, для обеспечения разрешающей способности 180 dpi (при печати одной графической колонки) требуется 3 байта информации, а для обеспечения разрешающей способности 60 dpi при печати одной графической колонки требуется 1 байт информации.

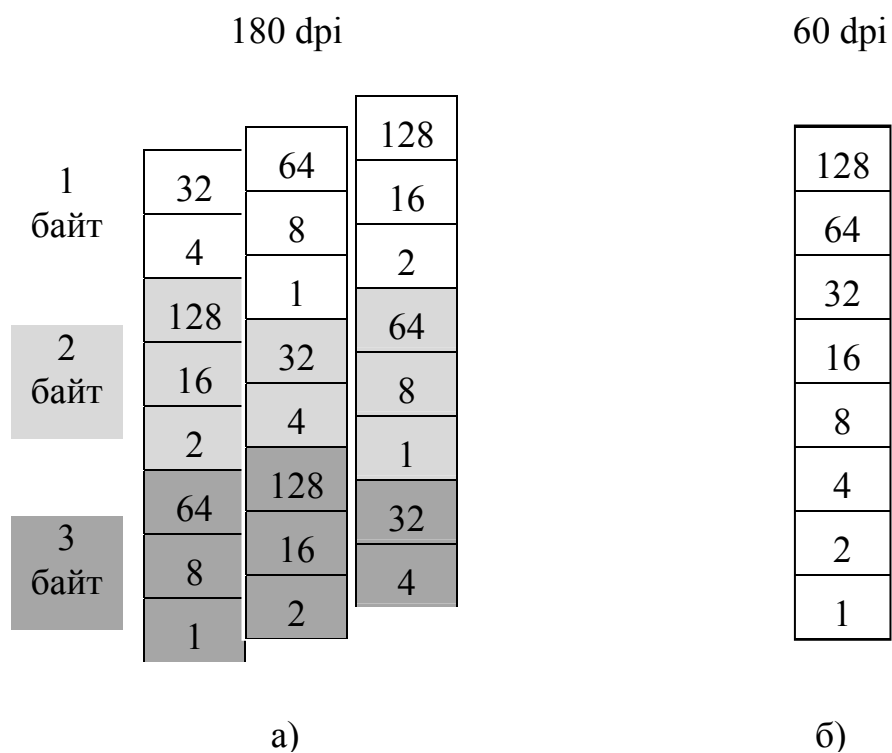


Рис.

Максимальное разрешение 360 dpi получается путем удвоения 180 dpi за счет печати в 2 прохода с перемещением бумаги с шагом 1/360 inch.

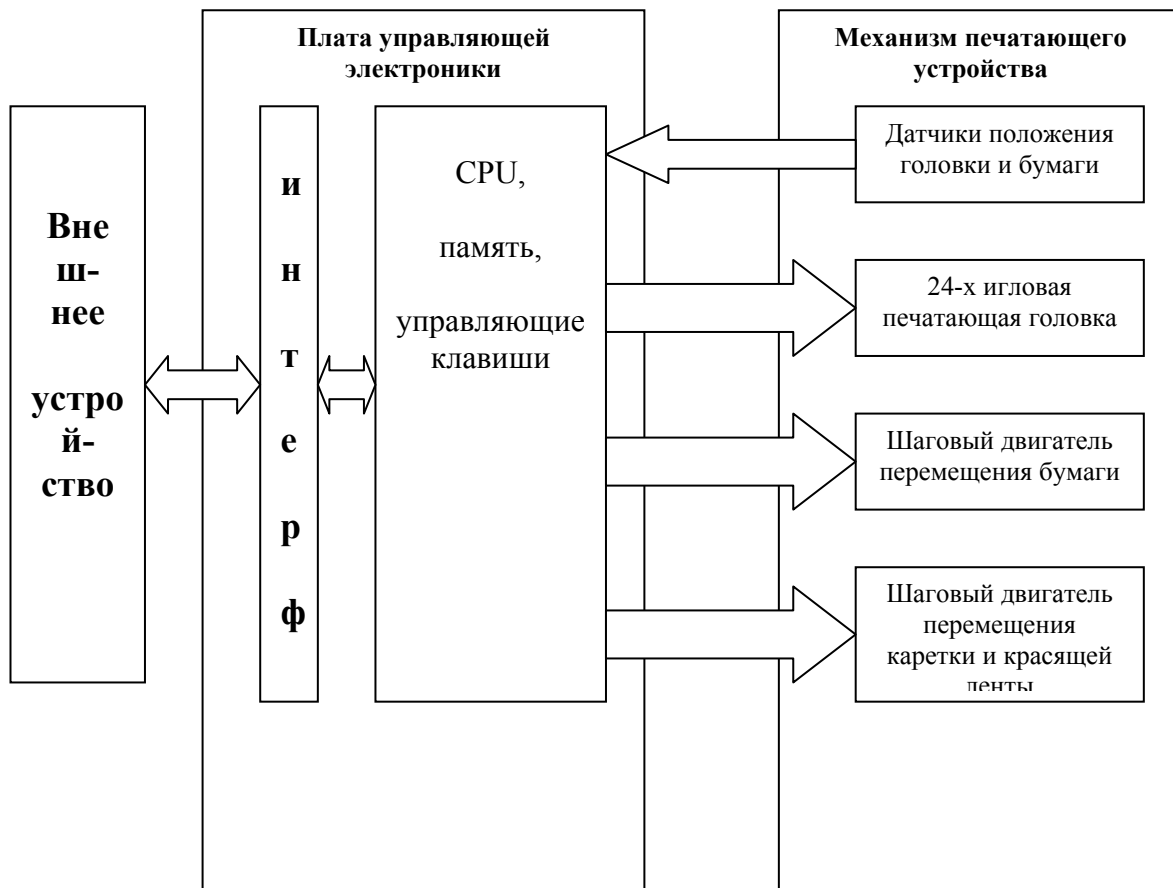


Рис 2

Общий вид структурной схема принтера представлен на рис. 2.

