

**K 6418.03**  
OEM

**Графопостроитель  
Robotron-REISS**

**VEB Robotron-Elektronik  
und Zeichentechnik  
Südring 6  
Bad Liebenwerda  
DDR-7950**

819 - 821  
819 - 455  
819 - 661

## **Содержание:**

1. Краткое описание
2. Технические данные
3. Конструкция
  - 3.1. Механическая конструкция
  - 3.1.1. Основные размеры
4. Органы управления
5. Инструкция о пользовании
  - 5.1. Наладка и подключение к управляющей ЭВМ
  - 5.2. Проверка работоспособности, тестовые программы
  - 5.3. Работа с устройством
  - 5.4. Техническое обслуживание и уход
6. Упаковка, хранение и перевозка
7. Указания пользователю
8. Качество защиты

### **1. Краткое описание**

Графопостроитель является графическим периферийным устройством для подключения к малым и микро-ЭВМ. Связь с вышестоящей ЭВМ реализуется с помощью интерфейса С2 или ИРПС. Через этот интерфейс ведущая ЭВМ графопостроителю передает команды с соответствующими параметрами. В обратном направлении принимается информация о состоянии. Обработка переданных команд осуществляется с помощью 8-битовой микро-ЭВМ, которая содержится в графопостроителе.

Максимальным форматом является формат А3, формат можно уменьшать любым образом. Бумага на поверхности графопостроения крепится электростатически. В качестве ручек используются специальные фломастеры, изготовленные предприятием „ФЕБ Маркант“ в сингвице. Ручки поставляются в четырех цветах.

### **2. Технические данные**

Формат бумаги	A3 (297 мм × 420 мм)
Рабочая поверхность	ось X — 370 мм ось Y — 270 мм электростатическое ось X и ось Y каждая с помощью 4-фазового шагового двигателя и канатных систем
Крепление бумаги	
Привод	
Наименьший программируемый шаг	0,1 мм
Воспроизводимость	0,2 мм
Точность	0,2 % (меньше 0,2 мм)
Скорость графопостроения	24 см/сек, 12 см/сек
Скорость точек	максимально 10 сек
Интерфейс	ИРПС (передатчик активен), ИРПС (пассивный), V.24
Код передачи	КОИ-8
Протокол передачи	ИРПС — программный протокол DC 1/DC 3 (XON/XOFF) — программный протокол DC 1/DC 3 (XON/XOFF) — технический протокол DTR
Набор символов	КОИ-8 и пользовательский набор символов на схеме, вставляемой на соединитель емкостью 2 КБ
Идентификация графопостроителя	OI (Output identification) (после получения команды OI графопостроитель передает свое имя и номер версии программного обеспечения ROM, например, СМ 6415 V 8.31)
Команды графопостроения	— для представления векторов — для деления координат — для генерации дуги окружности — для представления текста — для опускания пишущего элемента — для управляющей информации

Способ написания команд	в команде допускаются прописные и строчные буквы (например, PU или ru)
Программное обеспечение на ROM	
Соединитель емкостью 2 КБ для схемы с пользовательским набором символов (тип схемы U 2716)	
Автоматическая индикация ошибок	<ul style="list-style-type: none"> <li>— неправильная команда</li> <li>— неправильный параметр</li> <li>— неправильный символ</li> <li>— неправильное окно (ошибки четности не индицируются)</li> </ul>
пишущий элемент	фломастер, движущийся по координатам
Движение пишущего элемента	с помощью программы или вручную с помощью клавишей
Опускание пишущего элемента	с помощью программы или вручную с помощью клавишей поля управления (клавиши поля управления имеют приоритет перед командой PEN)
Буфер ввода	512 Байтов
Напряжение сети	220 В ±10 %
Частота	50 Гц
Потребляемая мощность	110 ВА
Предохранители сети	2×0,63 АТ
Температура окружающей среды	5°...40 °C
Размеры	510 мм×520 мм×140 мм
Вес	16 кг
Степень защиты	IP 20 согласно СТ СЭВ 778-77
Класс защиты	I

### 3. Конструкция

#### 3.1. Механическая конструкция

Алюминиевая плита толщины 6 мм образует стойку. На этой плите находятся конструкция рамы и все части движущегося механизма. На верхней стороне находится рабочая поверхность. На нижней стороне находятся трансформаторы, электропитание и управление шаговыми двигателями. На задней стороне конструкции рамы находится шина для штекера сети, для подключений интерфейсов и для предохранителей.



Рис. 1: Задний вид графопостроителя

Под полем управления, в передней части рамы, находится электроника ЭВМ (смотри рис. 6).

#### 3.1.1. Основные размеры

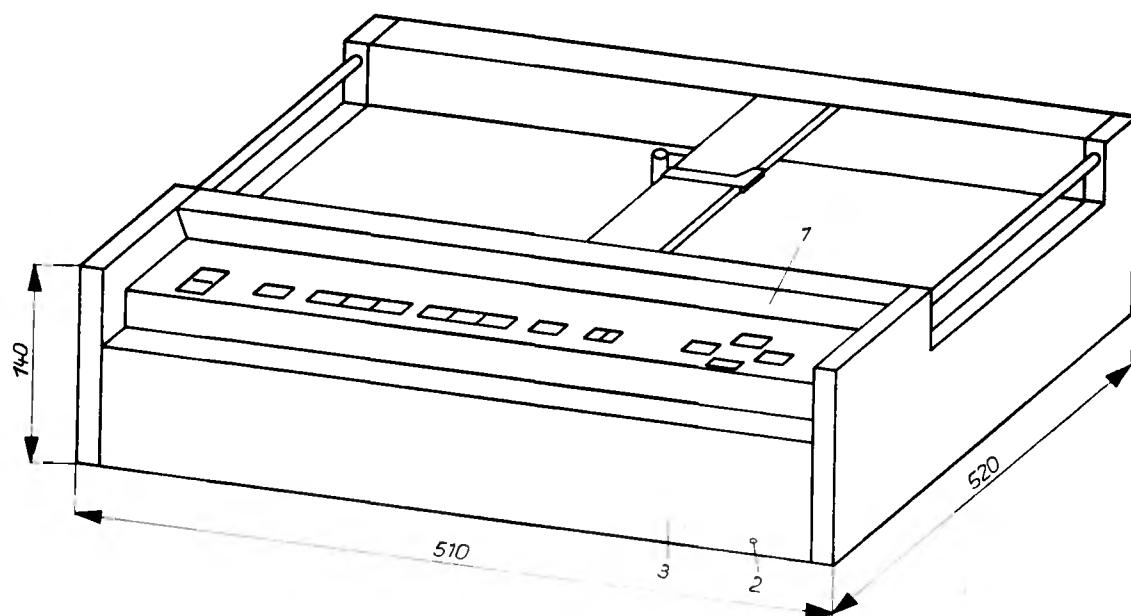


Рис. 2: Основные размеры графопостроителя

### 4. Органы управления

- |                             |   |
|-----------------------------|---|
| Включатель сети:            | указывает готовность к работе   |
| Клавиша сброса:             | для рестарта ЭВМ и для начальной установки ручки (установка памяти местонахождения) |
| Нажата клавиша Paper:       | включено электростатическое крепление бумаги, начата обработка программы            |
| Нажата клавиша Pen:         | ручка находится в отпущенном состоянии  |
| Нажата клавиша Pen:<br>P 1: | ручка находится в поднятом состоянии  |
| P 2:                        | клавиша для новой установки левого нижнего предела окна                             |
|                             | клавиша для новой установки правого верхнего предела окна                           |

Window:

клавиша для нарисования текущих пределов области (окна)

Нажата клавиша View:

прерывается обработка программы, ручка передвигается в верхний правый угол окна  
ручка возвращается в исходное положение, обработка программы продолжается

Отжата клавиша View:

Индикация на светодиодах, зеленый цвет: Ready светится мигает

Индикация на светодиодах, красный цвет: Error светится

выходит за пределы области вызваны сервисные программы

мигает

#### 4.1. Приспособление графопостроителя для подключения к ЭВМ

Устройство предусмотрено для подключения к ЭВМ с помощью кабелей интерфейсов ИРПС или С2. Интерфейс передатчика ИРПС можно выбрать активным или пассивным. Выбор осуществляется путем установки следующих перемычек накрутки:

активный режим  
пассивный режим

E3 — E4, E5 — E6  
E4 — E6

##### 4.1.1. Правило установки переключателей типа ДИП

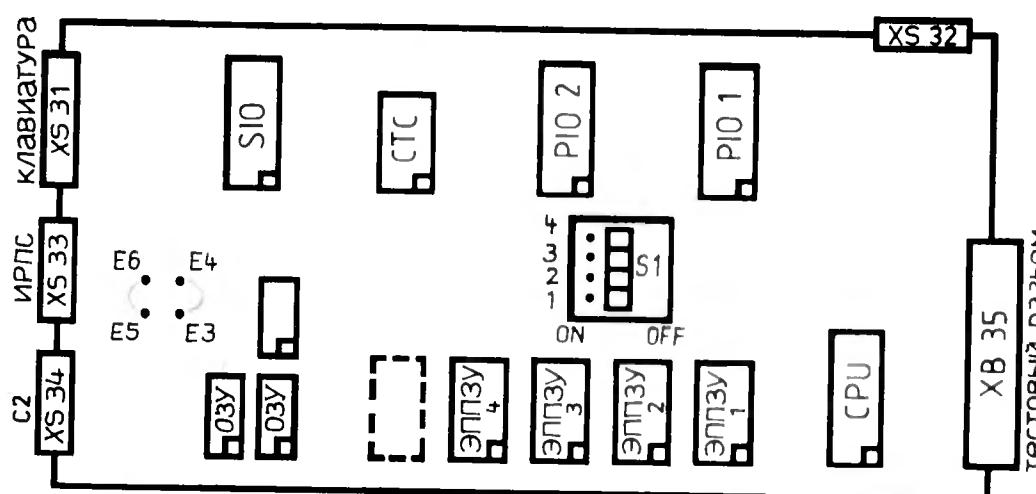


Рис. 3: Плата ЭВМ, расположение элементов

Функциональная таблица S 1

переключатель скорости графопостроения и скорости передачи данных  
(стандартная установка: скорость 240 мм/сек и 9600 бод)

Функциональная таблица

S 1.1	S 1.2	S 1.3	S 1.4	Скорость передачи/графопостроения
ВКЛ	ВКЛ	ВКЛ		9600
ВКЛ	ВКЛ	ВЫКЛ		4800
ВКЛ	ВЫКЛ	ВКЛ		2400
ВКЛ	ВЫКЛ	ВЫКЛ		1200
ВЫКЛ	ВКЛ	ВКЛ		600
ВЫКЛ	ВКЛ	ВЫКЛ		300
ВЫКЛ	ВЫКЛ	ВКЛ		150
ВЫКЛ	ВЫКЛ	ВЫКЛ		75
			ВКЛ	240 мм/сек
			ВЫКЛ	100 мм/сек

##### 4.1.2. Снятие платы ЭВМ

Снятие платы ЭВМ осуществляется следующим образом (смотри рис. 2 и 4):

- снятие заслона (1) путем вытягивания наверх (штепельные запоры на правой и левой стороне)
- вывинчивание 4 винтов (2)
- снятие передней части корпуса (3)
- снятие фиксации платы ЭВМ (4)
- снятие боковых разъемов (5)
- вытягивание платы ЭВМ

После установки переключателей типа ДИП для выбора скоростей передачи и графопостроения плата ЭВМ устанавливается в обратной последовательности, после чего корпус графопостроителя закрывается.

