

ПРОГРАММНО-АППАРАТНЫЙ КОМПЛЕКС "КОСМОТРОНИКА"

Язык графического представления объектов "графика".

Руководство пользователя

Аннотация

Настоящий документ предназначен для изучения языка описания экранных форм в АРМ "Телемеханика", методов составления, трансляции и проверки программ, принципов взаимодействия с экранными формами в процессе работы АРМ. Также, документ содержит описание входных и выходных данных. Приведены примеры программ и справочная информация об основных элементах языка.

Оглавление

1. Введение.
2. Структура описания экранной формы.
3. Специальные символы.
4. База данных настроек АРМ.
5. Структура таблицы Graphics.
6. Проект.
7. О задании цвета.
8. Используемые расширения файлов.
9. Раздел Var и типы переменных.
 - 9.1. Применение переменных.
10. Как определить состояния объекта.
 - 10.1. Состояние связи с объектом.
 - 10.2. Состояния аварий и предупреждений на объекте.
11. Состав окна для вывода экранных форм.
 - 11.1. О масштабировании картинка.
 - 11.2. Всплывающая подсказка.
12. Правила составления программы.
 - 12.1. Применение команды CallCell.
13. Раздел Base.
 - 13.1. Дополнительная экранная форма.
14. Раздел Input.
15. Раздел Picture.
 - 15.1. Pen
 - 15.2. Brush
 - 15.3. Color
 - 15.4. Font
 - 15.5. FontStyle
 - 15.6. Format
 - 15.7. ColorAvar
 - 15.8. ColorDisable
 - 15.9. ColorBlink

15.10. Vid
15.11. Scale
15.12. Angle
15.13. Style
15.14. Trim
15.15. Arc
15.16. Chord
15.17. Ellipse
15.18. Pie
15.19. Polygon
15.20. Polyline
15.21. Rect
15.22. RoundRect
15.23. Text
15.24. Image
15.25. CallCell
15.26. Int
15.27. Float
15.28. Clr
15.29. Point
15.30. String
15.31. Ident
15.32. SignTC
15.33. SignTI
15.34. SignTY
15.35. SignCN
15.36. Mouse
15.37. Menu
15.38. Button
15.39. PassTC
15.40. Header
15.41. OutInt
15.42. OutFloat
15.43. OutClr
15.44. OutPoint
15.45. OutString
15.46. OutIdent
15.47. OutSignTC
15.48. OutSignTI
15.49. OutSignTY
15.50. OutSignCN
15.51. MouseTC
15.52. Header1
15.53. OutInt1
15.54. OutFloat1
15.55. OutClr1
15.56. OutPoint1
15.57. OutString1
15.58. OutIdent1
15.59. OutSignTC1
15.60. OutSignTI1
15.61. OutSignTY1

- 15.62. OutSignCN1
- 15.63. Opt
- 15.64. Lines
- 15.65. TrimButton

16. Раздел Library.

- 16.1. Int
- 16.2. Float
- 16.3. Clr
- 16.4. Point
- 16.5. String
- 16.6. Ident
- 16.7. SignTC
- 16.8. SignTI
- 16.9. SignTY
- 16.10. SignCN
- 16.11. Attr
- 16.12. FreshTC, FreshTI
- 16.13. TableTI
- 16.14. TableTC
- 16.15. Сигналы “Телеизмерение”
- 16.16. Команды обработки телеизмерений
- 16.17. Телесигналы.
- 16.18. Fon
- 16.19. LevelTI
- 16.20. ArrowTI
- 16.21. MeterTI
- 16.22. PolylineTC
- 16.23. PolygonTC
- 16.24. RectTC
- 16.25. RoundRectTC
- 16.26. EllipseTC
- 16.27. GraphTI
- 16.28. PolylineTI
- 16.29. PolygonTI
- 16.30. RectTI
- 16.31. RoundRectTI
- 16.32. EllipseTI
- 16.33. Особенности работы команд TextTC, TextTI, ValueTI, TimeValueTI, TextValueTI.
- 16.34. TextTC
- 16.35. TextTI
- 16.36. ValueTI
- 16.37. TimeValueTI
- 16.38. TextValueTI
- 16.39. Особенности работы команд Multi...
- 16.40. MultilineTC
- 16.41. MultilineTI
- 16.42. MultigonTC
- 16.43. MultigonTI
- 16.44. MultiTextTC
- 16.45. MultiTextTI

- 16.46. Особенности работы команд S...
- 16.47. SPolylineTC
- 16.48. SPolygonTC
- 16.49. SRectTC
- 16.50. SRoundRectTC
- 16.51. SEllipseTC
- 16.52. STextTC
- 16.53. Особенности экранной формы щита и управление миганием.
- 16.54. SignalBlinks
- 16.55. StateBlinks
- 16.56. Особенности работы команд Button...
- 16.57. ButtonTC
- 16.58. RoundButtonTC
- 16.59. ButtonTI
- 16.60. RoundButtonTI
- 16.61. SButtonTC
- 16.62. SRoundButtonTC
- 16.63. Сигналы “Константа”
- 16.64. TableCN
- 16.65. ValueCN
- 16.66. DateTC
- 16.67. TimeTC
- 16.68. DateTimeTC
- 16.69. DateTI
- 16.70. TimeTI
- 16.71. DateTimeTI
- 16.72. DateCN
- 16.73. TimeCN
- 16.74. DateTimeCN
- 16.75. Columns
- 16.76. ButtonTCTI
- 16.77. RoundButtonTCTI
- 17. Раздел Insert.
 - 17.1. Chart.
- 18. Анимация.
- 19. Вывод текущего времени.
- Приложение А. Группы команд. Краткое описание команд.
- Приложение Б. Перечень основных значений цветов в командах
- Приложение В. Перечень стилей текстов
- Приложение Г. Перечень ключевых слов и символов.
- Приложение Д. Примеры программ.

1. Введение

В документе представлен язык описания экранных форм в АРМ “Телемеханика”. Язык имеет условное название “АРМ-графика”. Описанием экранной формы является программа на языке “АРМ-графика”. Результаты трансляции программ используются программным комплексом АРМ “Телемеханика” для отображения экранных форм на экране монитора диспетчера. Программа позволяет описать общую картинку экранной формы и визуальные образы, поведение которых зависит от поступающих в АРМ сигналов. Далее сигналами именуется поступающие на вход АРМ телесигналы и телеизмерения от датчиков и исполнительных устройств. Перечень экранных форм и

таблицы сигналов со своими атрибутами располагаются в базе данных настроек АРМ. Экранная форма может быть привязана к конкретному объекту или являться независимым графическим образом.

Описание экранных форм на языке “АРМ-графика” обеспечивает решение следующих задач:

- определение объекта как набора составляющих его ячеек (агрегатов);
- прорисовку графики на экране монитора в различных масштабах без потери качества, что реализуется описанием образов в векторном формате;
- наблюдение за поведением объекта, что обеспечивает привязка элементов экранных форм к сигналам - элементы графики (цвет, форма, местоположение, текст) меняются в зависимости от состояния сигналов;
- управление объектом посредством взаимодействия пользователя АРМ через мышь и клавиатуру с графическими образами объекта на экране;
- повторное применение ранее созданных описаний ведением библиотек элементов экранных форм.

Для составления программы на языке “АРМ-графика” разработан специализированный программный комплекс, который состоит из многооконного текстового редактора, графического редактора, транслятора, отображателя экранных форм и имитатора. Имитатор предназначен для наблюдения за поведением экранной формы в режиме поступления случайного набора сигналов.

Описания экранных форм после их трансляции выводятся на экран в виде картинки. Картинку выводит специализированный программный модуль, именуемый отображателем экранных форм. Этот же модуль используется программным комплексом АРМ “Телемеханика”. В комплексе АРМ программный модуль после вывода на экран первичного изображения переходит в режим обработки событий. Событиями являются поступающие в АРМ сигналы и управляющие воздействия пользователя (мышь, клавиатура). АРМ работает в режиме реального времени. Поступающие с объекта сигналы изменяют соответствующие описанию элементы экранной формы, а некоторые воздействия пользователя отправляют сигналы управления агрегатами объекта. Допускается иметь на экране одновременно два окна с различными экранными формами. Второе окно обычно вызывается пользователем посредством мыши и используется для вывода формы, например, специально составленной для управления выбранным агрегатом объекта. При подключении к компьютеру нескольких дисплеев АРМ поддерживает многомониторный режим, позволяя выводить на каждый дисплей свою экранную форму.

2. Структура описания экранной формы.

Объект может состоять из нескольких листов. Каждый лист объекта представляется экранной формой, помещаемой в отдельный файл (программу). Перечень листов, объектов и соответствующих им файлов задается в таблице Graphics базы данных настроек АРМ. Описание листа на языке “АРМ-графика” состоит из следующих разделов:

Base – содержит базовое описание листа, которое включает: номер объекта, номер листа, имя листа, размер листа, настройки, перечень имен внешних библиотек.

Var – состоит из перечня переменных.