Подключение принтера к ЭВМ аппаратно осуществляется посредством интерфейса Centronics, а программно на низком уровне осуществляется прямым программированием регистра команд с целью формирования импульса **-Strobe** для записи байта данных во входной регистр интерфейса [378h]. Форматы регистров команд [37Ah] и состояний [379h] представлены на рис. 3, программа распечатки символа вводимого с клавиатуры на принтере представлена на рис. 4, а временная диаграмма интерфейса Centronics на рис. 5

Форматы регистров:

Регистр управления [37Ah]		
Биты	Сигналы	Комментарии
0	-Strobe	Вызывает запись байта во вх. регистре
1	-Autofeed	Авто Lf после Cr

2	Init	Инициализация принтера , если 0
3	-Select in	Разрешает сигналы DC1/DC3
4	Inter	Разрешает прохожд. прерываний от Пчу.

Рис. 3.1

Регистр состояния [379h]		
Биты	Сигналы	Комментарии
3	-Error	Ошибка принтера
4	Select	Готовность принтера (On/Off Line)
5	PE	Конец бумаги
6	-Ack	Подтверждение приёма символа в ОЗУ
7	Busy	Принтер занят

Рис. 3.2

Программа вывода данных в порт lpt.

```

title OUT_LPT1
data segment para
message db 'Ошибка принтера $'
data ends
st_seg segment para stack 'stack'
db 64 dup('stack ')
st_seg ends
cseg segment para
assume cs:cseg, ds:data, ss:st_seg
main proc far
push ds
sub ax,ax
push ax
mov ax,40h ; Базовый адрес
mov es,ax ; lpt
mov dx,es:[8] ; в dx.
go: mov ah,1 ; Ввод с клавиатуры в al
int 21h ; с эхом на экране.
cmp al,'=' ; Выход из программы
je go1 ; по символу "=".
out dx,al ; символ на шине данных.

```

```

inc dx      ; Адресуемся к регистру
inc dx      ; управления 37Ah.
mov al,13   ; Цепочка бит для
out dx,al   ; импульса строга.
dec al      ; Формируем импульс
out dx,al   ; строга.
dec dx      ; Адресуемся к регистру статуса 379h.
not: in al,dx ; Вводим в al байт статуса.
test al,8   ; Ошибка?
jz error    ; Да - переход на ошибку.
test al,80h ; Принтер занят?
jz not      ; Если занят, то возврат.
dec dx      ; Адресуемся к регистру данных 378h.
jmp go      ;
error: lea dx,message ; Сообщение об ошибке
mov ah,9    ; вывести на экран.
int 21h
gol: ret
main endp
cseg ends
end main

```

Рис. 4

Временная диаграмма интерфейса.

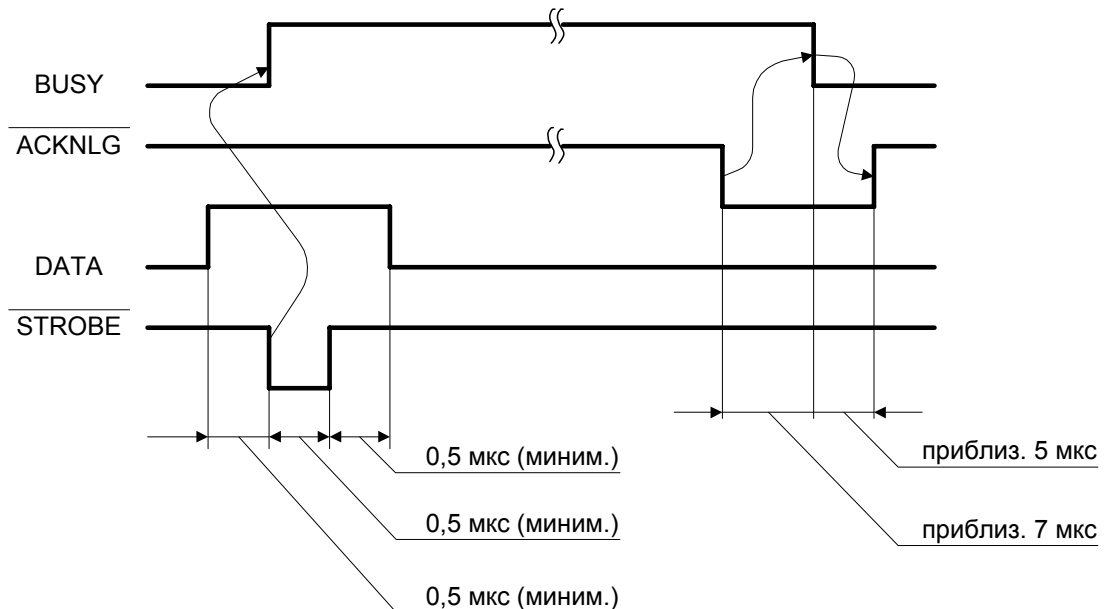


Рис. 5

Порядок выполнения лабораторной работы

1. Выполнить тестирование.

- 1.1. Заправить бумагу в принтер.
- 1.2. Нажать клавишу **FF** и удерживать.
- 1.3. Включить питание.
- 1.4. Для остановки тестирования следует выключить принтер.
- 1.5. Через 1 минуту снова включить принтер.