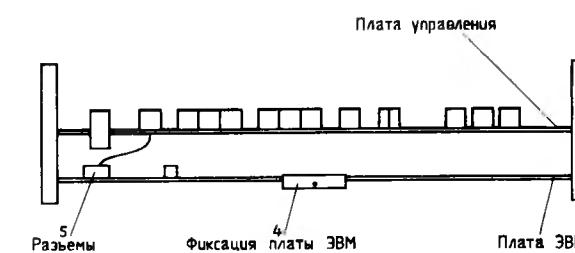


Рис. 4: Графопостроитель, передняя сторона без корпуса

##### Группа клавишей

Клавиши для ручного позиционирования ручки.

Клавиши нажаты: Ручка в соответствии с указанным для данной клавиши направлением движется в направлении X или Y. При кратковременном на-

жатии выполняются отдельные шаги, при длительном нажатии ( $= 0,5$  сек) скорость шагов медленно ускоряется до примерно 4 см/сек.

#### NMI

Клавиша нажата: Прерывание программы, вызов сервисных программ.  
Клавиша установлена в покрытом месте.

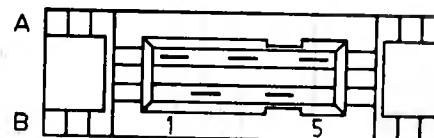


Рис. 5: Интерфейс ИРПС

A 1 — SD —	(данные передачи)
B 1 — SD +	(данные передачи)
A 3 — ED +	(данные приема)
B 4 — ED —	(данные приема)
A 5 — S	(экран)

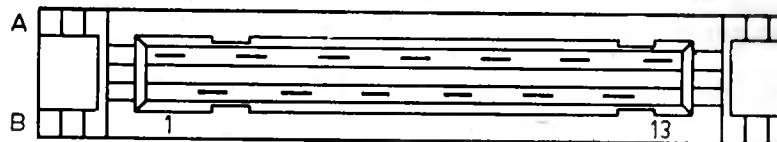


Рис. 6: Интерфейс С 2

A 1 — V 102	(рабочая земля)
B 2 — V 101	(защитная земля, экран)
A 3 — V 103	(данные передачи)
B 4 — V 104	(данные приема)
B 8 — V 108	(DTR)

#### 5.2. Проверка работоспособности, тестовая программа 1

После включения устройства ручка автоматически должна передвигаться в верхний правый угол. После этого с помощью клавиш движение ручки свободно можно передвигать ручку в пределах поля графопостроения. Для проверки динамического поведения движения можно вызвать следующие внутренние тестовые программы:

- тестовый контур: контурный треугольник (клавиша Р 1)
- тестовый контур: круг (клавиша Window)
- набор символов с 3 разными размерами шрифта  
(клавиша движения ручки  $+x$  маленький шрифт,  $-x$  средний шрифт,  $+y$  большой шрифт).

С целью вызова тестовых программ нужно нажать клавишу NMI, которая покрыто находится под заслоном ((1) в рис. 2). Мигание индикации Error указывает на установленный режим теста. Путем нажатия клавиши окна Р 1 вызывается тестовая программа „контурный треугольник”, путем нажатия

клавишей движения ручки  $-x$  или  $+x$  или  $+y$  вызывается „набор символов” с разными размерами шрифта. После нажатия клавиши вызывается программа построения круга.

Прерывание текущей тестовой программы возможно путем повторного нажатия клавиши NMI. Безошибочное продолжение работы в этом случае возможно только после нажатия клавиши Reset.

Выход из режима теста после окончания вызванной тестовой программы можно осуществлять с помощью нажатия клавиши движения ручки  $-y$ .

#### 5.3. Работа с устройством

С целью установки или снятия бумаги целесообразно передвигать ручку в верхний правый угол. Это достигается путем нажатия клавиши View или клавиши Reset. Установку бумаги на поверхность графопостроения следует осуществлять чисто и параллельно к осям. Путем нажатия клавиши Paper бумага фиксируется на поверхности графопостроения. Возможные пузыри выдавливаются в направлении к краю бумаги рукой, щеткой или другими пригодными инструментами.

После включения электростатической фиксации бумаги графопостроитель готов к выполнению программы. Это состояние указывается светящейся зеленой лампой готовности.

Если процесс графопостроения по правилам закончился командой NR, то начинать этот процесс заново можно только после выключения и повторного включения фиксации бумаги.

При использовании формата бумаги, меньше формата А 3, рекомендуется ограничить окно графопостроения. Если светится лампа „Ready”, то для этой цели сначала нажимается клавиша Reset. Ручка с помощью клавишей движения ручки передвигается в нижний левый угол бумаги, после чего нажимается клавиша Р 1.

После этого ручка передвигается в верхний правый угол и нажимается Р 2. Тогда ручку можно передвигать только в пределах заданного окна.

Снятие бумаги осуществляется при выключенном фиксации бумаги и нажатой клавише View. Ручка в этом случае находится в верхнем правом угле окна. Из-за имеющегося электростатического заряда один угол бумаги легко поднимается, после чего бумагу можно снять наверх.

Снятие бумаги путем отодвижения в сторону почти невозможно без повреждения бумаги, особенно при сухом воздухе и бумаге малой или средней толщины.

#### 5.4. Техническое обслуживание и уход

Устройство следует защищать от воздействия пыли. При необходимости, но позже всего после 6 недел, круглые палки (ходовой рельс для каретки x) нужно чистить чистой тряпкой. Использование кисти для чистки не рекомендуется.

Все подшипники промазаны на всю продолжительность службы, техническое обслуживание подшипников не требуется.

Вытекающую жидкость ручки немедленно нужно удалить от поверхности графопостроения спиртом или бензином.

Если время простоя превышает 10 минут, то нужно вывинчивать ручку и с помощью крышки защищать ее от высыхания.

Замена поврежденного каната x или y осуществляется сервисной службой.

Канатная система является сложной, кроме того нужно соблюдать натяжение канатов.

## 6. Упаковка, хранение и перевозка

Для графопостроителя СМ 6415 действуют пункты 1.8. и 1.9. СТ СЭВ 3185-81. Перевозка должна осуществляться в упаковке, которая предусмотрена изготавителем (при перевозке морским транспортом только в контейнерах). При этом механические нагрузки и климатические условия должны находиться в пределах норм, указанных в пункте 4.2. После выпуска из завода время перевозки не должно превышать 3 месяца. Во время перевозки действуют следующие предельные значения:

Механические условия:

Пиковое ускорение = 15 г  
Длительность импульса: = 10 мсек

Климатические условия:

Температура: —50 °C ... +50 °C  
Относительная влажность воздуха: = 95 % при 30 °C  
Давление воздуха: 85 ... 107 кПа

При загрузке, перевозке и разгрузке нужно выполнить требования, указанные на предупредительном щите/знаке на упаковке.

Хранение устройства разрешается только в упаковке. Время хранения не должно превышать 9 месяцев. Имеют место следующие условия хранения:

Температура: 5 °C ... 35 °C  
Относительная влажность воздуха: макс. 85 % при 30 °C

## 7. Указания пользователю

— Следующие условия окружающей среды считаются нормальными условиями эксплуатации:

Температура окружающей среды: +5 °C ... +40 °C  
Относительная влажность воздуха: 40 ... 90 % при 30 °C  
Давление воздуха: 84 ... 107 кПа

Воздух не должен содержать агрессивные газы.

— Для обеспечения названной надежности эксплуатации обслуживающий персонал тщательно должен соблюдать определения, указания и примечания эксплуатационной документации.

— Устройство графопостроителя СМ 6415 почти не требует ухода. Рекомендуется ежедневная чистка поверхности графопостроения. (Внимание! Использовать только неедкие средства для чистки, которые не содержат алкоголя!)

— Категорически запрещается:

— открывать устройство и проводить ремонтные работы при включенном электропитании  
— передать документации и программы другим лицам.

## 8. Качество защиты

Устройство соответствует требованиям ТГЛ 14283 и СТ СЭВ 3743-82. Дополнительные мероприятия по качеству защиты не требуются.

СМ 6415

(К 6418.031)

Техническое описание

## **Содержание:**

- 0. Введение
- 1. Описание интерфейса
- 2. Описание языка
  - 2.1. Синтаксис
  - 2.2. Формат параметров
  - 2.3. Список команд
    - 2.3.1. Команды для представления векторов
    - 2.3.2. Команды для представления текста
    - 2.3.3. Команды для генерации круга
    - 2.3.4. Управляющие команды
  - 2.4. Таблица представляемых символов
    - 2.4.1. Набор латинских символов
    - 2.4.2. Набор символов кириллицы
  - 2.5. Таблица стандартных значений
  - 2.6. Индикация ошибок
- 3. Описание программного обеспечения
  - 3.1. Построение платы ЭВМ и разделение памяти
    - 3.1.1. Разделение отдельных программ операционной системы
  - 3.2. Элементы управления графопостроителем СМ 6415
  - 3.3. Описание составных частей операционной системы
    - 3.3.1. Построение системы прерываний
    - 3.3.2. Память местонахождения
    - 3.3.3. Обслуживание интерфейса
    - 3.3.4. Алгоритм вектора
    - 3.3.5. Представление шрифта
    - 3.3.6. Алгоритм круга/дуги окружности
  - 3.4. Примечания к отдельным командам
    - 3.4.1. Команды для представления векторов PA и PR
    - 3.4.2. Команды ручки PD и PU
    - 3.4.3. Команды для типа штриха LT
    - 3.4.4. Дополнительные команды для деления осей координат TL, XT и YT
    - 3.4.5. Команды для генерации круга CI, AA и AR

## **0. Введение**

Графопостроитель СМ 6415 состоит из следующих узлов:

- вставка
- рама
- рельс типа
- пульт управления
- управление двигателем 1
- управление двигателем 2
- ЭВМ
- электропитание

Вставка образует основной элемент для механической части устройства. На нижней стороне алюминевой плиты толщины 6 мм монтированы следующие узлы:

- трансформаторы
- шаговые двигатели
- держатели роликов с роликами для канатной системы
- печатные платы, управление двигателем 1 и 2, электропитание.