Каждой переменной объявляется тип переменной. Область видимости переменных – вся программа. В любом месте программы переменной можно присвоить произвольное значение в соответствии с ее типом. Переменные могут использоваться в командах при записи их параметров. Присвоенное значение переменной действует до смены ее значения по ходу исполнения программы. Изменение значения переменной в подпрограмме

действует в пределах данной подпрограммы, т.е. при завершении исполнения подпрограммы переменные восстанавливаются. Переменным разрешается присваивать только константные значения. Выражения не допускаются.

Input – перечисляет входные параметры программы с указанием их типов. Может отсутствовать. Используется в многократно вызываемых программах с целью их настройки перед очередным запуском. Вызывающая программа определяет входные параметры вызываемой программы, используя перед командой вызова Mouse специальный набор команд (OutInt, OutString, и т.д.). Открытие окна вызываемой экранной формы и передача фактических значений входных параметров происходит при щелчке мышью в области, заданной командой Mouse. Перечень параметров из раздела Input рассматривается в программе как набор инициализированных при ее загрузке переменных.

Picture – содержит описание экранной формы и определяет ее поведение на действия пользователя и изменения состояний сигналов системы. Раздел Picture состоит из последовательности команд, в число которых входят:

- команды управления средой рисования;
- команды реагирования на действия пользователя;
- элементарные графические команды, не имеющие привязки к сигналам;
- команды CallCell вызова подпрограмм.

Команда CallCell является вызовом подпрограммы. Ее следует рассматривать как ссылку на библиотечное описание ячейки и ее настройку. Параметры команды CallCell определяют местоположение ячейки (координата базовой точки), ссылочное имя вызываемой подпрограммы из раздела Library и имя ячейки. Для передачи данных в подпрограмму перед командой CallCell необходимо перечислить фактические параметры, сгруппировав их по типам. Фактическим параметром является конкретное значение, например, координаты точки, идентификатор телесигнала и т.д. Поведение ячейки определяется ее описанием в библиотеке. Начать описание ячейки следует с перечня формальных параметров, состав которых должен строго соответствовать перечню передаваемых фактических параметров. Формальным параметром является произвольный идентификатор. При исполнении подпрограммы происходит замена формальных параметров фактическими параметрами, что позволяет многократно обращаться к однотипным ячейкам в различных местах раздела Picture с различными значениями фактических параметров.

Library – является внутренней библиотекой листа и предназначена для описания ячеек не используемых повторно при описании других экранных форм. Состоит из набора библиотечных ячеек. Каждая библиотечная ячейка имеет уникальное имя, именуемое типом ячейки. При отсутствии описания типа ячейки в разделе Library предполагается, что ее описание находится во внешней библиотеке. Структура внешней библиотеки идентична структуре данного раздела. Каждая внешняя библиотека является отдельным файлом с расширением glb. Ссылка к внешней библиотеке производится по имени файла. Все размеры и координаты графических элементов библиотечной ячейки указываются относительно ее базовой точки. Прорисовка ячейки на экране производится после пересчета координат библиотечной ячейки по месту, указанному в команде CallCell. Поведение библиотечной ячейки описывается набором формальных параметров с произвольным набором имен. При подстановке библиотечной ячейки в лист формальные параметры заменяются фактическими, указанными перед командой CallCell. Число и типы формальных и фактических параметров должны совпадать. Раздел Library состоит из набора подразделов Cell, по одному для каждой ячейки. Подраздел Cell может включать команды из набора элементарных и специализированных команд. Специализированные

команды предназначены для отображения поведения сигналов или их вывода в виде таблиц и графиков.

Insert – предназначен для описания окон с графиками, встраиваемых в экранные формы. Каждое окно содержит перечень телесигналов или телеизмерений, выводимых на график. Перечень сигналов определяют командами SignTC и SignTI. Окно с графиками задается командой Chart, применение которой возможно только в разделе Insert. В окне допускается не более четырех сигналов только одного типа. В текущем проекте АРМ во всех его экранных формах суммарное число окон с графиками не должно превышать 16-и. При запуске АРМ для каждого окна с графиками создается невидимая экранная форма, которая обладает собственной памятью и правилами обработки сигналов. В процессе работы АРМ поступающие сигналы передаются в окна, поддерживая полноту данных для прорисовки графиков в любой момент времени. При вызове экранной формы с разделом Insert соответствующие окна графиков привязываются по месту и переводятся в режим видимости. Визуально поведение окон ничем не отличается от поведения изображений от специализированных команд, за исключением того, что их графики (в отличие от команды GraphTI – устаревшая команда) сразу отображают все накопленные данные за указанный период наблюдения. Каждая команда Chart имеет индивидуальные средства настройки. Сохранение настроек обеспечивается в едином конфигурационном файле Insert.cfg.

Примечание.

Разработанный механизм внедрения позволяет привязывать к экранным формам по требованию заказчика сколь угодно сложные программные объекты с собственной памятью и оперативно отображать на мониторе состояние решаемых ими задач.

Все размеры и координаты задаются в условных единицах и являются целыми числами, трактовать которые составитель описания может по собственному усмотрению. Начало координат расположено в верхнем левом углу рисунка. Ось X направлена слева направо. Ось Y – сверху вниз.

При описании разделов и синтаксиса команд приняты следующие обозначения:

(...) – число (целое без знака);

[...] – идентификатор (алфавитно-цифровая последовательность символов);

<...> - текст, заключенный в кавычки;

(...)/(...) – координата (пара чисел через символ “слэш”);

Примечание.

Допускается использовать вместо ключевых слов CallCell и Cell слова CallSell и Sell.

3. Специальные символы.

// - начало комментария;

{ - открывающая скобка;

}- закрывающая скобка;

= - присвоение значения переменной;

+ - признак продолжения строки параметров на следующей строке. Должен быть последним в строке.

\n – пара символов. Все команды вывода текста позволяют отображать на экране монитора многострочные тексты. При описании текстов строки отделяются символами "\n" (без кавычек). Пример вывода двух строк:

Angle 45

Text 338/205,"Текст 45\nградусов"

В тексте допускается иметь не более 10-и строк.

! – при записи идентификаторов сигналов символ ‘!’ можно использовать вместо номера абонента, если номер абонента сигнала совпадает с номером объекта экранной формы.

Примечание. Номер объекта задается в разделе Base.

Пример применения символа ‘!’:

```
Base 520,1,"ПС-Агат" //номер объекта, номер листа, имя листа
```

```
...
```

```
Picture
```

```
{
```

```
...
```

```
SignTC Link! //вместо SignTC Link520
```

```
...
```

```
SignTI !_51_150 //вместо SignTI 520_51_150
```

```
...
```

```
}
```

Применение символа ‘!’ можно запретить в настройках программы трансляции экранных форм.

4. База данных настроек АРМ.

База данных настроек АРМ составляется в формате MDB. Состоит из ряда таблиц. При составлении программы для экранной формы используются данные из следующих таблиц БД настроек АРМ:

Objects – перечень объектов

Graphics – перечень программ с их привязкой к объектам

ТС – перечень телесигналов

ТІ – перечень телеизмерений

ТУ – перечень телеуправлений

Macro – перечень макрокоманд

Const – перечень констант

В таблицах ТС, ТІ, ТУ, Const при описании сигналов, помимо других параметров, задаются текстовые имена и числовые идентификаторы. Идентификатор состоит из трех чисел:

NumAbon - номер абонента в таблице Objects, которому принадлежит сигнал (целое без знака);

NomGr - номер группы (целое);

NomPar - номер параметра (целое без знака).

Номера групп и параметров – произвольные числа. Каждый сигнал имеет уникальный идентификатор. Заметим, что поля MinVal, MaxVal из таблицы ТІ используются программой Graph.exe в режиме имитации для определения диапазона возможного значения сигнала.

5. Структура таблицы Graphics.

Таблица имеет следующие поля:

NumAbon – номер объекта в АРМ (целое);

List - номер листа в объекте (целое - байт);

GraphFile – имя файла с описанием графики (текст 20 симв.).

Составной ключ таблицы Graphics имеет имя IdentGraphics, который организован по возрастанию полей NumAbon и List.

NumAbon может быть положительным или отрицательным числом.

Экранная форма с положительным значением NumAbon определяет номер объекта в АРМ и проверяется на наличие в таблице Objects из БД настроек АРМ. Один и тот же NumAbon может иметь несколько экранных форм, но с различными номерами листов. Каждый лист объекта оформляется отдельным файлом. Нумерация листов объекта должна представлять натуральный ряд чисел, начиная с единицы. Интерфейс окон для вывода экранных форм имеет удобные средства навигации по листам объекта. По желанию составителя экранной формы эти средства можно отключить.

Экранные формы с отрицательными значениями NumAbon далее именуются дополнительными экранными формами (ЭФ). Дополнительные ЭФ не имеют описаний в таблице Objects. Один и тот же NumAbon может иметь несколько листов. Каждый лист оформляется отдельным файлом. К дополнительным ЭФ, например, относятся формы с элементами управления, вызываемые из экранных форм объектов. Программы graph.exe и art.exe позволяют иметь на экране одновременно одну ЭФ объекта и одну или несколько дополнительных ЭФ. Максимально допустимое число одновременно видимых дополнительных ЭФ определяется при настройке программы. По умолчанию допускается одна дополнительная ЭФ.