2. Включить ПК. В процессе загрузки DOS в память будет загружен драйвер принтера и выполнится инициализация принтера (“дернется” каретка). С помощью драйвера принтер будет прописан в вычислительной системе и будет способен выполнять команды DOS.

В результате инициализации по умолчанию установятся параметры принтера, которые обеспечат следующие параметры печати:

- длина страницы 65 линий для формата A4,
- границы полей печати будут иметь значения:
 - Top margine **A**=8.5 мм (0.33 inches),
 - Left margine **Bl** =3 мм (0.12 inches),
 - Right margine **Br** =3.8 мм (0.15 inches),
 - Bottom **C**= 13.5 мм (0.53 inches)

Тогда для формата A4 = (210*297)мм максимальная зона печати составит

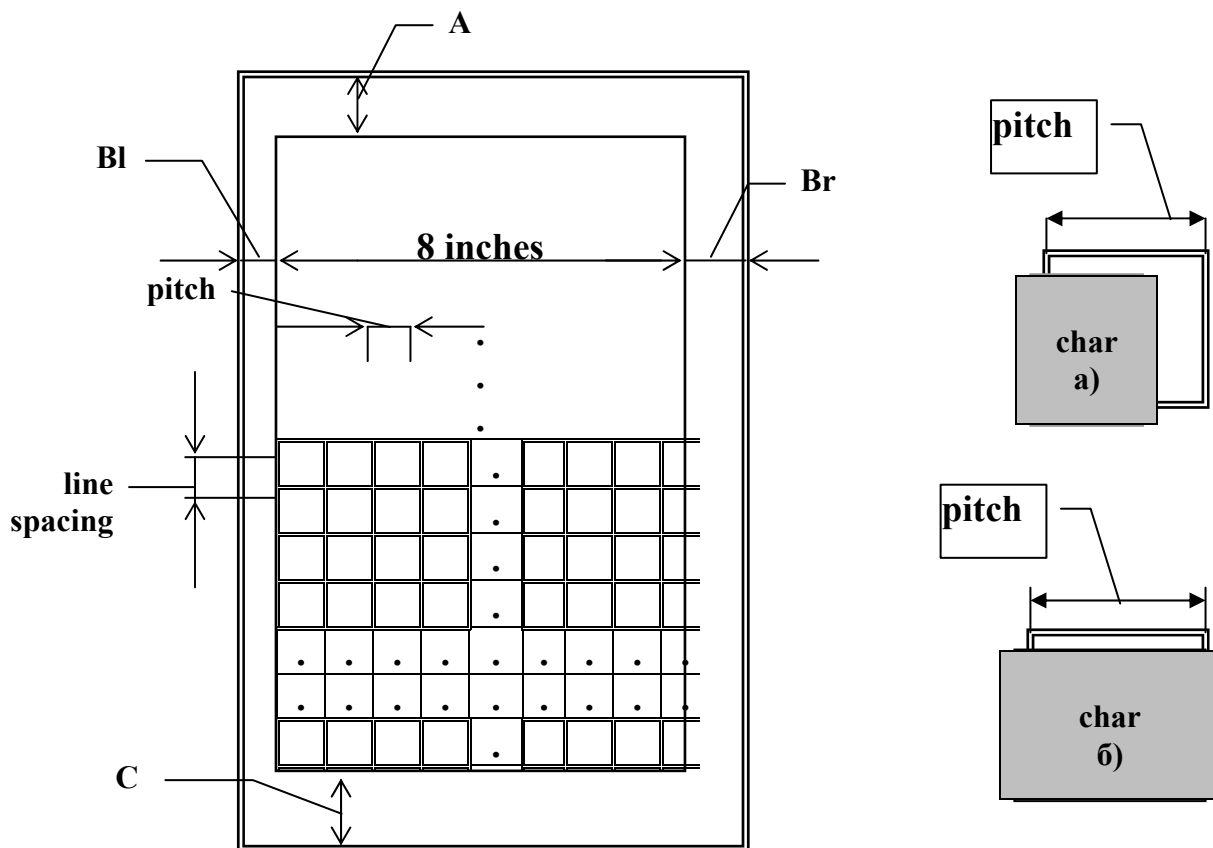


Рис6

(203.2*275)мм или (8*10.83)inches.

- расстояние между строками (line spacing) – 1/6 inch (тогда количество строк составит 65),
- размер знакоместа по горизонтали равен 10.5 точек
- количество литер на дюйм (char per inch -**сpi**) составит 10 cpi для всех шрифтов.

Обозначения параметров представлены на рис. 6, и рис.7, где **char** определяет знакоместо или решетку для конструирования символа, а **pitch** определяет регулярные точки отсчета по горизонтали к которым 'привязываются' решетки для конструирования символа. Рис. 7 поясняет почему размер знакоместа по