На передней стороне монтирована поверхность графопостроения. Вставка на раме крепляется с помощью углов из листовой стали. На обратной стороне рамы монтируется рельс типа с подключениями интерфейсов, с подключением сети и с элементами предохранителей.

Печатные платы, ЭВМ и пульт управления креплены на раме в передней части устройства.

Привод ручек графопостроения осуществляется двумя шаговыми двигателями с помощью 2 канатов (CNT производства VEB Catgut).

Стороны устройства закрываются 2 частями, изготовленными литьем под давлением. Все остальные стороны и печатная плата пульта управления закрываются частями из листовой стали, покрытыми черным лаком. Обозначение устройства и символы для обслуживания клавиш печатной платы пульта управления напечатаны белой краской.

Снизу устройство закрывается с помощью алюминиевой плиты, которая содержит отверстия для вентиляции и пластмассовые ножки.

## **1. Описание интерфейса**

Для графопостроителя СМ 6415 предусмотрен следующий режим работы: Графическое представление с использованием ниже описанного объема языка с высокой скоростью и использование внутреннего управления ускорением и торможением.

Представление векторов осуществляется только в пределах действительного окна. Если точка назначения находится за пределами окна, то определяется точка пересечения с краем окна и вектор представляется только до края окна.

Виртуальное местонахождение пишущего элемента управляет внутренней программой.

Если исходная точка и точка назначения вектора находятся за пределами окна, то представляется только часть вектора, которая пересекает окно (Clipping).

Виртуальное местонахождение пишущего элемента за пределами действительного окна указывается с помощью светящейся красной лампы Error.

## 1.1. Протокол передачи

### 1.1.1. Протокол передачи — программный протокол (XON/XOFF)

Протокол передачи использует стандартные символы управления устройством DC 1, DC 3, DC 4 и ENQ, которые имеют следующее значение:

DC 1 — соответствует символу 11 hexagonal кода ASCII  
— указывает на пустой буфер

— передается при переходе из режима „офф-лайн“ в „он-лайн“ или после DC 3 буфер опять освободился

DC 3 — соответствует символу 13 H кода ASCII

— сигнализирует заполненный буфер ввода и таким образом прерывание передачи вышестоящей ЭВМ

— первый раз передается, если в буфере находится 500 байтов; после этого каждая передача квитируется сигналом DC 3, причем до состояния заполнения буфера 512 передается без потери данных. Если передача продолжается, то первые символы буфера ввода переписываются новыми данными.

— передается после включения сети или RESET

DC 4 — соответствует символу 14 H кода ASCII

— указывает на состояние ошибки в графопостроителе и требует опроса байта состояния;  
— передается, если бит состояния ошибки изменяется из 0 в 1; если еще возникает ошибка, то перед опросом состояния DC 4 больше не передается.

ENQ — соответствует символу 05 H кода ASCII

— Enquiry (запрос) является запросом ЭВМ к графопостроителю для передачи байта состояния. При этом команда опроса состояния (ENQ) не принимается в буфер ввода, потому что она сразу распознается и обрабатывается.

Для байта состояния действует следующее:

бит 0 = 1 — ошибочная команда

бит 1 = 1 — параметр находится за пределами числового диапазона

бит 2 = 1 — параметр находится за пределами действительного размера окна

бит 4 = 0 — всегда 0 для различия от DCX

бит 5 = 0 — буфер может принимать данные  
= 1 — буфер заполнен

бит 6 = 0 — графопостроитель находится в режиме „офф-лайн“

бит 7 = 1 — графопостроитель находится в режиме выполнения программы (он-лайн, лампочка сигнализации Ready горит)

Данные передаются в следующем формате:

1 пусковый бит

8 битов данных

1 стоповый бит

Графопостроитель выполняет программу до возникновения ошибочной передачи, после этого включается красная лампа (ERROR) и выключается зеленая лампа (READY).

В случае неисправности технических средств обработка прерывается и это сигнализируется таким же образом. Байт состояния в таком случае не передается. Скорость передачи можно настраивать. Ее можно удваивать по шагам на стандартное значение 9600 бд, начиная от 75 бд.

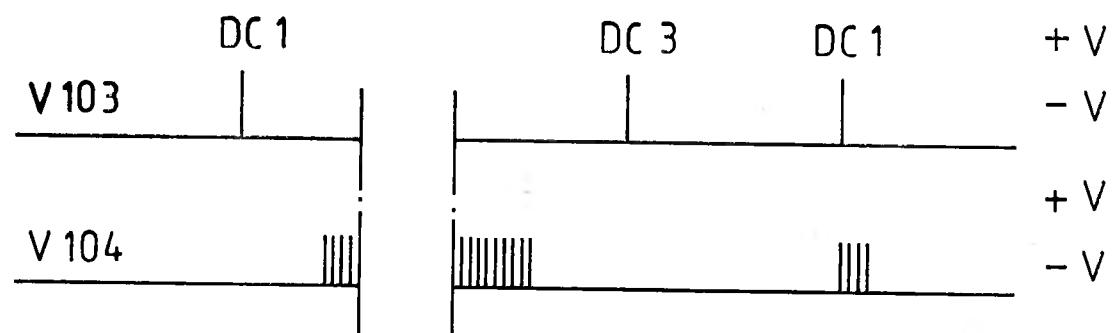
### 1.1.2. Протокол передачи — технический протокол (DTR только при применении интерфейса V.24)

При включении графопостроитель переключает канал сигналов V.108 на Low, т.е. графопостроитель не готов. Если нажимается клавиша Paper канал V.108 переключается на High (активное состояние). Кроме того, сейчас горит лампочка READY, что значит готовность графопостроителя к передаче данных.

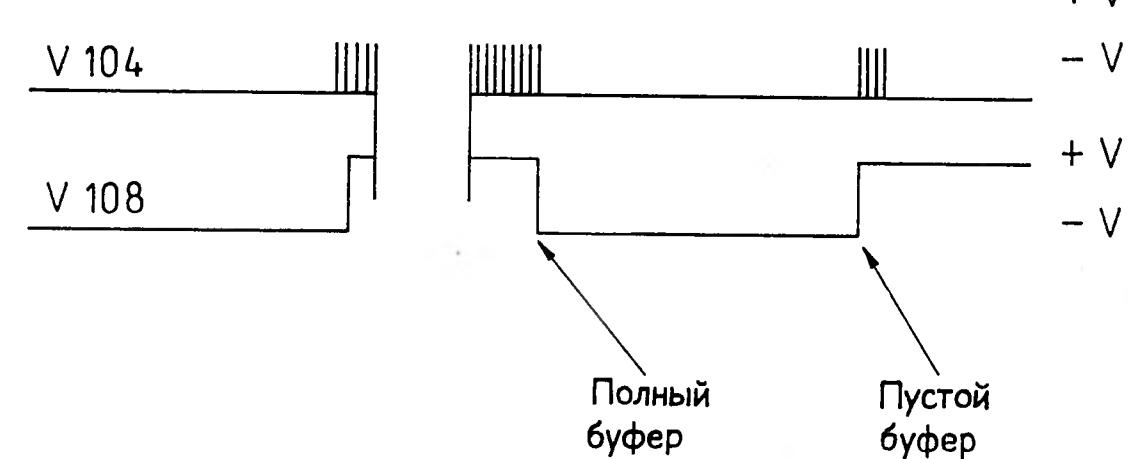
Если в буфере находятся 500 байт канал V.108 переключается на Low, таким образом передача данных больше не возможна, пока не буфер свободен. Нажатие клавиши VIEW тоже переключает канал V.108 на Low, что предотвращает передачу данных и тем самым потерю данных.

### 1.1.3. Временные диаграммы

Программный протокол (дуплексный режим)



Технический протокол





командой „PU“. Функции клавишей „PEN ▼“ и „PEN ▲“ имеют более высокий приоритет.

PU;

Pen up

После распознавания команды „PU“ ручка поднимается до отмены командой „PD“ (смотри примечание к команде PD).

LT number, length;

Line type instruction

Команда выбирает 6 разных типов штрихов

number — номер типа штриха

length — длина штриха в длинах шага для самого длинного штриха последовательности данного типа штриха

- number 1 сплошной штрих, любая длина штриха \_\_\_\_\_  
2 пунктирный штрих (длина штриха : промежуток = 1:1) ——  
3 пунктирный штрих 2:1 промежуток 1 -----  
4 пунктирный штрих 2:1:1 промежуток 1 - - -  
5 штрих-пунктирный штрих 2:1 — точка —.—.  
6 штрих-пунктирный штрих 2:1 — точка: 1 — точка —..—..

SM character;

Symbol mode instruction

Команда вызывает символ текста, который указан в виде параметра в коде КОИ-8, в конце вектора в текущем направлении записи и в текущем размере шрифта. Позиция ручки соответствует начальной позиции при представлении текста.

TL tp, tn;

Tick length instruction

С помощью параметров tp и tn определяется длина маркировки шкалы в длинах шага.

tp — положительная длина (наверх при „XT“ или направо при „YT“)

tn — отрицательная длина (вниз при „XT“ или налево при „YT“)

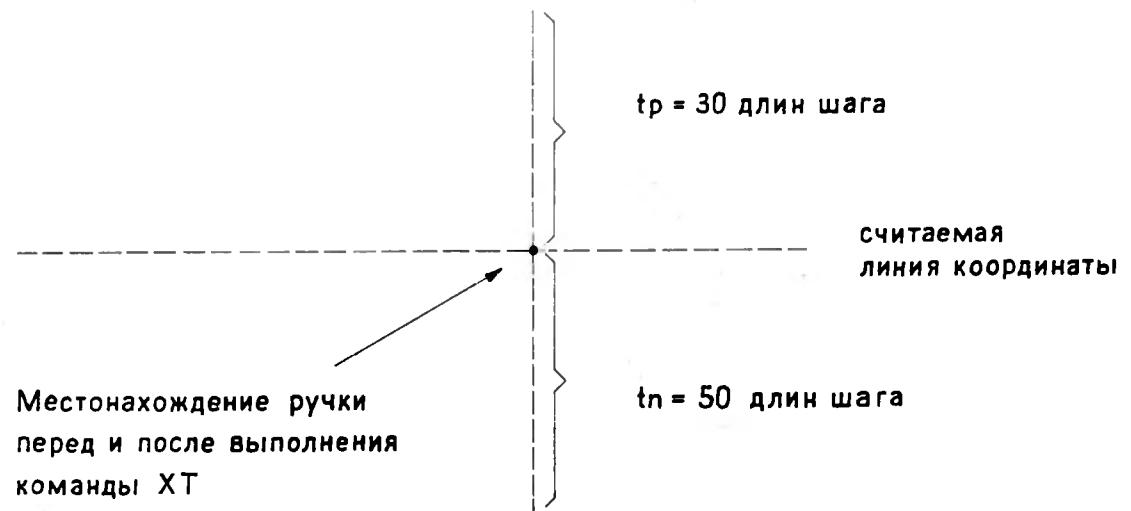
XT;

X-axis tick

Команда вызывает нарисование маркировки шкалы, определяемой с помощью команды TL tp, tn.