Программа graph.exe выводит картинку экранной формы при ее трансляции или отображении с имитацией. В программе art.exe выводятся экранные формы объектов при нажатии соответствующей объекту кнопки на главной форме программы. В обеих программах дополнительные ЭФ всегда выводятся щелчком левой кнопки мыши по области текущей картинки, определенной командой Mouse с предписанным действием: “Перейти к картинке объекта по имени файла” или командой или MouseTC. Если NumAbon в вызываемом файле не отрицательное число, то произойдет смена картинки текущего объекта на картинку нового объекта. Если NumAbon отрицательное число, то в зависимости от настройки появится новая дополнительная ЭФ или старая ЭФ заместится новой. Подробности см. П.13.1.

6. Проект.

Проектом является комплект исходных текстов программ, библиотек и результатов трансляции для описания графики отображаемого АРМом района. Программа ведения проекта “graph.exe” начинает работу с загрузки БД настроек АРМ. Привязка к БД настроек АРМ производится посредством файла “ graph.udl”, расположенного в папке с программой “graph.exe”. Настройка файла “ graph.udl” производится системной программой связи с данными OLE DB Core Services. Ее вызов осуществляется кликом по файлу “ graph.udl” в проводнике или любом файловом менеджере. Настройка состоит во вхождении на закладку “Подключение”, указании имени базы данных и нажатии кнопки ОК.

Результатом трансляции исходного файла является одноименный файл с расширением BIN. Местоположения всех типов файлов определяются настройками программы “graph.exe”. Для работы графической подсистемы программы “АРМ Телемеханика” требуются результаты трансляции и файлы с внедряемыми картинками (расширения bmp, ico, jpg, wmf, emf), если они присутствуют в текстах исходных программ.

7. О задании цвета.

Команды Brush, Color, ColorAvar, ColorDisable, ColorBlink, Colors, Fon и ряд специализированных команд в качестве параметров содержат описания цветов. Определен набор констант для стандартных имен цветов. Их перечень приведен в приложении Б. Описание нестандартного цвета представляется трехбайтовым шестнадцатеричным числом, байты которого определяют интенсивности синего, зеленого и красного цвета соответственно. Например, значение FF0000 соответствует чистому синему цвету, 00FF00 – чистому зеленому, 0000FF – чистому красному. 000000 – черный цвет, FFFFFFFF – белый.

Примеры:
Color clRed
Brush ff1f00
Colors clGreen,ff001

8. Используемые расширения файлов.

GRA - исходные тексты программ
GLB – исходные тексты библиотек
BIN – результаты трансляции

9. Раздел Var и типы переменных.

Раздел начинается с ключевого слова Var без параметров. Имеет открывающую и закрывающую фигурные скобки. Между скобками могут быть заданы объявления следующих типов переменных:

Int – целое число
Float – число с плавающей точкой
Clr - цвет
Point – координата точки
String – текстовая строка
Ident – идентификатор
SignTC – телесигнал из таблицы TC
SignTI – телеизмерение из таблицы TI
SignTY – телеуправление из таблицы TY
SignCN – константа из таблицы Const

Синтаксис строки объявления:

[Имя типа переменной] [Перечень имен переменных данного типа через запятую]

Имя переменной - уникальная алфавитно-цифровая последовательность символов для данного проекта. Имена переменных не должны совпадать с ключевыми словами языка АРМ-графика. Перечень ключевых слов см. в приложении Г.

Пример раздела Var:

```
Var  
{Int i1,i2  
Float f  
Point pt1,pt2  
String str  
Ident id
```

```
SignTC tc1,tc2
SignTI ti1,ti2
}
```

9.1. Применение переменных.

Переменные используются в командах в качестве параметров. До применения переменной ей должно быть присвоено значение. Синтаксис команды присвоения имеет вид:

[имя переменной]= {значение переменной}

Синтаксис значения переменной зависит от типа переменной:

Int – последовательность цифр
Float – последовательность цифр с отделением дробной части точкой
Point – пара чисел через символ “слэш”. В начале числа допускается знак минус (отрицательное значение)
Clr – имя стандартного цвета или шестнадцатеричное число для нестандартного цвета
String – алфавитно-цифровая последовательность, обрамленная кавычками
Ident – алфавитно-цифровая последовательность символов
SignTC – идентификация телесигнала в таблице TC
SignTI – идентификация телеизмерения в таблице TI
SignTY – идентификация телеуправления в таблице TY
SignCN – идентификация константы в таблице Const

Синтаксис идентификации сигнала:

(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Упомянутый сигнал должен быть описан в таблице TC, TI, TY или таблице Const базы данных настроек АРМ.

В качестве идентификатора сигнала может быть присвоено значение null, что означает исключение переменной из числа обрабатываемых сигналов.

Примечание.

Не допускается использовать переменные в команде Scale. Не допускается использовать внешние переменные типа Ident внутри Cell для идентификации формальных имен сигналов. Переменная в ячейке считается внешней, если ей было присвоено последнее значение в разделе Picture.

Пример использования переменных:

```
...
Var
{
Point pt1
String str
Ident id
SignTC tc1
```

```

}
Picture
{...
id=3D
Vid id
pt1=0/0
str="Фидер N 1"
tc1=1_2_33
SignTC tc1, 2_3_44 //первый сигнал объявлен через переменную
CallCell 100/100,Feeder,str
id=2D
pt1=0/20
str="Фидер N 2"
SignTC tc1, 2_3_45 //первый сигнал объявлен через переменную
CallCell 100/200,Feeder,str
...
}
Library
{Cell Feeder
{SignTC c1,c2
Vid id //id - переменная
Text pt1,str // оба параметра переменные
RectTC tc1,20/20,40/40 //Ошибка!!! Внешняя переменная для имени сигнала
tc1=c1
RectTC tc1,20/20,40/40 //теперь правильно
...
}
}
}

```

При описании команд приводится строка с синтаксисом команды, в которой типы параметров напечатаны курсивом. Тип переменной должен совпадать с типом параметра.

10. Как определить состояния объекта

10.1. Состояние связи с объектом.

Для выяснения состояния связи с объектом определены зарезервированные имена телесигналов, которые используются в команде SignTC при описании фактических параметров перед вызовом Cell командой CallCell. Синтаксис имени сигнала связи:

Link(N объекта)

Сигнал принимает единичное значение при установлении связи с объектом и нулевое при отсутствии или потере связи. При запуске системы сигналы Link имеют неопределенные состояния.

Пример: SignTC Link22

Необходимость в сигналах Link возникает, если объект имеет несколько подобъектов (контроллеров, серверов и т.д.), имеющих собственные сигналы связи. Задача сигнала Link – произвести анализ всех собственных связей объекта и выработать посылку сигнала в подсистему прорисовки и мониторинга экранных форм. Если все связи объекта присутствуют – сигнал Link устанавливается в единицу.

10.2. Состояния аварий и предупреждений на объекте.

Для выяснения аварий и предупреждений на объекте необходимо ввести в таблицу телесигналов БД настроек АРМ строки (телесигналы) со следующими значениями в поле Flag:

TroubleXX, WarningXX.

Где XX – число. Например: Trouble23, Warning132. Флаги означают, что телесигнал расчетный и существует только в АРМ. Число задает номер объекта в АРМ. Напоминаем, что номер объекта определен в поле numAbon таблицы Objects.

Значения телесигналов с флагами Trouble становятся равными единице при появлении на объекте хотя бы одного аварийного состояния (телесигнала, телеизмерения) и переходят к нулевому значению при их отсутствии.

Значения телесигналов с флагами Warning становятся равными единице при появлении на объекте хотя бы одного предупредительного состояния (телесигнала, телеизмерения) и переходят к нулевому значению при их отсутствии.

Телесигналы с флагами Trouble и Warning должны иметь уникальные идентификаторы. Номер объекта идентификатора должен принадлежать одному из объектов таблицы Objects. Поля IntrNul и IntrOne не имеют значения. Эти телесигналы всегда являются рабочими (сделать их аварийными или предупредительными невозможно). Используются при программировании экранных форм в командах SignTC для отображения состояний объектов. Не записываются в архив.

Пример использования в ЭФ телесигналов состояния объекта:

```
Base 3,1,"7ср РУ-6кВ КНС-1" //N абонента, N листа, Имя листа
{Size 629/429,Origin
Ground E9E9E9
}
Picture
{
Text 20/140,"Паклиновская"
Text 20/163,"Связь"
Clr clGreen
SignTC Link1 //зарезервированный сигнал связи с первым объектом
CallCell 80/170,Ind,"Связь"
Text 20/193,"Аварии"
Clr clRed
SignTC 1_100_1 //расчетный сигнал наличия аварий на первом объекте
CallCell 80/200,Ind,"Аварии"
Text 20/223,"Предупр."
Clr clYellow
SignTC 1_100_2 //расчетный сигнал наличия предупреждений на первом объекте
CallCell 80/230,Ind,"Предупреждения"
}
Library
{
//При появлении единичного сигнала круг начнет мигать
Cell Ind
{
Clr c
SignTC tc
Vid Brd
Pen 3
Color clGray
```

```
ColorBlink clWhite
SignalBlinks tc
StateBlinks 1
EllipseTC clLtGray,c,tc,-10/-10,10/10
}
}
```

11. Состав окна для вывода экранных форм.