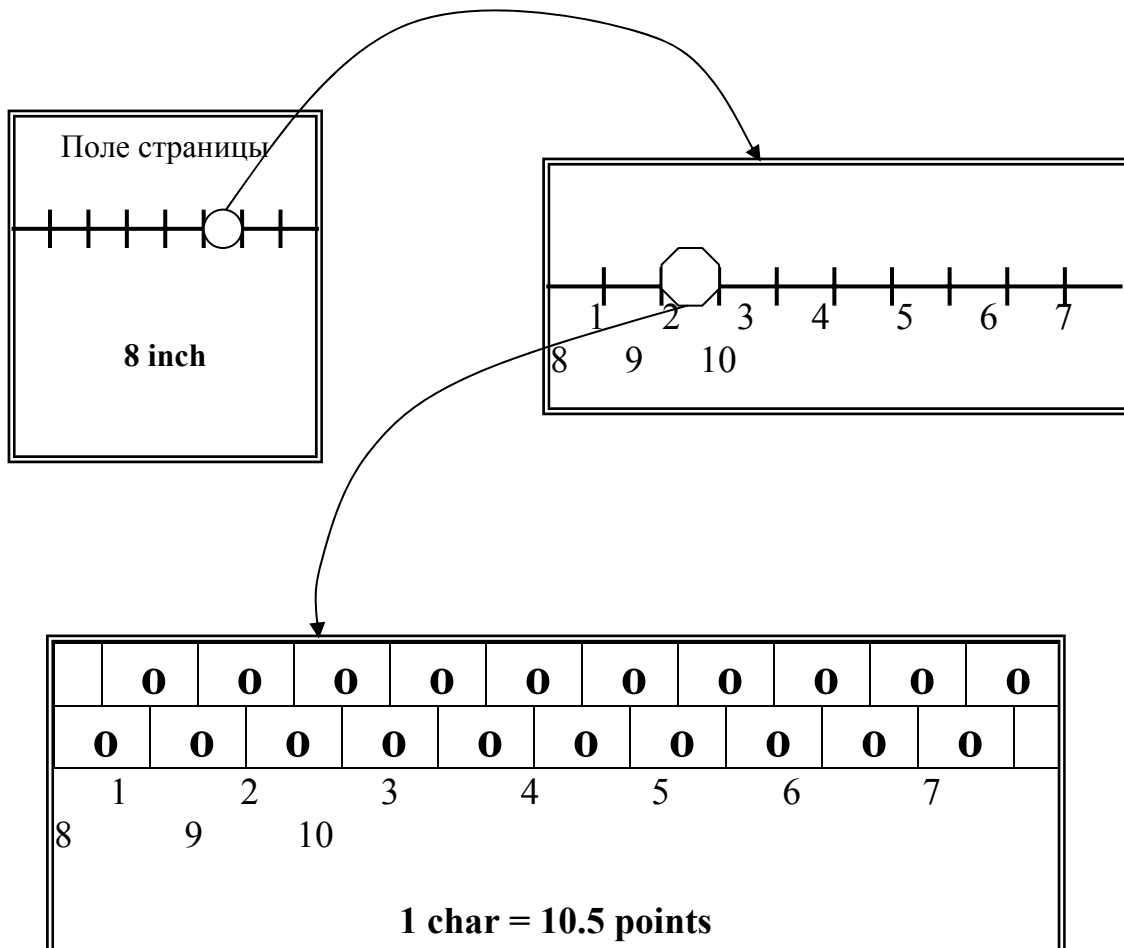


Рис 7

горизонтали равен 10.5 точек.

3. Исследовать алгоритм главной программы принтера

Алгоритм главной программы принтера представлен на рис8. Он отражает наличие 2-х фаз в работе устройства. Во время первой фазы осуществляется ввод текста в ОЗУ (буфер) принтера. Во время второй содержимое ОЗУ выводится на печать. Необходимо исследовать алгоритм перехода из первой фазы ко второй. Для чего:

- 3.1. Находясь в **NC**, запустить на решение программу **Out_Lpt1.exe**, исходный модуль которой представлен на рис. 4. Эта программа обеспечивает работу принтера в режиме пишущей машинки.
- 3.2. Набрать на клавиатуре подряд несколько символов (можно один символ) и подождать, пока они не распечатаются;
- 3.3. Повторить 3.2, только в конце набора последовательности символов сразу нажать **[Enter]**.
- 3.4. Нажимать подряд любую символьную клавишу до тех пор, пока принтер не перейдет в режим печати, при этом посчитать число символов при котором это произойдет.
- 3.5. Соотнести наблюдения с блок-схемой алгоритма главной программы на рис. 8.
- 3.6. Вернуться в **NC**.

4. Исследование границ зоны печати, установленных по умолчанию.

- 4.1. С помощью встроенного редактора **NC** создать текстовый файл **44444**, содержащий несколько строк.
- 4.2. Распечатать этот файл на принтере, используя команду **COPY**, предварительно установив на принтере шрифт печати **Draft**.
Файл будет распечатываться в зоне печати определенной границами установленными по умолчанию.
- 4.3. Посчитать количество символов в строке, а также расстояние до левой и правой границ текста.

5. Изменить количество символов в строке и высоту символов.

Для этой цели можно использовать команду **ESC X m n1 n2** (Select Font by Pitch and Point), в которой

m: Определяет pitch в единицах 360/m cpi

m = 0: Нет изменения в pitch

m = 1: Выбрать пропорциональный

m = 0, 1, 18, 21, 24, 30, 36, 42, 48, 60, 72

n = n1 + n2*256

n: Определяет point size в единицах 0.5 points

Total points = (n1 + n2*256)*0.5

n = 0: Нет изменений в колич. точек

n = 0, 16, 21, 24, 28, 32, 36, 40, 44, 48, 52, 56, 60, 64 (для Roman и Sans Serif)

n = 0, 21, 42 (для других гарнитур)

- Отменить **ESC X** можно **ESC P, M, g, p, !, @**.

- 5.1. С помощью встроенного редактора **NC** включить в файл **44444** команду **ESC X[60][21][0]**. Назвать файл **55555_1**.