Нужно считать, что текущее местонахождение ручки расположено на оси координат.

Пример: Последовательность TL 30, 50; XT; вызывает построение маркировки шкалы следующим образом:

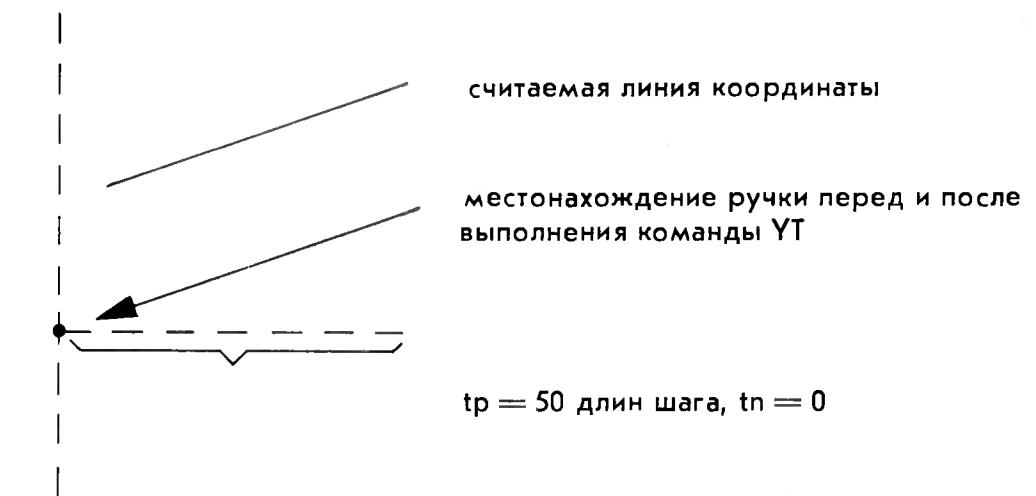


YT;

Y-axis tick

смотри объяснение команды XT

Пример: Последовательность TL 50, 0; YT; вызывает построение маркировки шкалы следующим образом:



### 2.3.2. Команды для представления текста

CP space, lines;

Character plot

Команда вызывает перемещение ручки на количество „space“ пробелов и „lines“ строк из текущего местонахождения ручки. Направление и пройден-

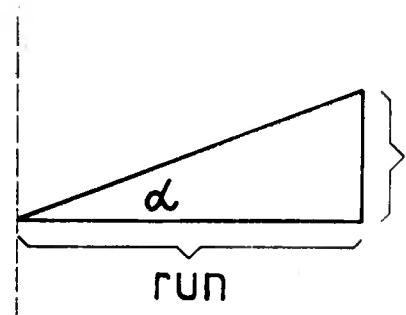
ное расстояние получаются с помощью текущих параметров команд для размера шрифта (SR или SI) и для направления записи (DR или DI). Оба параметра также могут быть отрицательными. Положительные параметры означают: новое направление записи или новая строка.

DR run, rise;  
DI run, rise;

Direction instruction

С помощью команд DR или DI определяется направление записи; обе команды имеют одно и то же действие. Направление записи результирует из отношения rise/run.

Имеют место следующие определения:



- $\text{tg } \alpha = \frac{\text{rise}}{\text{ГУП}} = 1$  горизонтальный шрифт
- $\text{tg } \alpha = \frac{\text{rise}}{\text{ГУП}} > 1$  вертикальный шрифт
- $\text{tg } \alpha = \frac{\text{rise}}{\text{ГУП}} < 0$  генерация зеркального шрифта в вышеозначенном направлении записи

Примеры:

$\text{tg } \alpha = 1$

Жрифт

• Антон

$\text{tg } \alpha = 2$

-----||-----

$\text{tg } \alpha = -0,5$

-----||-----

$\text{tg } \alpha = -2$

-----||-----

Антон

• Антон

НОТНА •  
НОТНА •

Точка указывает на местонахождение ручки в начале команды для представления текста.

SR width, height;  
SI width, height;

Character size instruction

Обе команды определяют размер текстового символа, они имеют одинаковое действие.

width — ширина символа как кратное базовой ширины  
height — высота символа как кратное базовой высоты

Растр базового символа имеет формат 5 (ширина) x 8 (высота) длин шага, из них возможные сокращенные длины равны двум длинам шага. Расстояние базовых строк равно двум длинам шага.

DT terminator

Define terminator instruction

Символ (в коде кой-8), который следует после команды DT, определяет дополнительный к **(ETX)** терминатор для распознавания конца текста. В качестве терминатора не допускаются: пробел, запятая, точка с запятой, **(CR)** возврат каретки, **(LF)** перевод строки и **(NL)** новая строка.

CA n

Designate alternative set

- n = 1 набор символов ASCII и ГОСТ (КОИ-8)
- n = 2 пользовательский набор символов (вставляемая EPROM 5) 20H - 3FH

С помощью этой команды выбирается альтернативный набор символов. По стандарту используется набор латинских символов. Альтернативно можно использовать набор символов согласно ГОСТ 13052-74, в котором вместо латинских строчных букв используются прописные буквы кириллицы (смотри пункт 1.5.2.).

LB ASCII-string **(ETX)**

Label ASCII-string

Каждый символ, который следует после „LB“, представляется согласно своему значению (CR, LF, NL как новая строка). Допускаются все символы кроме **(ETX)** (03 H) и возможного дополнительного терминатора. **(ETX)** в любом случае распознается в качестве конца текста. Непредставляемые символы таблицы кода Sif-1000 принимаются как пробелы. Представление текста начинается в текущем местонахождении ручки, причем используются текущий размер шрифта и текущее направление записи. При достижении края окна автоматически начинается новая строка (если это возможно). В другом случае представление текста прерывается.

SL n;

Slant instruction

Команда SL при представлении текста вызывает наклон символов текста. Параметр может принимать значения от  $-8$  до  $+8$ . Если значение параметра находится за пределами области значений, то используется соответствующее предельное значение ( $-8$  или  $+8$ ). Положительные параметры вызывают наклон направо, отрицательные параметры вызывают наклон налево. Используются следующие углы:

0   $\pm 1$   $\pm 2$   $\pm 3$   $\pm 4$   $\pm 5$   $\pm 6$   $\pm 7$   $\pm 8$
угол в градусах 0   $\pm 9$   $\pm 18$   $\pm 26$   $\pm 34$   $\pm 40$   $\pm 45$   $\pm 49$   $\pm 53$

### 2.3.3. Команды для генерации круга

CI radius;

Circle absolute

Команда „CI“ вызывает представление круга с радиусом R, который задается параметром „radius“. Текущее местонахождение ручки (Xm, Ym) определяет центр круга. Представление круга начинается и кончается вертикалью над местонахождением ручки в точке (Xm, Ym + R) и проходит по часовой стрелке.

AA x,y,phi;  
AR x, y,phi;

Arc absolute instruction  
Arc relative instruction

Команда вызывает рисование дуги окружности, которое начинается в местонахождении ручки с центральным углом фи.

При положительном угле дуги окружности рисуется против часовой стрелки, при задании положительного угла рисование проходит по часовой стрелке. При команде „AA“ параметры x и y обозначают положение центра круга; при команде „AR“ центр круга находится в расстоянии x и y от текущего местонахождения ручки.

Задание „phi“ осуществляется приближенно как частное круга, умноженное фактором:

$$\text{phi} = 4r \sqrt{2} \left( \frac{\pi}{360} \right)$$

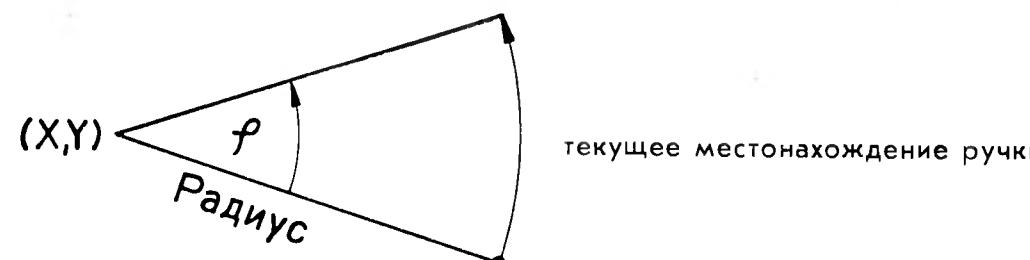
приближенная формула

г задается в длинах шага, фи в градусах.

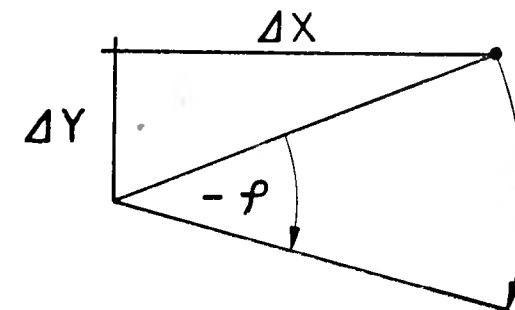
При использовании этой приближенной формулы для центрального угла фи может возникать максимальное отклонение 1,9 градусов.

Примеры команд для генерации дуги окружности:

AA x,y,phi;



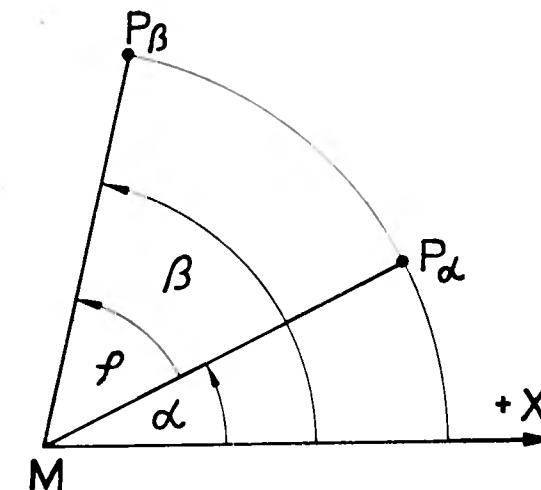
AR x, y,phi;  
(AR -400,-100,-phi;)



Точной формулой для вычисления фи является:

$$y = R \left[ (n\beta - n\alpha) \sin \frac{\pi}{4} + K_{\beta\alpha} \cdot K_{\alpha\beta} \right]$$

R: радиус      R max = 5000 инкрементов



$$\alpha = \alpha_0 + n\alpha \cdot \frac{\pi}{4} \quad \alpha_0, \beta_0 \leq 45^\circ$$

$$\beta = \beta_0 + n\beta \cdot \frac{\pi}{4} \quad n\beta, n\alpha \leq 0$$

если четное:  $K_{\alpha\beta} = \sin \alpha_0$

если нечетное:  $K_{\alpha\beta} = \sin \frac{\pi}{4} - \left( \frac{\pi}{4} - \alpha_0 \right)$

если нечетное:  $K_{\beta\alpha} = \sin \beta_0$

если четное:  $K_{\beta\alpha} = \sin \frac{\pi}{4} - \left( \frac{\pi}{4} - \beta_0 \right)$

Углы и измеряются начиная от рисованной оси абсцисс, они также могут быть отрицательными. В любом случае рисование дуги окружности проходит от позиции Р в позицию Р.