Окно содержит меню, панель инструментов и строку состояния в нижней части экрана. Между панелью инструментов и строкой состояния располагается рабочая область, отведенная для вывода экранных форм. При необходимости, рабочая область дополняется справа и снизу полосами прокрутки для навигации по экранной форме. Механизм подсказок сообщает о назначении кнопок панели инструментов при наведении на них курсора мыши.

В строке состояний слева указываются координаты текущего местоположения курсора в единицах проектирования экранной формы. Ось абсцисс располагается слева направо, а ординат – сверху вниз. Начало координат – верхний левый угол.

Второе поле в строке состояний сообщает имя файла экранной формы, номер текущего листа и число листов для данного объекта.

11.1. О масштабировании картинки.

В программе graph.exe вывод экранной формы в окно производится сразу после трансляции или при выборе пункта меню Показать/Графика+имитатор. В программе “АРМ Телемеханика” – при нажатии кнопки объекта на главной форме программы. При первом запуске картинка экранной формы выводится в масштабе, который вычисляется для подгонки картинки под размеры рабочей области окна. Окно экранной формы позволяет изменять собственные размеры стандартными в ОС Windows методами. Изменение размера окна вызывает подгонку внутреннего изображения, изменяя масштабы картинки. Окно экранной формы содержит управляющие элементы изменения масштаба картинки для приближения или удаления изображения. При изменении масштаба размеры перьев и шрифтов изменяются пропорционально.

11.2. Всплывающая подсказка.

В момент остановки курсора в области изображения библиотечной ячейки на экран над курсором выводится всплывающее окно с текстом. Окно исчезнет с экрана через пять секунд. В состав текста включаются:

- содержимое переменной “имя ячейки” из команды CallCell;
- содержимое переменных “текст” из всех команд Attr данной ячейки.

12. Правила составления программы.

Каждый раздел начинается с ключевого слова: Base, Var, Input, Picture, Insert или Library. Далее следует открывающая фигурная скобка, а в конце раздела - закрывающая фигурная скобка. За открывающей фигурной скобкой следуют команды раздела. Команда может занимать одну или несколько строк. В случае продолжения команды на следующей строке заносится символ “+” в конце текущей строки. Не допускается вводить в одной строке более одной команды. Допускаются комментарии. Комментарий начинается с

символов “//” в любом месте строки и заканчивается в конце строки. Команды в разделах Picture и Library исполняются в порядке их написания.

12.1. Применение команды CallCell.

Команда CallCell может быть использована только в разделе Picture и предназначена для исполнения подпрограммы. Имя вызываемой подпрограммы указывается в параметрах команды CallCell. Подпрограммы помещаются в раздел Library – внутренний или внешний. Описание каждой подпрограммы в разделе Library начинается с ключевого слова Cell с единственным параметром - именем подпрограммы. Команда CallCell играет важную роль для “оживления” экранной формы. Имеется ряд специальных команд рисования, применение которых возможно только в подпрограммах. Эти команды позволяют менять форму рисунка, его цвет, сопутствующий текст, а также вызывать мигание рисунка в зависимости от состояния сигналов или их комбинаций. Для предания возможности многократного вызова подпрограммы с различными наборами сигналов используется механизм фактических и формальных параметров.

До вызова подпрограммы, т.е. задания команды CallCell, необходимо указать набор фактических параметров. Объявление фактических параметров производится следующим набором команд:

Int – перечень целых чисел

Float – перечень чисел с плавающей точкой

Clr - перечень цветов

Point – перечень координат точек

String – перечень текстовых строк

Ident – перечень идентификаторов

SignTC – перечень идентификаторов телесигналов из таблицы TC

SignTI – перечень идентификаторов телеизмерений из таблицы TI

SignTY – перечень идентификаторов телеуправлений из таблицы TY

SignCN – перечень идентификаторов констант из таблицы Const

Заметим, что перечень объявлений фактических параметров совпадает с перечнем типов переменных.

Описание вызываемой подпрограммы в разделе Library следует начать с перечня формальных параметров. Объявление формальных параметров производится тем же набором команд, который был перечислен для фактических параметров. Но, если фактическими параметрами являлись конкретные значения, то формальными параметрами являются имена (идентификаторы) произвольного содержания. Составитель программы вправе задавать имена по своему усмотрению. Не допускается только их совпадение с ключевыми словами языка АРМ графика.

В процессе работы АРМ вызов подпрограммы для исполнения сопровождается подменой формальных параметров фактическими. Необходимым условием вызова подпрограммы является совпадение перечня команд объявления фактических и формальных параметров, а также их количества для каждого объявления.

Приведем пример вызова подпрограммы:

```
...
Picture
{...
SignTI 3_49_6,3_49_2 //фактические телеизмерения
String "F нм3/ч","F нм3" //фактические заголовки строк в таблице
CallCell 710/233,Tab2,"Корректор"
...
SignTI 3_49_5,3_49_1 //фактические телеизмерения
```

```
String "F м3/ч","F м3" //фактические заголовки строк в таблице
CallCell 348/748,Tab2,"UC10F001"
...
}
Library
{Cell Tab2 //Таблица телеизмерений на 2 строки
{Vid 2D
SignTI p1,p2 //формальные телеизмерения
String s1,s2 //формальные заголовки строк в таблице
Signals p1,p2
Color clBlack
Pen 1
Brush clWhite
Texts s1,s2
Format 7,2,trR
Fon b3b7ff,clYellow,clWhite,clWhite,clWhite,clYellow,b3b7ff
TableTI 2,1,clBlack,0/0,85/32
}
}
```

13. Раздел Base.

Раздел начинается с ключевого слова Base. Синтаксис предложения Base:

Base (номер объекта в АРМ),(номер листа),<имя листа>

Где: номер объекта в АРМ – число для идентификации объекта;

номер листа - число для идентификации листа в объекте. Не более 255. Объект может иметь один или несколько листов;

имя листа – текст не более 79-и символов.

Номер объекта и номер листа сверяются с ранее записанными в таблице Graphics и при отсутствии заносятся в таблицу. Счет листов начинается с единицы. Имя листа необязательный параметр. При его наличии выводится в заголовок окна с графикой. При отсутствии имени листа в заголовок окна выводится имя объекта из таблицы Objects.

Вслед за открывающей скобкой раздела вводится предложение с ключевым словом Size. Синтаксис предложения Size:

Size (размер картинки по оси X)/ (размер картинки по оси Y),[перечень признаков]

Размеры картинки должны задаваться с учетом масштабирования изображения. См. описание команды Scale. Признаки перечисляются через запятую и являются необязательными параметрами. Допускаются следующие признаки:

Fixed - признак немасштабируемого изображения. При его наличии размеры картинки объекта на экране соответствуют указанным в предложении Size и не изменяются при воздействии на элементы управления изображением.

Simple – определяет оформление выводимого на экран окна. При его отсутствии окно содержит меню, инструментальную панель, строку состояний и позволяет пользователю изменять размеры окна. При наличии признака Simple окно содержит только картинку. Используется, например, для создания экранной формы с органами управления, вызываемой из других экранных форм. Экранная форма с признаком Simple должна иметь один лист. Меню можно оставить в окне, если на закладке "Условия" по пункту меню "Настройки\Конфигуратор" поставить галочку в строке "Показать меню в простых (Simple) экранных формах".

Origin – определяет метод раскладки картинки экранной формы. При отсутствии данного параметра картинка подгоняется под размер рабочей области окна, возможно, с нарушением пропорций по вертикали и горизонтали. Достигается полное использование

рабочей области окна. Наличие параметра Origin задает вывод картинке без нарушения пропорций. Круги, квадраты и т.д. будут оставаться кругами и квадратами при произвольном изменении размеров окна стандартными средствами Windows. Возможны потери рабочей области окна по краям выводимой картинке. При первичной прорисовке центр картинке совмещается с центром рабочей области окна. Масштаб картинке подбирается с целью оптимизации подгонки под размер рабочей области.

Далее могут присутствовать необязательные предложения с ключевым словом Library. Каждое предложение со словом Library в качестве параметра содержит имя файла внешней библиотеки, используемой в данном проекте. Проект может использовать более одной внешней библиотеки.

Раздел Base может иметь предложения для управления видом окна экранной формы. В их число входят:

1) MainMenu [Off, On]

Например: MainMenu Off

Определяет видимость строки с основным меню окна.

Off – меню отсутствует.

On – меню присутствует. По умолчанию меню присутствует.

2) Tools [Off, On, Arise]

Например: Tools Arise

Определяет видимость панели инструментов.

Off – панель отсутствует.

On – панель присутствует. По умолчанию панель присутствует.

Arise – панель всплывающая. При наличии Arise после открытия окна панель инструментов отсутствует. Однако, переместите курсор мыши к верхнему краю рабочей области окна и панель инструментов появится. Её можно использовать. Далее, переместите курсор мыши ниже панели инструментов и панель исчезнет.