- 5.2. Распечатать полученный файл с помощью команды **COPY** 2 раза. Один раз в режиме **Draft** , а другой раз в режиме **Roman**.
- 5.3. Изменить предыдущую команду на **ESC X[30][21][0]**. Новый файл назвать **55555_2**.
- 5.4. Распечатать полученный файл с помощью команды **COPY** 2 раза. Один раз в режиме **Draft** , а другой раз в режиме **Roman**.
- 5.5. Изменить предыдущую команду на **ESC X[60][40][0]**. Новый файл назвать **55555_3**.
- 5.6. Изменить предыдущую команду на **ESC X[60][16][0]**. Новый файл назвать **55555_4**.
- 5.7. Распечатать оба файла с помощью команды **COPY** в режиме **Roman**.
- 5.8. Обработать результаты , выявить соответствие значений **pitch** and **point** в команде и на распечатке.

Примечание:

- команду **ESC** набирать на цифровой клавиатуре справа как **[Alt]/27**,
- параметр **[30]** набирать как **[Alt]/30** и т.д.
- параметр **[0]** нельзя набрать как **[Alt]/0**. Для его набора, находясь в редакторе NC, следует ещё раз нажать клавишу **[F4]** и перейти в режим HEX- отображения содержимого байт файла. Только в этом режиме в требуемый байт можно записать 0 с клавиатуры.

6. Изменить левую и правую границы печати, установленные по умолчанию

- 6.1. С помощью встроенного редактора NC включить в файл **44444** команды осуществляющие
 - 6.1.1. инициализацию принтера.....**ESC @**
 - 6.1.2. установку левой границы зоны печати**ESC I [8]**
 - 6.1.3. установку правой границы зоны печати.....**ESC Q [16]**
- 6.2. Распечатать полученный текстовый файл **66666** командой **COPY**.

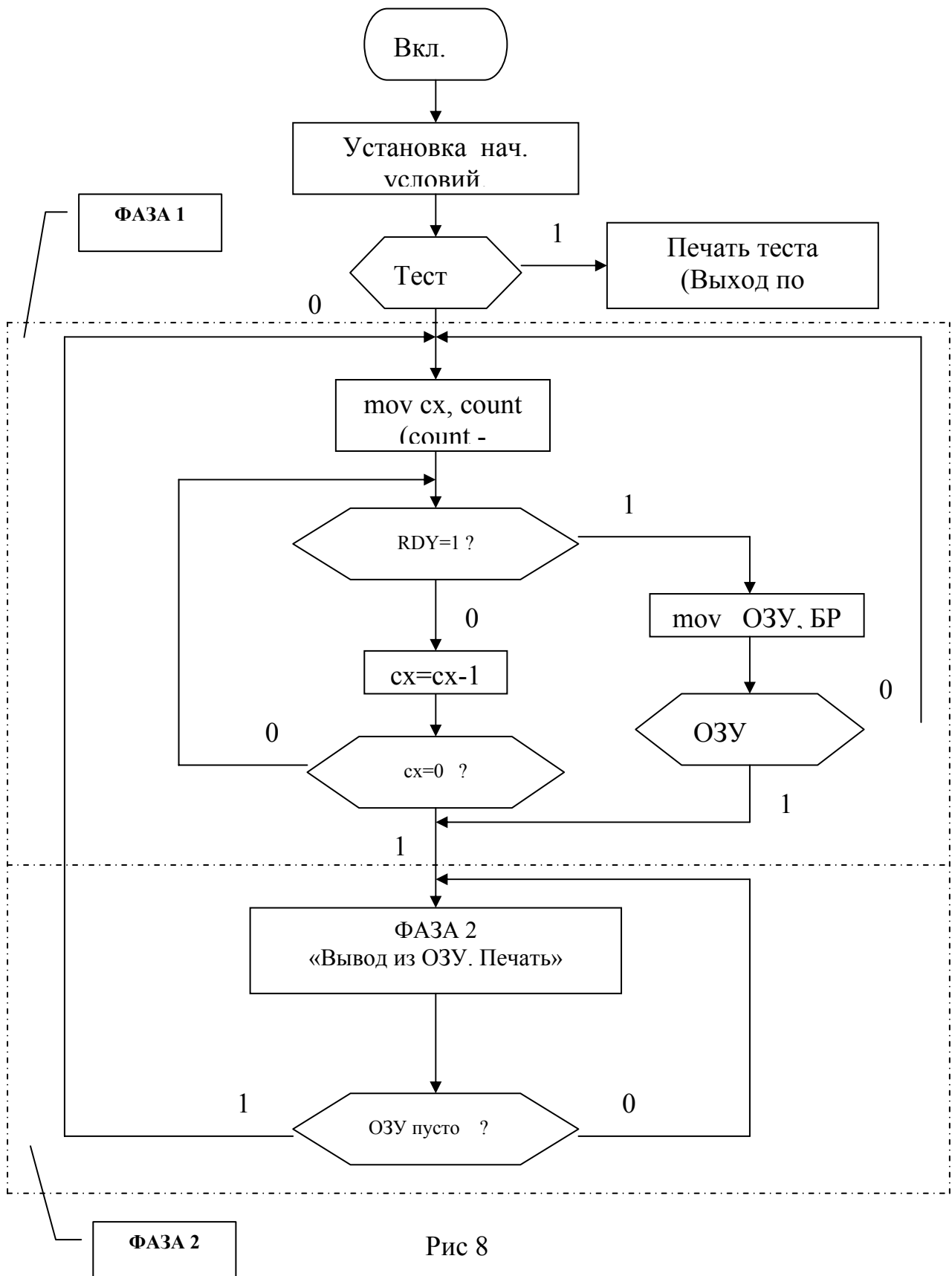


Рис 8

7. Изменить границы печати, установленные по умолчанию.