### 2.3.4. Управляющие команды

IN;

Initialize instruction

Команда вызывает инициализацию графопостроителя. Действие является таким же как после включения графопостроителя: нулевая юстировка, стирание буферной памяти, запись стандартных значений.

DF;

Default instruction

Команда вызывает запись стандартных значений в программную память, напр. для размера окна, размера шрифта, направления записи (смотри пункт 4.). Не выполняется нулевая юстировка, буферная память не стирается.

IW x1,y1,x2,y2;

Input window

С помощью команды „IW“ определяются новые координаты окна. Координатная точка x1, y1 описывает нижний левый угол окна, x2, y2 верхний правый угол окна. Ручка передвигается в нижнюю левую позицию x1, y1. Окно расположено параллельно к осям поверхности графопостроения.

VS n;

Select velocity

С помощью параметра n выбирается определенная кривая ускорения и торможения для представления вектора. Ускорение и конечная скорость определяются значениями таблицы для кривой ускорения и торможения. Их нельзя выбирать независимо друг от друга.

n = 1      максимальная скорость оси 12 см/сек или

n = 2      24 см/сек

SW;

Step width

Команда „SW“ задает виртуальную длину шага 0,025 мм; физическая длина шага 0,1 мм сохраняется. (После команды SW все значения параметров

длины разделяются на 4 и после этого используются для представления на графопостроителе.)

Действие команды „SW“ снимается только командами „DF“, „IN“ или с помощью нажатия клавиши сброса.

RO alpha;

Rotate

Команда RO вызывает вращение координатной системы x,y и угла. Базис x,y соответствует базовым осям приводных двигателей графопостроителя: привод x направо, привод y наверх, точка (0,0) находится в нижнем левом углу. Угол вращения округляется на полные шаги градуса. Преобразование осей выполняется в соответствии со следующей формулой:

$$\begin{aligned}x' &= x \cos - y \sin \\y' &= x \sin + y \cos\end{aligned}$$

SS;

Single step instruction

Команда SS вызывает постоянную скорость рисования, т.е. она обходит обычное управление ускорением. С помощью команды SS независимо от предварительно выбранной скорости записи достигается постоянная скорость в направлении оси примерно 6 см/сек.

NR;

No ready instruction

С помощью команды „NR“ заканчивается процесс графопостроения. После этого графопостроитель переходит в состояние „не готов“ (not-ready-state), т.е. внутренняя операционная система реагирует на возможные нажатия клавишей передвижения ручки и клавишей для функций окна ( P1, P2, Window). Ручка поднимается.

OI;

Output identification

С помощью команды „OI“ графопостроитель передает вышестоящей ЭВМ свое имя и номер версии программного обеспечения ROM (например, СМ 6415 V 8.31).

## 2.4. Таблица представляемых символов

### 2.4.1. Набор латинских символов

шестнадцатеричное значение/действие

03H	ETX, конец текста	46H	F	прописная буква
0AH	LF, line feed,	47H	G	прописная буква
	перевод строки	48H	H	прописная буква
0DH	CR, Carriage return,	49H	I	прописная буква
	возврат каретки	4AH	J	прописная буква
1EH	NL, возврат каретки	4BH	K	прописная буква
	с переводом строки	4CH	L	прописная буква
20H	SP, пробел	4DH	M	прописная буква
21H	! восклицательный знак	4EH	N	прописная буква
22H	" кавычки	4FH	O	прописная буква
23H	# двойной крест	50H	P	прописная буква
24H	% знак доллара (\$)	51H	Q	прописная буква
25H	%% процент \$	52H	R	прописная буква
26H	&	53H	S	прописная буква
27H	' апостроф	54H	T	прописная буква
28H	( левая круглая скобка	55H	U	прописная буква
29H	) правая круглая скобка	56H	V	прописная буква
2AH	*	57H	W	прописная буква
2BH	+	58H	X	прописная буква
2CH	,	59H	Y	прописная буква
2DH	-	5AH	Z	прописная буква
2EH	.	5BH	[	квадратная скобка
2FH	/	5CH	\	обратная косая черта
30H	0 цифра 0	5DH	{	квадратная скобка
31H	1 цифра 1	5EH	↑	стрелка вверх
32H	2 цифра 2	5FH	—	под-штрих
33H	3 цифра 3	60H		пробел
34H	4 цифра 4	61H	a	строчная буква
35H	5 цифра 5	62H	b	строчная буква
36H	6 цифра 6	63H	c	строчная буква
37H	7 цифра 7	64H	d	строчная буква
38H	8 цифра 8	65H	e	строчная буква
39H	9 цифра 9	66H	f	строчная буква
3AH	:	67H	g	строчная буква
3BH	:	68H	h	строчная буква
3CH	< левая острая скобка	69H	i	строчная буква
3DH	= знак равенства	6AH	j	строчная буква
3EH	> правая острая скобка	6BH	k	строчная буква
3FH	?	6CH	l	строчная буква
40H	пробел	6DH	m	строчная буква
41H	A прописная буква	6EH	n	строчная буква
42H	B прописная буква	6FH	o	строчная буква
43H	C прописная буква	70H	p	строчная буква
44H	D прописная буква	71H	q	строчная буква
45H	E прописная буква	72H	r	строчная буква

73H	s	строчная буква	7AH	z	строчная буква
74H	t	строчная буква	7BH	j	
75H	u	строчная буква	7CH	l	
76H	v	строчная буква	7DH	m	
77H	w	строчная буква			
78H	x	строчная буква			

#### 2.4.2. Набор символов кириллицы

шестнадца- теричное значение	ГОСТ (прописные буквы)	шестнадца- теричное значение	ГОСТ (строчные буквы)
C0H	Ю	E0H	ю
C1H	А	E1H	а
C2H	Б	E2H	б
C3H	Ц	E3H	ц
C4H	Д	E4H	д
C5H	Е	E5H	е
C6H	Ф	E6H	ф
C7H	Г	E7H	г
C8H	Х	E8H	х
C9H	И	E9H	и
CAH	Й	EAH	й
CBH	К	EBH	к
CCH	Л	ECH	л
CDH	М	EDH	м
CEH	Н	EEH	н
CFH	О	EFH	о
D0H	П	F0H	п
D1H	Я	F1H	я
D2H	Р	F2H	р
D3H	С	F3H	с
D4H	Т	F4H	т
D5H	У	F5H	у
D6H	Ж	F6H	ж
D7H	В	F7H	в
D8H	Ь	F8H	ь
D9H	Ы	F9H	ы
DAH	З	FAH	з
DBH	Ш	FBH	ш
DCH	Э	FCH	э
DDH	Щ	FDH	щ
DEH	Ч	FEH	ч

Область 0H...1FH резервирована для управляющих символов и, кроме в качестве терминатора, не может быть использована для представления текста.

Управляющие функции CR, LF и NL можно использовать в операторах текста.

#### 2.5. Таблица стандартных значений

Режим работы	— интеллигентный графопостроитель
Длина шага	— 0,1 мм
Скорость/ускорение	— в зависимости от установки переключателя
Размер окна	— максимальный формат 370 мм×270 мм
Тип штриха	— полный штрих (для представления вектора) — tn = tp = 15 длин шага
Маркировка шкалы (tick length)	— 6-кратное увеличение для высоты и для ширины (базовый символ 2,4 мм×3,6 мм)
Размер шрифта	— горизонтальный шрифт
Направление шрифта	— <ETX>, также для дополнительного терминаатора
Терминатор цепи	— латинский; кириллицы (КОИ-8)-
Набор символов	— без (n = 0)
Наклонность	

#### 2.6. Индикация ошибок

зеленая лампа	— обрабатывается содержание буферной памяти (интерфейсный режим)
зеленая и красная лампа	— ошибочная команда
	— ошибочный параметр или ошибочное количество параметров
	— переполнение текста (невозможен ввод новой строки)
красная лампа	— переполнение позиций с прерыванием программы; продолжение невозможно; новый пуск с помощью клавиши сброса

### 3. Описание программного обеспечения

#### 3.1. Построение платы ЭВМ и разделение памяти

Для графопостроителя СМ 6415 была разработана специальная печатная плата ЭВМ, которая содержит следующие основные узлы:

1 процессорная микросхема	U 880
1 микросхема СTC	U 857
1 микросхема PIO	U 855
4 ЭППЗУ	U 2716
2 статическое ОЗУ 4 Кбайта	U 214
1 микросхема SIO	U 856
1 гнездо ЭППЗУ	—

Таким образом имеется программная память объема 10 Кбайтов и оперативная память объема 1 Кбайта. Программная память занимает адресное

пространство 0H...1CFFH и оперативная память занимает адресное пространство 3800H...3BFFH.

Область 2000H...2FFFH (гнездо) предусмотрена для набора символов пользователя.

Оперативная память разделена на 3 области.

- область памяти для стековых операций 3800H...38FFH
- рабочая оперативная память для операционной системы 3900H...39FFH
- буферная память для управления интерфейсом 3A00H...3BFFH

Оба варианта интерфейса аппаратными средствами реализованы на плате ЭВМ, выбор желаемого варианта осуществляется с помощью подключения соответствующего кабеля интерфейса. Программное обеспечение для обоих вариантов интерфейса является одинаковым.