3) StatusBar [Off, On]

Например: StatusBar On

Определяет видимость строки состояний.

Off – строка состояний отсутствует.

On – строка состояний присутствует. По умолчанию строка состояний присутствует.

4) Ground [цвет подложки]

Например: Ground clBlack

Строка предназначена для задания исходного цвета рабочего пространства окна, на котором далее рисуется экранная форма. По умолчанию цвет подложки считается белым.

Примечание.

При наличии в предложении Size признака Simple предложения MainMenu, Tools и StatusBar не оказывают влияния на вид окна экранной формы.

5) Position [Default, Designed, LeftTop, RightTop, RightDown, LeftDown, ScreenCenter]

Например: Position ScreenCenter

Допускается одно предложение с командой Position и одним параметром. Определяет местоположение экранной формы (ЭФ) на мониторе.

В случае многомониторности выбор монитора для вывода ЭФ, без углубления в детали, производится по следующим правилам:

- Все мониторы объединены в единый рабочий стол. Первый монитор единого рабочего стола именуется первичным. Остальные – вторичными.

- ЭФ с номером реального объекта выводится на свободный в данный момент монитор. Обычно ЭФ объекта выводится путем нажатия на кнопку объекта, расположенную на главном окне АРМ. Если все мониторы заняты, то выбирается очередной по порядку монитор для замены его картинке текущей ЭФ.

- ЭФ с отрицательным номером, как правило, используется для ее вызова через команду Mouse путем клика мышкой по определенной области на экране. В этом случае ЭФ выводится под курсором мыши на том же мониторе.

Команда Position выполняется на выбранном для ЭФ мониторе.

Опишем назначения параметров команды Position:

- Default. ЭФ с положительными номерами абонента помещаются на первичном мониторе в центре экрана, а на вторичных – в левом верхнем угле. С отрицательными номерами – под курсором.

- Designed. ЭФ помещается на экране по указанной координате. Координата задается следом за командой Designed. Например:

Position Designed=100/200

Координата задается в единицах разрешения монитора (в пикселях) и является смещением левого верхнего угла ЭФ от левого верхнего угла выбранного монитора. Допускаются только положительные значения координат.

- LeftTop. ЭФ помещается в левом верхнем угле выбранного монитора.

- RightTop. ЭФ помещается в правом верхнем угле выбранного монитора.

- RightDown. ЭФ помещается в правом нижнем угле выбранного монитора.

- LeftDown. ЭФ помещается в левом нижнем угле выбранного монитора.

- ScreenCenter. ЭФ помещается в центре выбранного монитора.

Пример описания раздела Base:

```
Base 12,1,"Насосная станция" // N объекта в АРМ (целое), N листа(целое) и имя объекта
{Size 640/480,Origin // вывод картинки без нарушения пропорций
Library "MyLibrary" // имя внешней библиотеки
Library "YouLibrary" // имя внешней библиотеки
Tools Arise
StatusBar Off
Ground DFFFFFF
Position Designed=100/100
}
```

13.1. Дополнительная экранная форма.

Каждая экранная форма (ЭФ) определяется в разделе Base номером абонента и номером листа. Для одного номера абонента можно создать несколько ЭФ, но с разными номерами листов. Не отрицательные номера абонентов из ЭФ должны присутствовать в таблице Objects БД настроек АРМ. ЭФ с отрицательными номерами абонентов именовются дополнительными. ЭФ предназначены для отображения на экране монитора "живой" картинки поведения объекта, изменяя цвета и формы элементов в соответствии с поступившими в АРМ от объекта сигналами, а также выдавая звуковые сообщения. Пользователь может управлять поведением объекта посредством манипулятора "мышь", воздействуя кнопками на предписанные области ЭФ (через команду Mouse). Дополнительные ЭФ, также, вызываются из ЭФ объекта манипулятором "мышь" посредством команд Mouse или MouseTC и содержат рисунок, не уместившийся на ЭФ исходного объекта, или имеют элементы подтверждения определенных действий (Включить, Отключить), или иные элементы рисунка по требованию заказчика.

В настройках программ Arm.exe и Graph.exe имеются пункты: "Число дополнительных экранных форм". По умолчанию в данном пункте задается единица, что означает наличие на экране не более одной дополнительной ЭФ. При вызове новой дополнительной ЭФ предыдущая удаляется. При задании в пункте "Число дополнительных экранных форм"

числа большего единицы на экран монитора можно вывести несколько дополнительных ЭФ, но не более заданного числа.

Внимание!

При задании в пункте “Число дополнительных экранных форм” числа большего единицы соблюдайте следующие правила:

- номера абонентов дополнительных ЭФ должны соответствовать натуральному ряду убывающих чисел, начиная с -1, образуя ряд: -1, -2, ...
- каждый номер абонента дополнительной ЭФ может составлять набор ЭФ с различными номерами листов. Нумерация листов - натуральный ряд положительных чисел, начиная с единицы.
- абсолютное значение наименьшего номера абонента из ряда дополнительных ЭФ не должно превышать заданное число в пункте “Число дополнительных экранных форм”.
- максимально допустимое число дополнительных ЭФ не должно превышать число 16 (без учета листов).

Примечание.

Если у ранее вызванной на экран ЭФ и вызываемой ЭФ номера абонентов одинаковые, а номера листов разные – произойдет замещение старой ЭФ на новую.

14. Раздел Input.

Содержит набор переменных, которым задаются значения при запуске программы. Определить параметры запуска программы с разделом Input можно в другой программе перед командой Mouse. Mouse должна быть запрограммирована на: “Вызов программы по имени”. Запускаемая программа с разделом Input должна иметь отрицательный номер абонента и ее экранная форма выводится в дополнительное окно. Значения переменных из раздела Input определяет вызывающая программа посредством включения в нее команд из следующего набора:

- OutInt – передача целых чисел
- OutFloat – передача чисел с плавающей точкой
- OutClr - передача перечня цветов
- OutPoint – передача координат точек
- OutString – передача текстовых строк
- OutIdent – передача идентификаторов
- OutSignTC – передача перечня телесигналов из таблицы TC
- OutSignTI – передача перечня телеизмерений из таблицы TI
- OutSignTY – передача перечня телеуправлений из таблицы TY
- OutSignCN – передача перечня констант из таблицы Const

Команды из данного набора должны располагаться перед командой Mouse и гасятся (теряют свое содержимое) после команды Mouse. Фактический вызов программы произойдет при щелчке мышью на экранной форме вызываемой программы по области, определенной командой Mouse.

Раздел начинается с ключевого слова Input без параметров. Имеет открывающую и закрывающую фигурные скобки. Между скобками могут быть заданы объявления следующих типов переменных:

- Int – целое число
- Float – число с плавающей точкой
- Clr - цвет
- Point – координата точки

String – текстовая строка
Ident – идентификатор
SignTC – телесигнал из таблицы TC
SignTI – телеизмерение из таблицы TI
SignTY – телеуправление из таблицы TY
SignCN – константа из таблицы Const

Синтаксис строки объявления:

[Имя типа переменной] [Перечень имен переменных данного типа через запятую]

Имя переменной - уникальная алфавитно-цифровая последовательность символов для данного проекта. Имена переменных не должны совпадать с ключевыми словами языка АРМ-графика. Перечень ключевых слов см. в приложении Г.

Переменные из раздела Input при составлении программы используются по правилам, определенным для обычных переменных из раздела Var. Единственное отличие переменных раздела Input от переменных раздела Var состоит в задании им значений при загрузке программы вызываемой экранной формы. Порядок следования типов и число переменных должны строго соответствовать порядку написания команд Out... в вызывающей программе.

Для задания имени окна вызываемой экранной формы имеется команда Header с параметром типа String. Команда Header располагается в вызывающей программе до команды Mouse. Гасится после команды Mouse.

Пример применения раздела Input:

```
//Имя файла вызываемой программы: A01
Base -1,1,"Вызываемая программа"
{Size 260/300,Simple}
Input
{String name
 SignTC tc1,tc2
}
Picture
{
...
Text 100/10,name
SignTC tc1,tc2
CallCell 115/100,Sub,"Проба"
...
}

Base 1,1,"Вызывающая программа"
{Size 1270/880 ,Origin}
Picture
{
...
//Описание передаваемых параметров
Header "Вызвана программа A01 N1"
OutString "Тест1"
```

```

OutSignTC 1_121_1,1_121_2
//Описание вызова программы A01 в дополнительное окно поверх текущего
Mouse 500/100,560/120, 1, A01
...
//Еще раз определим вызов, но с другими параметрами
Header "Вызвана программа A01 N2"
OutString "Тест2"
OutSignTC 1_121_11,1_121_12
//Область вызова на 100 единиц ниже предыдущего
Mouse 600/100,660/120, 1, A01
...
}

```

Команда MouseTC, также, позволяет передать параметры в вызываемую экранную форму. Ввиду того, что команда MouseTC содержит пару имен экранных форм, перед ней, при необходимости, указывают два комплекта команд передачи параметров. Для первой экранной формы используются команды из перечня: Header, OutInt, OutFloat, OutClr, OutPoint, OutString, OutIdent, OutSignTC, OutSignTI, OutSignTY, OutSignCN. Для второй экранной формы используются команды из перечня: Header1, OutInt1, OutFloat1, OutClr1, OutPoint1, OutString1, OutIdent1, OutSignTC1, OutSignTI1, OutSignTY1, OutSignCN1. Правила применения и синтаксис команд из второго перечня идентичны описанным ранее командам передачи параметров (имена которых перечислены в первом перечне).