Для чего:

- 7.1. добавить в исходный файл **44444** команду...**ESC(c40[104][1][104][1]** , которая с указанными параметрами установит верхнюю и нижнюю границы печатного поля равными 1 дюйму;
- 7.2. и команды **ESC I[20]** и **ESC Q[50]**, которые определяют начало левой границы печати с 21 колонки и конец границы печати на 50 колонке включительно. Действие команд отражает рис.9.

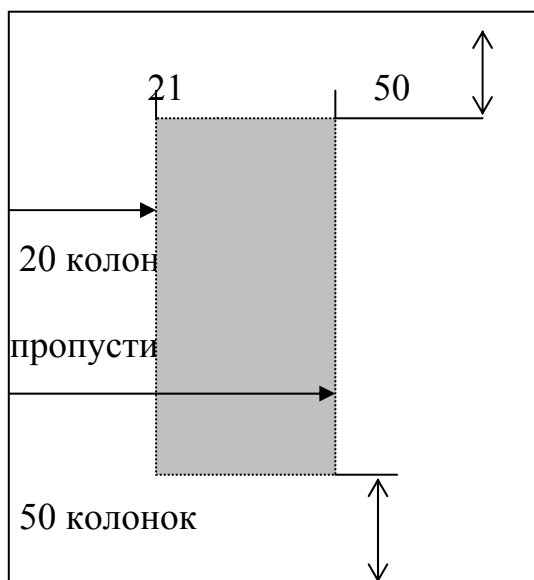


Рис. 9

- 7.3. Распечатать полученный файл **77777** с помощью команды **COPY**.

8. Создать файл для распечатки текста как на рис. 10.

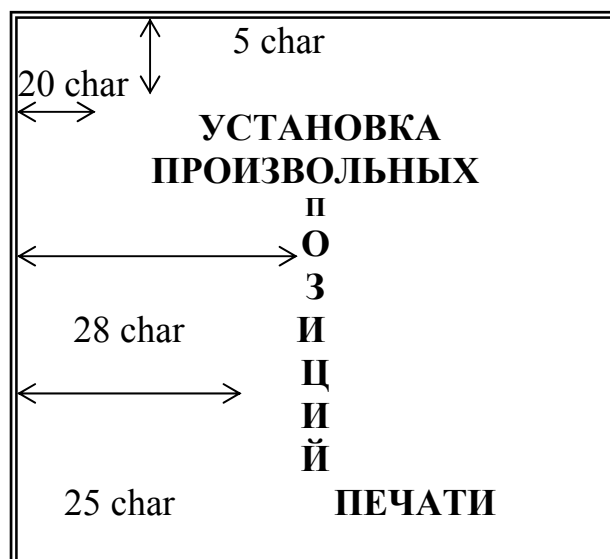


Рис.10

- 8.1. Для форматирования использовать команды установки левой границы позиции печати **ESC I n** . См. приложение 1.

ESC@ [10][10][10][10][10] ESCI[20] Установка произвольных ESCI[25] и [10] о [10] з [10] и [10] ц [10] и [10] й [10] ESCI[25] печати [10][10][10][10][10]

8.2. Распечатать полученный файл **88888** с помощью команды **COPY**.

9. Определить знак пользователя

Пользовательские знаки можно определять посредством команды

ESC&0n1n2d0d1d2data,

где:

- n1-первая литера заменяемая пользователем;
- n2-последняя литера, заменяемая пользователем.
- d0 – левый промежуток литеры,
- d1 – ширина литеры
- d2 – правый промежуток литеры
- data –для каждой колонки матрицы на которой строится литера требуются 3 байта, соответственно 3 рядам иглол.

9.1.Для примера в качестве заменяемой литеры выберем А[65], тогда n1 = 65 и n2 = 65.

9.2.Будем учитывать, что по умолчанию размер литеры равен **10.5** точки. Это означает, что литера строится на матрице состоящей из **21** колонки, причем каждая иголка не может печатать в смежной колонке, а только через одну.

Символьная матрица																						
	№	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	128	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
	64																					
	32																					
	16																					
	8																					
	4																					
	2																					
	1																					
2	128	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
	64																					
	32																					
	16																					
	8																					
	4																					
	2																					
	1																					
3	128	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○
	64																					
	32																					

б а й т	16																			
	8																			
	4																			
	2																			
	1	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○	○

Рис. 11

Для исследования действия такой команды составим файл **99999**, который 10 раз распечатывает на принтере пользовательский знак, который сконструирован на рис. 11. Пользовательский знак в примере представляет четыре черточки состоящие из совокупности точек с кодом ААБ (русские) для одной колонки.

ESC&0AA[0][21][0] ААБ ... ААБ ESC%[1] ААААААААААSDFGHJK ESC%[0]

ААААААААААSDFGHJK [10][10][10][10][10]

Комментарии к командам файла:

ESC&0AA[0][21][0]	ААБ ... АА	ESC%[1]	ААААААААААSDFGHJK
Создать символ на месте А [65](английск.)	21 тройка русских литер А[128], Б[129]	Включить символы пользователя	Символы на печать

ESC%[0]	ААААААААААSDFGHJK	[10][10][10][10][10]
Выключить символы пользователя	Символы на печать	5 раз перевести строку

Примечание: Команда *Определить знак пользователя* [ESC&0n1n2d0d1d2data] используется совместно с командой *Включить/Выключить символы пользователя* [ESC%n]

9.3. Распечатать полученный файл с помощью команды COPY 2 раза: один раз для шрифта **Roman**, другой раз для шрифта **Draft** .