### 3.1.1. Разделение отдельных программ операционной системы

- ЭППЗУ 1 адресное пространство 0H...3FFH  
инициализация; разветвление в зависимости от нажатой клавиши; обслуживание программы для ручки управления; генератор векторов (основная программа)  
адресное пространство 400H...7FFF  
классификация длины векторов и таблицы для управления скоростью;  
тестовые программы: фигура, круг, большой, средний и маленький шрифт;  
программа рамы теста; программы обслуживания клавишей
- ЭППЗУ 2 адресное пространство 800H...0BFFH  
представление текста; типы линий; программа интерфейса  
адресное пространство 0C00H...0FFFH  
декодирование команды; декодирование параметров
- ЭППЗУ 3 адресное пространство 1000H...13FFH  
Clipping; Windoing; тестовые программы; вращение осей  
адресное пространство 1400H...17FFF  
генерация круга; дуга окружности; ручка управления; управление скоростью; наклонность; вращение осей
- ЭППЗУ 4 адресное пространство 1800H...1FFFH  
память изображения символов для изображения текста для разных типов шрифта
- ЭППЗУ 5 адресное пространство 2000H...27FFF  
набор символов пользователя

### 3.2. Элементы управления графопостроителем СМ 6415

Графопостроитель СМ 6415 содержит следующие элементы управления:

- выключатель сети
- функциональная клавиша NMI  
для вызова внутренних тестовых программ (построение тестовых фигур)
- функциональная клавиша RESET  
аппаратный сброс для платы ЭВМ
- 2 функциональных клавиши PEN UP и PEN DOWN
- 2 функциональных клавиши координат окна Р1 и Р2 для определения координат окна в левом нижнем и в правом верхнем углу окна

- функциональная клавиша VIEW  
для прерывания текущей программы графопостроения и для передвижения ручки в угол окна Р2
- 4 функциональных клавиши передвижения ручки для медленного и быстрого передвижения ручки в определенную позицию
- функциональная клавиша PAPER  
вкл./выкл. фиксация бумаги
- функциональная клавиша WINDOW  
для рисования текущей области окна  
Лампы индикации „Ready“ и „Error“ управляются программными (Ready) и/или аппаратными (Error) средствами.

### 3.3. Описание составных частей операционной системы

#### 3.3.1. Построение системы прерываний

Из периферийных микросхем на плате ЭВМ прерывания вызываются только микросхемой СТС и микросхемой Р102.

Периферийные микросхемы заняты следующим образом:

- |              |  |
|--------------|--|
| СТС, канал 0 | свободный, не занят  |
| СТС, канал 1 | прерывание, отдельный шаг (частота вывода)                                       |
| СТС, канал 2 | без прерывания   |
| СТС, канал 3 | прерывание, загружать в СТС, каналы 2 и 3<br>следующее значение кривой ускорения |

- |               |                        |
|---------------|------------------------|
| P101, канал А | без прерывания         |
| P101, канал В | без прерывания         |
| P102, канал А | без прерывания         |
| S10, канал А  | передатчик             |
| канал В       | приемник с прерыванием |

Указанная последовательность соответствует приоритету. (Значение отдельных каналов Р10 указывается в описании платы ЭВМ.)

Опрос всех управляющих клавиш осуществляется в режиме периодического опроса в следующей последовательности:

- клавиша VIEW
- запрос интерфейса
- клавиша Р1
- клавиша Р2
- клавиша Window
- клавиша передвижения ручки —Y (JOMY)
- клавиша передвижения ручки +Y (JOPY)
- клавиша передвижения ручки —X (JOMX)
- клавиша передвижения ручки +X (JOPX)

Если нажата клавиша VIEW, то все остальные функции графопостроителя кроме аппаратного сброса блокируются. Периодический опрос начинается, когда вектор полностью обработан. Это значит, что текущий вектор нельзя прерывать даже клавишей „View“; прерывание возможно только с помощью клавиши сброса или с помощью клавиши NMI.

При нажатой клавиши View также не обрабатывается возможный запрос интерфейса, хотя сам запрос принимается (приоритет прерывания 3). Для хранения используется бит 1 регистра IX (бит 1 = 1: имеется запрос, бит 1 = 0: нет запроса интерфейса).

### 3.3.2. Память местонахождения

Графопостроитель СМ 6415 содержит так называемую память местонахождения. Для этой цели поверхность графопостроения графопостроителя разделена в виртуальное поле координат. Точки координат соответствуют длинам шага: максимально 3700 в направлении X и 2700 в направлении Y. Точка координат (0,0) находится в левом нижнем углу, точка (3700,2700) в правом верхнем.

Нулевая юстировка, т.е. установка памяти местонахождения, осуществляется автоматически после включения графопостроителя и в любое время может повторяться нажатием клавиши сброса. Ручка в этом случае переходит в позицию (3700,2700) и ячейки памяти HARDX и HARDY, которые используются для памяти местонахождения, загружаются этими значениями. Если сейчас вызывается рисование вектора, то в начале проверяется, находится ли конечная позиция внутри действительного окна. Если да, то новая конечная позиция записывается в ячейки HARDX и HARDY для соответствующих значений координат. Таким образом актуализируется память местонахождения. После этого рисуется вектор.

Если указанная конечная позиция находится за пределами действительного окна, то вектор рисуется только до края окна. Это имеет место также для векторов, которые начинаются за пределами окна; они изображаются только внутри действительного окна (clipping).

С помощью соответствующей команды (IW x1, y1, x2, y2;) или путем нажатия управляющих клавиш можно уменьшать окно.

Сначала нужно вводить точку координат P1 (нижний левый угол), после этого P2 (правый верхний угол). Если нажата клавиша P1, то в память местонахождения HARDX и HARDY записываются значения 0, при нажатии P2 в GWX и GWY принимаются соответствующие значения для HARDX и HARDY. Каждый из этих указанных новых векторов сейчас проверяется на то, находится ли конечная точка внутри окна. Действия соответствуют вышеописанным.

Из сказанного видно, что новое окно можно выбрать только меньше предыдущего. Увеличение окна возможно только путем сброса. Сброс автоматически вызывает наибольшее возможное окно, которое после этого можно постепенно уменьшать.

### 3.3.3. Обслуживание интерфейса

В случае существования запроса интерфейса проверяется, имеется ли хотя бы одно место в буферной памяти. Если да, то считаются данные канала B SIO и загружаются на следующее свободное место памяти. Если буферная память содержит 500 байтов, то выдается DC 3 (буфер занят). После этого можно передать максимально 12 байтов. Каждая из этих передач подтверждается с помощью выдачи DC 3. Пустой буфер указывается с помощью DC 1 (один раз). Буферная память организована в виде кольцевой памяти емкости 512 байтов. Для управления памятью имеется два указателя, которые указывают „уровень заполнения“ и текущую ячейку при обработке программы. Управляющая программа следит за тем, чтобы до окончания обработки не осуществляется перезапись памяти.

После запроса программы интерфейса из буферной памяти предоставляет следующий байт (передача в регистре A).

Если операционная система выдает запрос на следующий байт буферной памяти, то выполняется декодирование команды. В результате декодирования выполняется или разветвление к специальным частям программы, или (если соответствующая команда не определена) возврат в операционную систему, после чего для декодирования команды используется следующий байт буферной памяти. В случае положительного результата осуществляется декодирование параметров (преобразование 7-битового кода ISO в двоичный код). Если находящееся в буферной памяти число больше 7FFFH, то в результате декодирования параметров записывается это значение. Это имеет место также для отрицательных чисел, которые находятся за пределами области чисел. Разряды после десятичной точки игнорируются.

Если с помощью команды установлена виртуальная длина шага 0,25 мм, то результат декодирования параметров разделяется на 4. В этом случае даже при арифметическом переполнении (установка результата 7FFFH) обеспечивается, что при вызове представления вектора осуществляется прерывание программы; в любом случае конец вектора находился бы за пределами наибольшего возможного окна.

### 3.3.4. Алгоритм вектора

Алгоритм вектора должен выполнять следующие основные функции:

- управление траекторией (соединение двух заданных точек с помощью прямой)
- реализация поведения ускорения и торможения
- разделение векторов по признакам различия (напр. шрифт, большая длина вектора) и соответствующее присвоение категории таблицы для управления ускорением
- тест возможности представления заданного вектора
- управление памятью местонахождения

Алгоритм вектора различает основные и побочные направления движения. Основным направлением движения является то направление оси  $\pm X$  или  $\pm Y$ , в котором при представлении вектора выполняется наибольшее количество шагов. Таким образом различаются движения в 8 октантах. Алгоритм уточняется с помощью рис. 1.

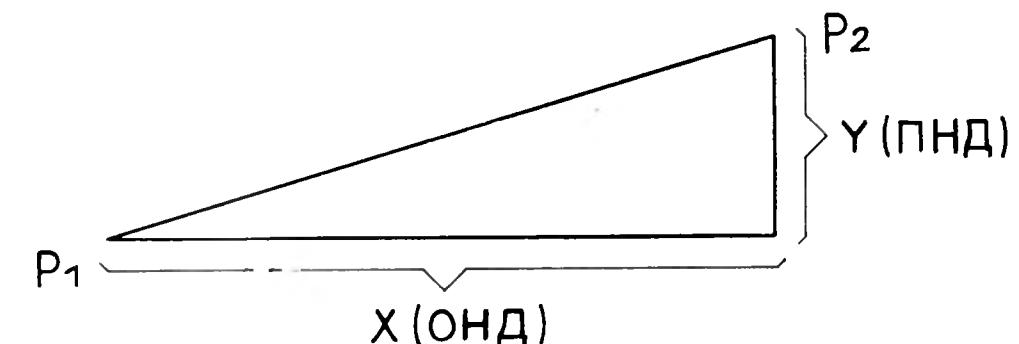


Рис. 1: Интерполяция прямой

Нужно рисовать прямую от P1 в P2. В направлении  $+x$  нужно выполнять наибольшее количество шагов:  $+x$  является основным направлением движения (ОНД),  $+y$  является побочным направлением движения (ПНД). Алгоритм выполняется следующим образом:

$$\begin{aligned}A &= 2y - x \\B &= 2y \\C &= 2y - 2x\end{aligned}$$

Когда  $A = 0$ , то один шаг ОНД и  $A := A + B$

Когда  $A = 0$ , то один шаг ОНД+ПНД и  $A := A + C$

Отсюда видно, что после каждого импульса вывода (или только ОНД или ОНД и ПНД вместе) заново вычисляется рабочая ячейка A. Для следующего вывода шага опять выполняется тест A на 0 или  $= 0$ .

С помощью этого алгоритма безошибочно генерируется направление. С помощью счета импульсов вывода дополнительно нужно определять или проверять длину вектора, при необходимости при последнем выводе шага осуществляется исправление.

Расстояния импульсов для вывода шага изменяются с помощью определенной таблицы, которая оптимально построена экспериментальным образом. Для этой цели используется канал 1 СТС в режиме датчика времени. После истечения постоянной времени вызывается прерывание, которое ведет к выводу следующего правильного импульса шага, определенного с помощью вышеописанного алгоритма. Одновременно в связанные каналы 2 и 3 СТС принимается импульс счета. В этих каналах запоминается количество шагов в основном направлении движения для каждой ступени ускорения. После обработки одной ступени таблицы ускорения в канале 3 СТС осуществляется прохождение через ноль, которое вызывает прерывание и загрузку следующих значений таблицы в каналы 1...3 СТС. При торможении таблица обрабатывается в обратном направлении (т.е. снизу вверх). Таблица обеспечивает равномерный процесс ускорения. Между фазами ускорения и торможения гарантируется минимальное количество шагов с постоянной скоростью, т.е. между обеими фазами нет отрывочного перехода. Несколько при этом обрабатывается таблица ускорения или какая таблица одной категории таблиц используется, зависит от длины вектора и от конечной скорости, выбранной переключателем скорости или соответствующей командой. Конечная скорость и максимальное ускорение описывают соответственно одну категорию.