Пример передачи параметров через команду MouseTC:

```

//Передаваемые параметры в KotelOn
Header "Включить котел"
OutTY 1_2_100
//Передаваемые параметры в KotelOff
Header1 "Отключить котел"
OutTY1 1_2_101
//Состояния TC (1_2_3): 0 – котел отключен, 1 – котел включен
MouseTC 100/10,200/30,1_2_3,KotelOn,KotelOff

```

15. Раздел Picture.

Раздел начинается с ключевого слова Picture без параметров. Имеет следующий набор элементарных графических команд:

15.1. Pen – задает ширину всех последующих примитивов черчения до новой команды Pen.

Синтаксис команды: Pen (ширина)
 Типы параметров: Pen *Int*

По умолчанию предполагается Pen 1.

Допускается задание Pen 0. При нулевой ширине команды, рисующие замкнутые контуры с закраской, не выводят обрамления. К их числу относятся команды, изображающие прямоугольники, эллипсы, Chord, Pie, RoundRect и полигоны. Нарисованные фигуры представляются однотонными. Команды рисующие линии при задании Pen 0 изображают линии единичной ширины при любом масштабе выводимой картинке.

Пример: Pen 5

Примечание.

Синтаксис большинства команд подразумевает прямое или косвенное задание координат изображаемых фигур. В большинстве случаев координаты должны кодировать средние линии обрамления (контура) фигуры, а не ее предельные значения. Ширина обрамления задается командой `Pen`. Например, последовательность команд

```
Pen 6
```

```
Rect 10/10,20/20
```

изобразит прямоугольник с координатами верхнего левого угла - 13/13 и нижнего правого угла - 17/17. Координаты верхнего левого угла всей фигуры с учетом обрамляющего контура – 7/7, а нижнего угла – 23/23.

15.2. Brush – задает цвет закраски замкнутых областей, рисуемых командами: `Chord`, `Ellipse`, `Pie`, `Rect`, `RoundRect`, `Polygon`, `Button`. Используется в командах: `TableTI`, `TabteTC`, `TabteCN`, `ArrowTI`, `MeterTI`, `LevelTI`, `LevelCN`, `GraphTI`. Действует до появления следующей команды `Brush`.

Синтаксис команды: `Brush` [имя константы]

Типы параметров: `Brush` *Clr*

Перечень констант цвета см. в приложении Б.

Пример: `Brush clBlue` `Brush ff0000`

15.3. Color - задает цвет всех последующих примитивов черчения до новой команды `Color`.

Синтаксис команды: `Color` [имя константы]

Типы параметров: `Color` *Clr*

Перечень констант цвета см. в приложении Б.

Пример: `Color clRed` `Color 0000ff`

15.4. Font – задает шрифт и высоту текста для последующего применения в командах вывода текста (`Text`, `TextTC`, `STextTC`, `MultiTextTC`, `TableTC`, `LevelTI`, `ArrowTI`, ...). Действует до появления следующей команды `Font`.

Синтаксис команды: `Font` (высота в пикселах),[имя фонта]

Типы параметров: `Font` *Int,Ident*

Пример: `Font 8, Courier`

15.5. FontStyle – задает набор специальных характеристик текста для последующего применения в командах вывода текста. Действует до появления следующей команды `FontStyle`. Число параметров – произвольное.

Синтаксис команды: `FontStyle` [Стиль текста 1],[Стиль текста 2],...

Типы параметров: `Brush` *Ident,Ident,...*

Стиль текста – идентификатор, выбираемый из перечня, приведенного в приложении В. При отсутствии параметров производится сброс стилей в состояние по умолчанию.

Пример: `FontStyle fsBold, fsItalic, fsUnderline`

15.6. Format – задает формат вывода чисел с плавающей запятой и подгонки текста в специализированных командах TableTI, TableTC, TableCN, ValueCN и ValueTI. Действует до появления следующей команды Format.

Синтаксис команды:

Format (Общее число знаков),(Число знаков после запятой),(Отступ)[Правило подгонки текста]

Типы параметров: Format *Int,Int,[Int][Ident]*

Общее число знаков включает и десятичную точку. По умолчанию задается общее число знаков – 8, а число знаков после запятой – 2.

Отступ – необязательный параметр. Используется со следующими правилами подгонки текста: trR, trL (см. ниже). Задает для выводимых текстов или числовых значений отступы от краев ячеек. Отступ задается в проектных единицах экранной формы. По умолчанию (при его отсутствии) отступ равен нулю. Используется в командах TableTI, TableTC, TableCN.

Для trR отступ отсчитывается от правого края ячейки.

Для trL отступ отсчитывается от левого края ячейки.

Правило подгонки текста – необязательный параметр. Используется в командах TableTI, TableTC, TableCN, ValueTI, ValueCN. Принимает значения:

trR – заголовки и числовые значения выводимых в таблицу телеизмерений подгоняются по правым краям клеток таблицы;

trC подгоняются по центрам клеток;

trL - подгоняются по левым краям клеток.

По умолчанию имеет значение trL.

Пример: Format 8,2,4trR

Максимальное число, выведенное по формату из примера, имеет вид: 99999.99

15.7. ColorAvar – определяет цвет мигания в специальных командах при нахождении сигнала команды в аварийном состоянии. Мигание прекращается при квитировании аварии. Действует до появления следующей команды ColorAvar. По умолчанию ColorAvar имеет цвет clWhite.

Синтаксис команды: ColorAvar [имя константы]

Типы параметров: ColorAvar *Clr*

Перечень констант цвета см. в приложении Б.

Пример: ColorAvar clWhite ColorAvar 0000ff

15.8. ColorDisable – определяет цвет закраски в специальных командах при нахождении сигнала команды в неопределенном состоянии. Действует до появления следующей команды ColorDisable. По умолчанию ColorDisable имеет цвет clLtGray.

Синтаксис команды: ColorDisable [имя константы]

Типы параметров: ColorDisable *Clr*

Перечень констант цвета см. в приложении Б.

Пример: ColorDisable clGray

15.9. ColorBlink – определяет цвет мигания в ряде специальных команд при наличии у телесигнала мигания указанного значения. Сигналы мигания задаются командами SignalBlinks и StateBlinks. Мигание прекращается при переходе сигнала мигания к противоположному значению. Действует до появления следующей команды ColorBlink. По умолчанию ColorBlink имеет цвет c!White.

Синтаксис команды: ColorBlink [имя константы]

Типы параметров: ColorBlink *Clr*

Перечень констант цвета см. в приложении Б.

Пример: ColorBlink c!White

15.10. Vid – задает вид изображения для команд:

Arc, Rect, RoundRect, Ellipse, Polyline, Button, PolylineTC, ButtonTC, RoundButtonTC, ButtonTI, RoundButtonTI, SPolylineTC, SButtonTC, SRoundButtonTC, RectTC, RoundRectTC, PolylineTI, RectTI, RoundRectTI, EllipseTC, EllipseTI, TableTC, TableTI, TableCN, GraphTI, SRectTC, SRoundRectTC, SEllipseTC, ArrowTI, MeterTI. Действует до появления следующей команды Vid.

Синтаксис команды: Vid [имя вида]

Типы параметров: Vid *Ident*

Имя вида принимает одно из следующих значений: 2D, 3D, Brd или Border.

2D – плоские изображения картинок от перечисленных команд.

3D – выпуклые изображения.

Brd (или Border) – контур изображения имеет вид рельефного бордюра.

По умолчанию предполагается 3D.

Пример: Vid 2D

Объемное изображение.

Объемной изображением будем считать трехмерный (выпуклый) объект. Источник освещения объекта расположен над ним. Для включения режима прорисовки объемных изображений задайте в файле конфигурации предложение volume=yes.

Команды Polyline, PolylineTC, SPolylineTC, PolylineTI в режиме Vid 3D рисуют объемные отрезки полилинии, а в режиме Vid Brd отрезки имеют вид рельефного бордюра. Команды Chord, Pie, Ellipse, EllipseTC, EllipseTI, SEllipseTC рисуют объемные фигуры, если перед ними задать Pen 0 и Vid 3D.

Для сохранения преемственности с заделом можно отключить объемную прорисовку этих команд, задав в файле конфигурации предложение volume=no.

15.11. Scale – масштабирует выводимое изображение.

Синтаксис команды:

Scale (коэффициент масштабирования по оси X), (коэффициент масштабир. по оси Y)

Типы параметров: Scale *Int,Int*

Коэффициент масштабирования – число с плавающей запятой. Все последующие размеры и координаты во всех командах умножаются на заданные коэффициенты масштабирования до появления новой команды Scale. По умолчанию коэффициент

масштабирования равен единице. Высота шрифта и ширина пера умножаются на коэффициент масштабирования по оси Y.