10. Исследование команды битового образа (Bit image)

Команда имеет вид [ESC * m n1 n2 data] , где

n = n1+n2*256;

n: общее число битовых колонок;

data: общее количество байт данных в команде

$data = (n1+n2*256)*t$; где t = число байт на колонку.

Параметр m определяет горизонтальную и вертикальную плотности печати в dpi и соответствующее им количество иголок и количество байт на 1 колонку (t).

m	Horizontal density (dpi)	Vertical density (dpi)	Pins	Adjacent dots print	t
0	60	60	8	Yes	1
1	120	60	8	Yes	1
2	120	60	8	No	1
3	240	60	8	No	1
4	80	60	8	Yes	1
6	90	60	8	Yes	1
32	60	180	24	Yes	3
33	120	180	24	Yes	3
38	90	180	24	Yes	3
39	180	180	24	Yes	3
40	360	180	24	No	3

Примечание: Adjacent – смежные (точки).

10.1. Для исследования действия такой команды составим файлы **bim_1**; **bim_2**; и **bim_3** в соответствии с пунктами 1, 2, 3 рис.13, которые печатают точки с помощью игл печатающей головки, обозначенных на рисунке 12.

1 б а й т	128	○
	64	○
	32	○
	16	○
	8	○
	4	○
	2	○
1	○	
2 б а й т	128	○
	64	○
	32	○
	16	○
	8	○
	4	○
	2	○
1	○	
3 б а й	128	○
	64	○
	32	○
	16	○
	8	○

Рис. 12

1	ESC*[0][30][0]		[1].....[1]	[10].....[10]					
			30 штук	5 штук					
2	ESC*[0][30][0]		[1]....[1]	ESC*[39][30][0]	[1].....[1]	[10].....[10]			
			30 штук		90 штук	5 штук			
3	ESC*[39][24][0]								
	[1][0][0]	[2][0][0]	[4][0][0]	[8][0][0]	[16][0][0]	[32][0][0]	[64][0][0]	[128][0][0]	
	[0][1][0]	[0][2][0]	[0][4][0]	[0][8][0]	[0][16][0]	[0][32][0]	[0][64][0]	[0][128][0]	
	[0][0][1]	[0][0][2]	[0][0][4]	[0][0][8]	[0][0][16]	[0][0][32]	[0][0][64]	[0][0][128]	[10]...[10]

Рис. 13

10.2. Распечатать полученный файл с помощью команды **COPY**. Результаты поместить в отчет.

11. Исследование команд растровой графики (Graphics)

Имеется 2 команды. Первая команда: **ESC(G10[1]** – включает режим растровой графики. Вторая команда **ESC.c v h m n1 n2 data** осуществляет печать различного типа линий. В последней команде:

c = 0: полный графический режим

1: режим сжатой графики.

v = 10,20- вертикальная плотность в единицах 3600/v dpi;

h = 10,20- горизонтальная плотность в единицах 3600/h dpi;

- не разрешена комбинация v=10 и h=20

m: количество печатающих иголок $1 \leq m \leq 24$;

n: - количество горизонтальных точек $n = n1 + n2 * 256$

n задаёт количество точек, которое определяет длину распечатываемой линии.

Например, при $hdpi = 180$ dpi, параметр $n = 7$ определяет длину распечатываемого отрезка равной 1мм по формуле

$$L = (7 * 25.4) / 180 = 1 \text{ мм}$$

Выбор количества литер data определяется из расчета 1 литера на 8 точек, т.е. data = ближайшему целому числу от $n/8$ в сторону увеличения. Так, если $n = 7$, то data = 1 литере (например [255]), если $n = 32$. то data = 4 литерам (например [255][255][255][255]) /

Ориентировочно можно считать, что для $hdpi = 180$ dpi необходима 1 литера на 1мм или 23 литеры на 1 дюйм.

11.1. Распечатать отдельные точки [m=1]:

11.1.1. Составить файл **rg_1: Esc.[0][20][20][1][1][0]a[10][10][10][10][10]** и скопировать его на принтер.

11.1.2. Составить файл **rg_2: Esc.[0][20][20][1][2][0]a[10][10][10][10][10]** и скопировать его на принтер.

11.2. Распечатать вертикальный отрезок [m=24]:

11.2.1. Составить файл **rg_3** и скопировать его на принтер.

rg_3	Esc.[0][20][20][24][1][0]	[255].....[255]	[10][10][10][10][10]
		24 штуки	5 штук

11.3. Распечатать вертикальную линию с началом в точке с координатами [40,60]мм.

Горизонтальную позицию печати установим с помощью команды **ESC \$ n1,n2**. Т.к. вес пользовательской единицы составляет $Hunit = 1/60$ дюйма, то, используя соотношение $L = (n * 25.4) / Hunit$, определим n для $L = 40$ мм при $Hunit = 1/60$ дюйма. Величина $n = n1 + n2 * 256$ должна быть равна $(40 * 60) / 25.4 = 94$. Значит $n1 = 94$, $n2 = 0$.

Аналогично вертикальную позицию печати установим с помощью команды **ESC (V 20 n1,n2**. Т.к. в этом случае вес пользовательской единицы составляет $Vunit = 1/360$ дюйма, то величина $n = n1 + n2 * 256$ должна быть равна $(60 * 360) / 25.4 = 850$. Значит $n1 = 82$, $n2 = 3$.

Вертикальная линия состоит из отрезков максимальная величина которых может быть установлена равной $m = 24$ кванта (по количеству иголок), что при плотности 180 dpi составит величину $l = (24 * 25.4) / 180 = 3.38$ мм.