Алгоритм вектора нужно обрабатывать в реальном масштабе времени; т.е. он имеет абсолютное преимущество. Согласно этой цели построена система прерываний. Для оптимальных по времени вычислений используется второй набор регистров; он используется исключительно алгоритмом вектора. Вывод импульсов ОНД и ПНД и соответствующей информации об октантах осуществляется с помощью микросхемы PIO. Эти импульсы оцениваются электроникой графопостроителя, из них получаются необходимые сигналы для движения шаговых двигателей.

### 3.3.5. Представление шрифта

Представление отдельных символов (прописные буквы, строчные буквы, цифры и специальные символы) определено в памяти представления символов. Начальные адреса памяти представления символов содержатся в

списке адресов. Начальные адреса для символов кода КОИ-8 26H, 40H, 60H, 7EH, 7FH содержат адрес пробела (смотри таблицу представляемых символов текста, которая содержится в описании языка СМ 6415). Здесь имеется место для символов, определенных пользователем. Управляющие символы кода КОИ-8 обходятся.

После для представления символа равно  $5 \times 8$  основным единицам (смотри рис. 2).

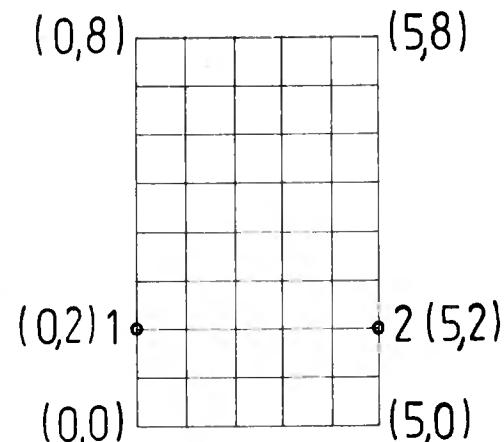


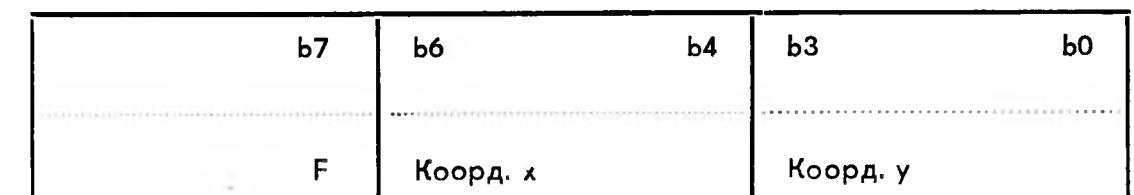
Рис. 2: Растр символов

Представление символа для каждого символа начинается в позиции 1 и кончается в позиции 2. Расстояние к следующему символу уже содержится в растре символа. Представление символов осуществляется в следующем размере:

	ширина	высота	расстояние символов
Прописные буквы	4	6	1
Строчные буквы	3	4 (6)	2
Цифры	3	6	2
Сокращенная длина	—	2	—
Расстояние между строками	—	4	—
Пробел / обратный пробел	5/—5	—	—

Рис. 3: Размер символов

Текстовые символы составляются с помощью коротких векторов. Для каждого вектора в памяти вида символа требуется один байт. Байт памяти вида символа кондирован следующим образом:



b0...b3 координата у в растре символа  
 b4...b6 координата x в растре символа  
 b7 информация для ручки, 0: ручка поднята  
 1: ручка отпущена

Рис. 4: Кодирование байта

Текстовая программа декодирует каждый отдельный байт памяти вида символа. Таким образом получаются информации для представления нового вектора. Для высоты и ширины символа по отношению к базовому символу ( $5 \times 8$  длин шага) используются множители. По стандарту в обоих случаях в качестве множителя используется 3; с помощью команды можно определить другие и даже разные значения для высоты и для ширины. Множитель ширины также используется для расстояния символов, для пробелов и для „возврата на шаг”. Множитель высоты используется для расстояния строк. Расстояние прописных букв меньше чем расстояние строчных букв или цифр. Путем определения постоянного раstra символов имеется возможность использования табуляторных функций и применения „возврата на шаг”, напр. для подчеркивания, обрамления и т.д.

С помощью использования команды CA имеется возможность представления прописных букв набора символов кириллицы (смотри описание языка).

#### Примеры для построения памяти вида символа

##### 1. Прописная буква V

Последовательность байтов: 08H — 86H — 0A2H — 0C6H —  
0C8H — 52H

вызывает следующие действия: передвижение к точке

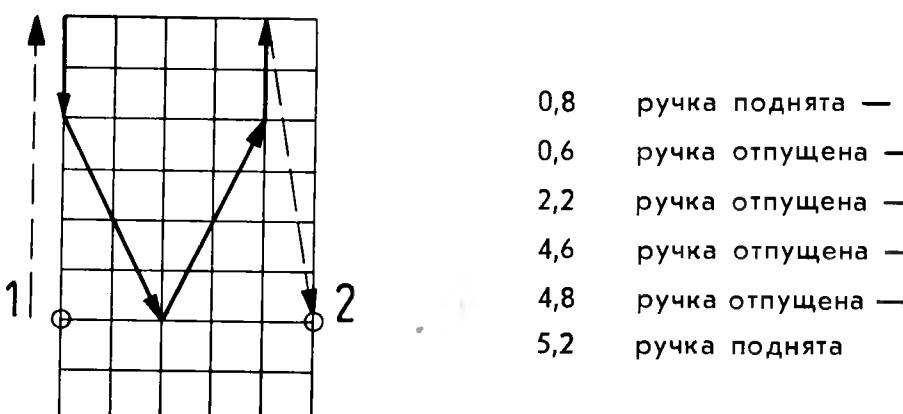


Рис. 5: Прописная буква V

##### 2. Строчная буква g

Последовательность байтов: 33H — 0A2H — 92H — 83H — 85H —  
96H — 0A6H — 0B5H — 36H — 0B1H —  
0AOH — 90H — 81H — 52H

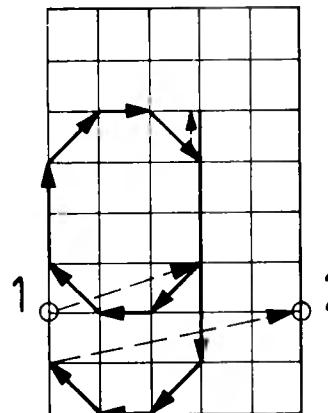


Рис. 6: Строчная буква g

Каждая последовательность байтов памяти вида символа кончается кодом 52H. Поэтому это содержание байта одновременно используется для распознавания конца.

#### 3.3.6. Алгоритм круга/дуги окружности

Используется алгоритм, описанный GERLICH (алгоритм описан в Lampe/Jorke/Wengel: „Algorithmen der Mikrorechentechnik”, VEB Verlag Technik Berlin 1983, 188—194).

По этому алгоритму вычисляются инкременты угла 0,224 градусов.  
(Для каждого градуса вычисляются 256 инкрементов.)

Алгоритм построен таким образом, что траектория вычисляется рекурсивно с помощью предыдущих значений. При этом передвижение по траектории осуществляется в математически положительном направлении. Если описывает параметр передвижения, то для четных и нечетных пар значений  $x_i, y_i$  получаются разные формулы для вычисления

$$x_{2i+1} = x_{2i} - (\Delta\varphi) y_{2i}$$

$$y_{2i+1} = (\Delta\varphi) \cdot x_{2i+1} + y_{2i} + (\Delta\varphi)^2 y_{2i}$$

для нечетных и

$$y_{2i+2} = (\Delta\varphi) x_{2i+1} + y_{2i+1} - (\Delta\varphi)^2 y_{2i+1}$$

$$x_{2i+2} = x_{2i+1} + (\Delta\varphi)^4 x_{2i+1} - (\Delta\varphi) y_{2i+2} - (\Delta\varphi)^3 y_{2i+2}$$

для четных пар значений. Это значит, что вычисление осуществляется по способу двойных шаг. После каждого третьего двойного шага проводится следующее исправление:

$$x_{2i+2} := x_{2i+2} + (\Delta\varphi)^3 y_{2i+2}$$

$$y_{2i+2} := y_{2i+2} - (\Delta\varphi)^3 x_{2i+2}$$

Умножение реализуется путем сдвигов байта перед сложением или вычитанием. Таким образом не требуется использование операций умножения, на которые расходуется много времени. Если работа ведется с боль-

шими инкрементами углов, т.е. 8, то нужно выполнить дополнительные операции сдвига. Операционная система использует  $m = 8$  для радиусов = 400 длин шага,  $m = 7$  для радиусов = 200 длин шага и  $m = 6$  для радиусов = 100 длин шага. Для радиусов = 50 длин шага для вывода предоставляется только каждое второе вычисляемое значение, при радиусах 50 только каждое четвертое вычисляемое значение. При таких условиях ошибка меньше половины длины шага, так что даже при самом маленьком инкременте угла нет возможности для более высокой точности представления на графопостроителе. За то при маленьких углах получается значительный выигрыш по времени (примерно фактор 5).

Алгоритм прерывается, когда следующая пара значений в  $x$  становится отрицательной. Таким образом получаются удобные возможности исправления ( $x = 0$ ,  $y = \text{радиус}$ ). Новый квадрант начинается начальными значениями  $x = \text{радиус}$ ,  $y = 0$ .

Это значит, что вычисление виртуально всегда осуществляется в первом квадранте, представление круга достигается путем переставления и при необходимости дополнительного обращения знака вычисляемых значений  $x$  и  $y$ .

Количество вычисляемых пар значений  $x_i$ ,  $y_i$  проверяется. Для полного круга требуются 1608 пар значений. При генерации дуг окружности центральный угол задает количество необходимых пар значений. Программа генерации круга прерывается, когда достигнуто заданное число шаг вычисления.

Направление вращения при представлении круга или дуги окружности достигается путем соответствующей перестановки вычисляемых значений  $x$  и  $y$  аналогично представлению квадрантов (смотри рис. 7).

Квадрант	Движение			
	против часовой стрелки	по часовой стрелке	ENDX	ENDY
0	x	y	x	y
1	-y	x	-y	-x
2	-x	-y	-x	y
3	y	-x	y	x

Расположение квадрантов

Рис. 7: Присвоение вычисляемых значений для  $x$  и  $y$

### 3.4. Примечания к отдельным командам

#### 3.4.1. Команды для представления векторов PA и PR

Переданные параметры декодируются в 2 байтовых двоичных числа в дополнительном коде. При команде PA параметры непосредственно обозначают точки координат виртуальной матрицы графопостроения. Поскольку генератору векторов нужны относительные расстояния в ячейках памяти ENDX и ENDY, абсолютные координаты преобразуются в относительные, которые после этого передаются генератору векторов. Команда PR уже содержит относительные координаты, которые непосредственно загружаются в ENDX и ENDY. Только при командах PA и PR проверяется, используется ли в качестве терминатора „точка с запятой“. При всех других командах операционной системе знакомо число параметров, т.е. достаточно осуществлять проверку, не содержит ли байт цифру или десятичную точку.