Пример: Scale 1.5,1.75

Примечание

В разделе Base параметру Size должен задаваться размер экранной формы с учетом масштабирования изображения. Например, для программы вида:

```
Base ...
{Size 600/400...
Picture
{
...
}
```

введем масштабирование, увеличивающее изображение в полтора раза. В этом случае программа должна иметь вид:

```
Base ...
{Size 900.600... //увеличен в полтора раза
Picture
{Scale 1.5,1.5
...
}
```

15.12. Angle – задает последующим командам Text, TextTC, TextTI, ValueTI, DateValueTI, TimeValueTI, Button, ButtonTC, RoundButtonTC, ButtonTI, RoundButtonTI, ButtonTCTI, RoundButtonTCTI, SButtonTC, SRoundButtonTC угол поворота выводимых текстовых строк. По умолчанию угол поворота считается равным нулю. Перед командами Button, ButtonTC, RoundButtonTC, ButtonTI, RoundButtonTI, SButtonTC, SRoundButtonTC допускаются значения 0 или 90.

Синтаксис команды:

Angle (угол поворота в градусах)

Угол поворота отсчитывается от оси X против часовой стрелки.

Типы параметров: Angle *Int*

Пример: Angle 45

15.13. Style - управляет режимом рисования линий в команде Polyline или контуров в командах Polygon и Rect.

Синтаксис команды: Style [значение]

Типы параметров: Style *Ident*

Принимает значения:

psSolid - сплошные линии

psDash - штриховые линии

Пример: Style psDash

15.14. Trim – определяет подгонку (выравнивание) строк многострочных текстов при их выводе на экранную форму командами Text, TextTC, TextTI. Напомним, что строки в тексте разделяются парой символов: “\n” (без кавычек). На однострочный текст команда Trim влияния не оказывает.

Синтаксис команды: Trim [значение]

Типы параметров: Trim *Ident*

Принимает значения:

trR – строки текста подгоняются по правым краям;

trC – строки текста подгоняются по центрам;
trL – строки текста подгоняются по левым краям.
По умолчанию имеет значение trL.

Пример:
Trim trR
Angle 45
Text “Выравнивание двух строк\ппо правому краю”

15.15. Arc – рисует дугу окружности или эллипса.

Синтаксис команды: Arc (x1)/(y1),(x2)/(y2),(x3)/(y3),(x4)/(y4)
Типы параметров: Arc *Point,Point,Point,Point*

Рисует периметр эллипса против часовой стрелки от начальной до конечной точки. Эллипс вписывается в прямоугольник с координатами углов Arc (x1)/(y1),(x2)/(y2). Начальная точка располагается в точке пересечения эллипса с прямой, проходящей через центр эллипса и точку (x3)/(y3). Конечная точка располагается в точке пересечения эллипса с прямой, проходящей через центр эллипса и точку (x4)/(y4). . Ширина и цвет дуги предварительно задаются командами Pen, Color. Команда Vid определяет вид контура. При наличии команды Vid Brd рекомендуется задать Pen не менее 3.
Пример: Arc 10/10,20/20,30/30,40/40

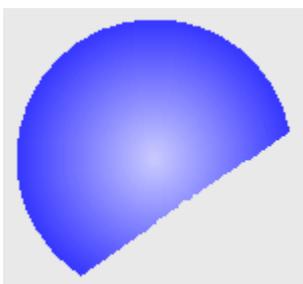
15.16. Chord – рисует замкнутую закрашенную фигуру, ограниченную дугой окружности или эллипса и хордой.

Синтаксис команды: Chord (x1)/(y1),(x2)/(y2),(x3)/(y3),(x4)/(y4)
Типы параметров: Chord *Point,Point,Point,Point*

Хорда состоит из части эллипса, который вписан в прямоугольник с координатами (x1)/(y1),(x2)/(y2). Эллипс усекается линией, которая проходит через точки (x3)/(y3),(x4)/(y4). Периметр эллипса пробегает против часовой стрелки от (x3)/(y3) вдоль эллипса до (x4)/(y4) и прямо назад в (x3)/(y3). Если (x3)/(y3),(x4)/(y4) не на поверхности эллипса, то соответствующими углами на хорде являются ближайшие точки на периметре, которые пересекают линию. Ширина контура, цвет контура и цвет заливки предварительно задаются командами Pen, Color и Brush.

Chord можно нарисовать объемным, если перед командой задать Pen=0 и Vid 3d, а в файл конфигурации внести предложение volume=yes.

Пример:
Pen 0
Vid 3D
Brush ff8282
Chord 235/181,379/325,377/238,270/313



15.17. Ellipse – рисует закрашенную окружность или эллипс.

Синтаксис команды: `Ellipse (x1)/(y1),(x2)/(y2)`

Типы параметров: `Ellipse Point,Point`

$(x1)/(y1)$ – координата верхнего левого угла, $(x2)/(y2)$ – координата нижнего правого угла. Ширина контура, цвет контура и цвет закрашки предварительно задаются командами `Pen`, `Color` и `Brush`. Команда `Vid` определяет вид контура. При наличии команды `Vid Brd` рекомендуется задать `Pen` не менее 3.

Эллипс можно нарисовать объемным, если перед командой задать `Pen=0` и `Vid 3d`, а в файл конфигурации внести предложение `volume=yes`.

Не закрашенную окружность или эллипс следует рисовать командой `Arc` с совпадающими начальной и конечной точками.

Пример:

```
Color clGray
```

```
Pen 6
```

```
Vid Brd
```

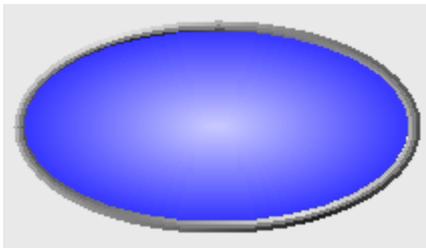
```
Arc 17/227,223/333,40/280,40/280 //обрамление
```

```
Pen 0
```

```
Vid 3D
```

```
Brush ff8282
```

```
Ellipse 20/230,220/330 //выпуклый эллипс
```



15.18. Pie – рисует закрашенный сектор окружности или эллипса, вписанного в прямоугольник с координатами $(x1)/(y1)$, $(x2)/(y2)$. Рисунок секции определяется эллипсом, ограниченным двумя линиями, исходящими из центра эллипса через точки $(x3)/(y3)$ и $(x4)/(y4)$.

Синтаксис команды: `Pie (x1)/(y1),(x2)/(y2),(x3)/(y3),(x4)/(y4)`

Типы параметров: `Pie Point,Point,Point,Point`

Ширина контура, цвет контура и цвет закрашки предварительно задаются командами `Pen`, `Color` и `Brush`.

`Pie` можно нарисовать объемным, если перед командой задать `Pen=0` и `Vid 3d`, а в файл конфигурации внести предложение `volume=yes`.

Пример:

```
Pen 0
```

```
Vid 3D
```

```
Brush ff8282
```

```
Pie 26/92,100/166,100/94,26/94
```

```
Pie 26/107,101/182,26/108,101/108
```



15.19. Polygon – рисует замкнутую закрашенную фигуру с кусочно-линейной границей.

Синтаксис команды:

Polygon - команда без параметров

До команды Polygon должна присутствовать команда Points.

Синтаксис команды **Points**:

Points (x1)/(y1),(x2)/(y2) ... - перечень координат точек.

Типы параметров: Points *Point,Point*, ...

Ширина контура, цвет контура и цвет закрашки предварительно задаются командами Pen, Color и Brush. Стил контура задает команда Style. При задании стиля psDash линии контура рисуются как штриховые, если они параллельны осям координат. Иначе - как сплошные линии. Цвет промежутков в штриховых линиях определяется предшествующей командой Brush.

Пример:

Style psSolid

Points 10/20,10/40,20/20

Polygon // нарисует треугольник

15.20. Polyline – рисует кусочно-линейную кривую.

Синтаксис команды:

Polyline - команда без параметров

До команды Polyline должна присутствовать команда Points.

Синтаксис команды Points:

Points (x1)/(y1),(x2)/(y2) ... - перечень координат точек.

Типы параметров: Points *Point,Point*, ...

Ширина и цвет линий предварительно задаются командами Pen и Color. Стил линий задает команда Style. Она учитывается при прорисовке, если задан Vid 2D. В этом случае при задании стиля psDash линии рисуются как штриховые, если они параллельны осям координат. Иначе - как сплошные линии. Цвет промежутков в штриховых линиях определяется предшествующей командой Brush.

Вид Polyline определяет команда Vid.

1) При задании Vid 2D или наличии в файле конфигурации предложения volume=no команда Polyline рисуется круглой маской, диаметр которой задает команда Pen. Углы наклонов составляющих ее отрезков произвольны. Исполняет команду Style при ее наличии.

2) При задании Vid 3D команда Polyline рисуется как объемная фигура.