Каждый раз после распечатки очередного отрезка линии необходимо восстанавливать горизонтальную позицию печати (команда **ESC \$ n1,n2**)и смещать вертикальную позицию печати на величину отрезка (24/180) дюйма или (48/360) дюйма (команда **ESC(v20 [48][0]** т.к. вес пользовательской единицы в командах установки вертикальной позиции печати составляет Vunit=1/360 дюйма.

11.3.1. Составить файл **rg_4**, команды которого включают графический режим и устанавливают начальную позицию печати.

rg_4: ESC(G10[1] ESC(V20[82][3] ESC\$[94][0]

11.3.2. Составить файл **rg_5**, команды которого печатают вертикальный отрезок, восстанавливают горизонтальную позицию печати и смещают вертикальную позицию печати на 48 единиц по 1/360 дюйма.

rg_5: ESC.[0][20][20][24][1][0] [255]....[255] ESC\$[94][0] ESC(v20[48][0]
24 штуки

11.3.3. Скопировать на принтер последовательно файлы:

rg_4 +rg_5 +rg_5 +rg_5 +rg_5

11.3.4. Результаты обработать и внести в отчёт.

Требования к отчёту.

Отчёт должен содержать:

1. Архитектуру принтера, как специализированной ЭВМ.
2. Основные характеристики принтера **LQ+**.
3. Форматы регистров **Centronics** и его временную диаграмму.
4. Алгоритм главной программы принтера.
5. Описание изученных команд принтера.
6. Распечатки результатов работы команд принтера

ПРИЛОЖЕНИЕ 1. Система команд принтера

ASCII

операция с печатающим устройством

ESC @	Инициализация печатающего устройства
ESC EM n	Управление загрузкой бумаги
<i>управление подачей бумаги</i>	
CR	Возврат каретки
FF	Перевод формата
LF	Перевод строки
ESC 0	Выбор 1/8-дюймового интервала между строками
ESC 2	Выбор 1/6-дюймового интервала между строками
ESC 3 n	Выбор n/180-дюймового интервала между

ESC + <i>n</i>	строками Выбор <i>n</i> /360-дюймового интервала между строками
Установка формата бумаги	
ESC (с <i>mn</i>	Определить формат страницы ESC(c40 <i>m1m2n1n2</i> $m = m1 + m2 * 256$ <i>m</i> : Верхняя граница поля печати в единицах 1/360-дюйма $n = n1 + n2 * 256$ <i>n</i> : Нижняя граница поля печати в единицах 1/360-дюйма
ESC C <i>n</i>	Установка длины страницы в строках
ESC C0 <i>n</i>	Установка длины страницы в дюймах
ESC l <i>n</i>	Определить левую границу поля печати <i>n</i> =числу пропущенных колонок от левой границы листа до первой печатаемой колонки
ESC Q <i>n</i>	Определить правую границу поля печати <i>n</i> =числу пропущенных колонок от левой границы листа до последней печатаемой колонки – 1.
Установка позиций печати	
ESC\$ <i>n1n2</i>	Устанавливает абсолютную горизонтальную позицию печати $n = n1 + n2 * 256$ <i>n</i> : Определяет позицию печати от левой границы в единицах 1/60-дюйма по умолчанию
ESC\ <i>n1n2</i>	Устанавливает относительную горизонтальную позицию печати $n = n1 + n2 * 256$ <i>n</i> : Определяет позицию печати относительно текущей в единицах: 1/120 дюйма для черновой печати и 1/180-дюйма для качественной печати по умолчанию
ESC(V <i>nn</i>	Устанавливает абсолютную вертикальную позицию печати ESC(V20 <i>n1n2</i> $n = n1 + n2 * 256$ <i>n</i> : Определяет позицию печати от верхней границы в единицах 1/360-дюйма по умолчанию
ESC(v <i>nn</i>	Устанавливает относительную вертикальную позицию печати ESC(v20 <i>n1n2</i>

$$n = n1 + n2 * 256$$

n: Определяет позицию печати относительно текущей в единицах 1/360 дюйма по умолчанию

ESC J *n* Перемещение бумаги на *n*/180-дюйма

повышение печатного качества

ESC E Выбор режима выделенного шрифта

ESC F Отмена режима выделенного шрифта

ESC G Выбор двухударного режима

ESC H Отмена двухударного режима

ESC S0 Выбор режима верхнего индекса

ESC S1 Выбор режима нижнего индекса

ESC T Отмена режима верхнего/нижнего индекса

обработка текста

ESC SP *n* Выбор межзнакового промежутка

наборы знаков

ESC 4 Выбор режима курсива

ESC 5 Отмена режима курсива

ESC P Выбрать 10.5 точек, 10 cpi

ESC M Выбрать 10.5 точек, 12 cpi

ESC g Выбрать 10.5 точек, 15 cpi

Виды печати

SI 0Fh Установить сжатую печать

DC2 12h Отменить сжатую печать

SO 0Eh Установить печать двойной ширины (1 строка)

DC4 14h Отменить печать двойной ширины (1 строка)

ESC W 1/0 Печать двойной ширины
установить/выключить

ESC w 1/0 Печать двойной высоты
установить/выключить

определяемые пользователем знаки

ESC & *nn* Определение определяемых пользователем знаков

ESC % 1/0 Символы пользователя включить/выключить

Литература

1. User's Guide LQ 100+, EPSON, Nagano, Japan, 1995.
2. Джордейн Р. Справочник программиста персональных компьютеров типа IBM, PC, XT и AT. С.-П.: Финансы и статистика, 1992.