#### 3.4.2. Команды ручки PD и PU

Эти команды вызывают изменение содержания рабочей ячейки FEDER. Операционная система эту ячейку использует многофункционально (b0 — Ready, зеленая лампа; b1 — Error, красная лампа; b2 — шаг в направлении  $-x$ ; b3 — информация для ручки; b4 — импульс шага  $+x$ ; b5 — импульс шага  $-y$ ; b6 — импульс шага  $+y$ ; b7 — механический упор).

Информация для ручки содержится в бите 3. При команде PD устанавливается  $b3 = 1$ , при команде PA  $b3 = 0$ .

После изменения содержания ячейки FEDER вывод байта осуществляется через канал В PIO 1 (OUT 9).

Вывод осуществляется независимо от того, нажата ли клавиша управления ручкой (которая имеет более высокий приоритет) или нет.

#### 3.4.3. Команда для типа штриха LT

При каждой операции вывода генератора векторов осуществляется переход к подпрограмме типов штриха. Эта подпрограмма определяет поднятое или отпущенное состояние ручки. Параметр длины команды LT определяет длину самого длинного штриха последовательности типа штрихов в длинах шага. Загружается максимальное количество 7 информации длин в ячейки памяти, которые отражают построение последовательности (напр. длинный штрих — промежуток — точка — промежуток — точка — промежуток для типа штриха 6). Последний байт последовательности загружается нулем; это одновременно является признаком для конца последовательности. Точка символизируется штрихом длины 0,5 мм.

Алгоритм для вида штриха осуществляется следующим образом:

Представление вектора начинается отпущененной ручкой. В регистр A' загружается первая ячейка памяти последовательности. Для каждой операции вывода регистр A' декрементируется один раз. При проходе через ноль информация для ручки (бит 3 ячейки памяти FEDER) обращается, содержание следующей ячейки памяти последовательности загружается в регистр A' и проверяется равенство с нулем. Если содержание равно нулю, то таким образом распознается конец последовательности и последовательность начинается заново. Если содержание не равно нулю, то последовательность продолжается по вышеописанному алгоритму.

Тип штриха таким образом является независимым от выбранной скорости. В зависимости от наклона вектора отдельные штрихи имеют разную длину.

При горизонтальном и при вертикальном направлении рисования длина штриха равна количеству длин шага, при наклоне получается удлинение до максимального фактора 2.

#### 3.4.4. Дополнительные команды для деления осей координат TL, XT и YT

Всегда имеет место, что с помощью команд XT и YT можно вызвать рисование маркировки шкалы, которая предварительно определена командой TL tp, tn. Рисование маркировки шкалы осуществляется только один раз. Каждую новую позицию предварительно нужно определять с помощью PA x,y; или PR  $\Delta$  x,  $\Delta$  y.

#### 3.4.5. Команды для генерации круга CI, AA и AR

Как уже описано, генерация круга осуществляется с оптимальной точностью. Задание погрешности (chord tolerance c.t.) является избыточным. Если для c.t. задается значение, то оно игнорируется программой.

При генерации круга при радиусах  $>= 400$  длин шага для угла 90 градусов вычисляются точно 402 инкремента углов. После этого в зависимости от квадранта с помощью  $|x| = R$ ,  $y = 0$  или  $|y| = R$ ,  $|x| = 0$  автоматически осуществляется компенсация до правого угла. При представлении круга первая точка обращения после  $x < 0$  в соответствии с вышеописанным алгоритмом вычисления принимается как точная точка и все остальные углы 90 градусов компенсируются по отношению к этим координатам. Центральный угол фи может быть больше 360 градусов. В этом случае дуги рисуются друг на друга. Счет квадрантов может циклически изменяться (Q4 соответствует Q0 и т.д.). Циклическое изменение достигается путем блокировки битов 0...2 запоминающей ячейки для счета квадрантов QUAD.

СМ 6415

(К 6418.031)

Тестовая программа

Настоящая тестовая программа для графопостроителя СМ 6415 служит для проверки всех функций графопостроителя.

Программа построена таким образом, что описанные в описании языка команды графопостроения используются хотя бы один раз. Таким образом проверяется и демонстрируется правильное действие этих команд.

Следующая программа графопостроения представляет собой обзор функций графопостроителя СМ 6415 и служит для демонстрации части спектра возможностей устройства:

```
DT!;PA400,2500;SR15,15;
LBPLOTTER K 6418 — FUNKTIONSÜBERSICHT!;
PA402,2502;
LBPLOTTER K 6418 — FUNKTIONSÜBERSICHT!;
PA404,2504;
LBPLOTTER K 6418 — FUNKTIONSÜBERSICHT!;
DF;PA200,2300;
LBPlotter-Schriftsatz 1 ←
Standardgroesse: ←
ABCDEFGHIJKLMNOPQRSTUVWXYZ ←
abcdefghijklmnopqrstuvwxyz ←
1234567890 ←
```

```
!“#%‘()*+,-./:;<=>? {} ||| ↑ ETX;
```

```
PA200/1800;SR8,6;SL1 ←
LBMann kann schreiben: ←
————— ← ETX;
```

```
SR20,20;SLO;LBgross,ETX;
SR4,4;LB klein,ETX;
PA200,1400;SR6,12;LBschmalETX;
SR20,12;LB breit,ETX;
PA200,1250;SR8,6;SL1;LBschrägETX;
SL2;LB nachETX;SL3;LB rechts,ETX;
PA200,1150;SL-1;LBschrägETX;
SL-2;LB nach ETX;SL-3;LB links,ETX;
PA200,1000;SLO;SR6,6;
LBund in jede Richtung, z.B.:ETX;
PA500,550;PD; PR100,0;PU; PR-50,50;PD;
PRO,-100;PU;
PA650,550;SL2;SR6,6;LBO GradETX;
PA620,620;RO45;LB45 GradETX;RO90;PA650;980;
LB90 GradETX;RO135;PA420,1080;LB135 Grad;
RO180;PA200,1000;LB180 GradETX;DF;
PA200,300;SR6,6;LBundETX;
PA1000,300;DI-2,1;LBSPIEGELSCHRIFTETX;DF;
PA1200,2300;
LBPlotter-Schriftsatz (kyrill.) ←
```

```
Standardgroesse: ETX;
PA 1200,2160;
```

```
LBAБЦДЕФГХИЙКЛМНОПЯРСТУЖВЬ ←
ЫЗШЭЩЧЮ ←
абцдефгхийклмнопярстужвь
ызшэщчю
PA1200,1800;SR8,6;SL1;
LBLinientypen: ←
————— ETX;
PA1200,1600;LB1 ETX;LT1,50;PD;PR700,0;PU;
PA1200,1500;LB2 ETX;LT2,50;PD;PR700,0;PU;
PA1200,1400;LB3 ETX;LT3,50;PD;PR700,0;PU;
PA1200,1300;LB4 ETX;LT4,50;PD;PR700,0;PU;
PA1200,1200;LB5 ETX;LT5,50;PD;PR700,0;PU;
PA1200,1100;LB6 ETX;LT6,50;PD;PR700,0;PU;
DF;TW1200,100 2000,900;PA2000,900PD
PR-800,0 0,-800 800,0 0,800;PU;
PA1500,500; CI100; PA1400,500; CI200;
PA1300,500; CI500; PA1200,500; CI400;
PA1100,500; CI500; PA1000,500; CI600;
PA1700,500; CI100; PA1800,500; CI200;
PA1900,500; CI300; PA2000,500; CI400;
PA2100,500; CI500; PA2200,500; CI600;
IW2200,1100 3400,2300;
PA2200,1100;PD;PR1200,0 0,1200-1200,0 0,1200;PU;
PA2800,1100;AA2200,1100,1000;
PA2850,1100;AA2200,1100,1000;
PA2900,1100;AA2200,1100,1000;
PA2950,1100;AA2200,1100,1500;
PA3000,1100;AA2200,1100,1500;
PA3050,1100;AA2200,1100,1500;
PA3400,1700;AA3400,1100,1000;
PA3400,1750;AA3400,1100,1000;
PA3400,1800;AA3400,1100,1000;
PA3400,1850;AA3400,1100,1500;
PA3400,1900;AA3400,1100,1500;
PA3400,1950;AA3400,1100,1500;
PA2800,2300;AA3400,2300,1000;
PA2750,2300;AA3400,2300,1000;
PA2700,2300;AA3400,2300,1500;
PA2650,2300;AA3400,2300,1500;
PA2600,2300;AA3400,2300,1500;
PA2550,2300;AA3400,2300,1500;
PA2200,1700;AA2200,2300,1000;
PA2200,1650;AA2200,2300,1000;
PA2200,1600;AA2200,2300,1500;
PA2200,1550;AA2200,2300,1500;
PA2200,1500;AA2200,2300,1500;
PA2200,1450;AA2200,2300,1500;
PU;DF;
PA2300,200;PD;
PR0,100;YT;PR0,100;PR0,100;YT;PR0,100;YT;PR0,100;YT;
```

```

PRO,200 -25,-50;PU;PR50,0;PD;PR-25,50;PU;
PA2300,200;TL0,15;PD;PR100,0;XT;
TL0,30;PR100,0;XT;TL0,15;PR100,0;XT;
TL0,30;PR100,0;XT;TL0,15;PR100;OXT;
TL0,30;PR100,0;XT;TL0,15;PR100,0;XT;
PR200,50-50,25;PU;PR0,-50;PD;PR50,25;PU;
PA2200,900;SR8;SLO;LBYETX;
PA2300,200;PD;PR100,100 50,200 300,25
200,-50 50,0 200,200;SMA;PU;
PA0,0;PD;PR3700,0 0,2700 -3700,0 0,-2700;
NR;

```

## PLOTTER K 6418 – FUNKTIONSUBERSICHT

Plotter-Schriftsatz (dt..)  
 Standardgroesse:  
 ABCDEFGHIJKLMNOPQRSTUVWXYZ  
 abcdefghijklmnopqrstuvwxyz  
 1234567890  
 !@#\$%^&\*()\_+=?;:.,{ }\_{}()

Man kann schreiben:

- GROSS / klein.      1 \_\_\_\_\_
- schmal, breit,      2 —— —— ——
- schraeg nach rechts,      3 —— —— ——
- schraeg nach links,      4 —— —— ——
- und in jede Richtung, z.B.:

$\rho \theta \cdot \cos \varphi$        $\rho \theta \cdot \sin \varphi$   
 $\rho \theta \cdot \cos \varphi + \theta \text{ Grad}$   
 und  
 TRANSFORME