3) При задании Vid Brd команда Polyline имеет вид рельефного бордюра.

В случаях 2) и 3) имеем следующие особенности:

- В файле конфигурации необходимо наличие команды volume=yes.

- Команда Polyline сглаживает углы между составляющими ее отрезками. Параметры сглаживания задает первый параметр команды Lines.
- Составляющие Polyline отрезки (кодируются в предшествующей команде Points) должны быть параллельны осям координат. При нарушении этого правила рисование происходит в режиме Vid 2D без учета команды Lines.
- Если в Points первая и последняя координаты совпадают, то происходит их сглаживание в соответствии с командой Lines и имеем замкнутую сглаженную фигуру во всех ее углах.
- Конфигурацию концов Polyline определяют второй и третий параметры команды Lines, которые задают глубину проникновения (врезки) начала (конца) полилинии в другую соприкасающуюся полилинию. При нулевых значениях параметров врезок полилинии имеют ровные окончания.

Пример:

Color c1Blue

Pen 20

Vid 2D

Points 320/90,380/90,380/144,320/144

Polyline

Vid 3D

Lines 30/30,0,0

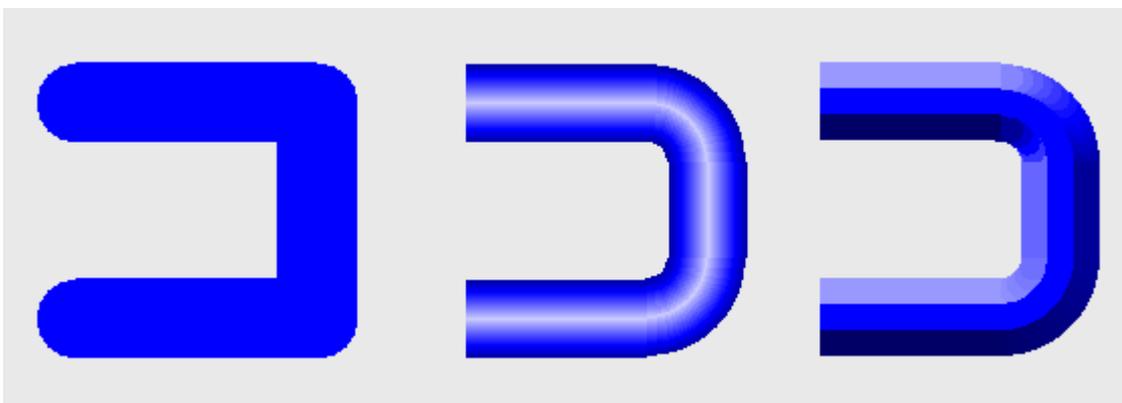
Points 417/90,477/90,477/144,417/144

Polyline

Points 505/90,565/90,565/144,505/144

Vid Brd

Polyline



С целью поддержки задела допускается отключить прорисовку Polyline в режимах Vid 3D и Vid Brd, всегда рисуя в режиме Vid 2D вне зависимости от заданного Vid. Отключение производится в конфигураторах графического редактора (Graph.exe) и АРМ (Arm.exe).

15.21. Rect – рисует закрашенный прямоугольник.

Синтаксис команды: `Rect (x1)/(y1),(x2)/(y2)`

Типы параметров: `Rect Point,Point`

$(x1)/(y1)$ – координата верхнего левого угла, $(x2)/(y2)$ – координата нижнего правого угла. Ширина контура, цвет контура и цвет закрашки предварительно задаются командами `Pen`, `Color` и `Brush`. Предварительно заданная команда `Vid` определяет вид контура. При наличии команды `Vid Brd` рекомендуется задать `Pen` не менее 3. Стиль контура задает команда `Style`. При задании стиля `psDash` линии контура рисуются как штриховые. Цвет промежутков в штриховых линиях определяется предшествующей командой `Brush`.
Пример: `Rect 10/20,10/40`

15.22. RoundRect – рисует закрашенный прямоугольник со скругленными углами.

Синтаксис команды: `RoundRect (x1)/(y1),(x2)/(y2),(x3)/(y3)`

Типы параметров: `RoundRect Point,Point,Point`

$(x1)/(y1)$ – координата верхнего левого угла, $(x2)/(y2)$ – координата нижнего правого угла. Кривая скругленных углов совпадает с кривой эллипса с шириной $X3$ и высотой $Y3$. Ширина контура, цвет контура и цвет закрашки предварительно задаются командами `Pen`, `Color` и `Brush`. Предварительно заданная команда `Vid` определяет вид контура. При наличии команды `Vid Brd` рекомендуется задать `Pen` не менее 3. Если `Pen` равен нулю – контур отсутствует.
Пример: `RoundRect 10/20,10/40,5/5`

15.23. Text - рисует текст.

Синтаксис команды: `Text (x1)/(y1),<текст>`

Типы параметров: `Text Point,String`

$(x1)/(y1)$ – координата верхнего левого угла первого символа текста. Цвет фона и угол поворота текста предварительно задаются командами `Color` и `Angle`. Текст может состоять из множества строк, разделяемых символами “\n” (без кавычек). Число строк в тексте – не более 10-и. Можно задать способ подгонки строк командой `Trim`.

Пример:

`Angle 45`

`Color clBlack`

`Trim trC`

`Text 338/205,"Текст 45\nградусов" // вывод двух строк`

15.24. Image – рисует изображение, записанное в файле.

Синтаксис команды: `Image (x1)/(y1),(x2)/(y2),<имя файла>`

Типы параметров: `Image Point,Point,String`

Команда `Image` настраивается на вывод по расширению имени файла. Имя файла – без пути. Путь задается в настройках.

(x1)/(y1),(x2)/(y2) – координаты прямоугольника для вывода картинка. (x1)/(y1) – верхний левый угол прямоугольника. (x2)/(y2) – нижний правый угол прямоугольника.

Допустимые расширения:

Bmp – битовая матрица;

Jpg – сжатое изображение по протоколу JPEG;

Ico – пиктограмма;

Wmf – метафайл (16-и разрядный);

Emf – метафайл (32-х разрядный).

Пример: Image 10/20,400/300,"myPicture.bmp"

15.25. CallCell - вызов библиотечной ячейки.

Синтаксис команды:

CallCell (x1)/(y1),[тип библиотечной ячейки],< имя ячейки >

Типы параметров: CallCell *Point,Ident,String*

(x1)/(y1) - координаты местоположения базовой точки библиотечной ячейки на листе.

Рекомендуется считать базовой точкой левый верхний угол полного изображения ячейки.

В этом случае координаты графических составляющих ячейки в ее описании будут положительными.

Тип библиотечной ячейки - ссылочное имя. Описание выбирается по ссылочному имени из внутренней библиотеки, а при его отсутствии – из внешних библиотек, перечисленных в разделе Base командой Library. Ссылочное имя не должно быть длиннее 19-ти символов. Имя ячейки используется во всплывающей подсказке и не должно быть длиннее 79-ти символов.

Для передачи параметров в вызываемую подпрограмму команде CallCell должны присутствовать описания фактических параметров в виде команд: Int, Float, Clr, Point, String, Ident, SignTC, SignTI, SignTY, SignCN. Эти команды предназначены для группировки фактических параметров по типам. Фактическим параметром является конкретное значение, например, целое число, координата точки, идентификатор телесигнала и т.д. Область определения – раздел Picture. Формальным параметром является произвольный идентификатор, область определения которого ограничивается подпрограммой, в которой он задан. Формальные параметры рекомендуется размещать в начале подпрограммы с помощью того же набора команд: Int, Float, Clr, Point, String, Ident, SignTC, SignTI, SignTY, SignCN.

Для каждого типа число и порядок следования фактических и формальных параметров в ячейке должны быть одинаковыми. Состав параметров определяется алгоритмом работы подпрограммы. Для команд SignTC и SignTI допускаются исключения, Если ячейка имеет число формальных сигналов большее, чем требуется для данного вызова команды CallCell, то неиспользуемые сигналы задаются в SignTC и SignTI зарезервированным словом "null".

В командах SignTC, SignTI, SignTY, SignCN допускается задавать не имена фактических сигналов, а их значения. В командах SignTC, SignTY значениями являются числа 0 или 1. В командах SignTI, SignCN значениями являются действительные числа с плавающей точкой, например: -.5 или 12.37. При выполнении подпрограммы соответствующему формальному параметру передается указанное значение, которое не меняется на протяжении работы приложения. Применение значений вместо имен фактических сигналов оправданно на начальном этапе разработки программы экранной формы, когда БД настроек АРМ еще не содержит полного перечня сигналов в таблицах

ТС и ТI. По мере заполнения БД настроек АРМ необходимо избавляться от описания сигналов их значениями.

Примечание.

Команды Pen, Brush, Color, Font, FontStyle, Angle, Format, ColorAvar, ColorDisable, ColorBlink, Vid, Scale, Menu, Points, PassТС изменяют текущие значения одноименных внутренних переменных при исполнении программы АРМ-графика. До начала исполнения этим переменным задаются значения по умолчанию.

Смена значений Pen, Brush, Color, Font, FontStyle, Angle, Format, ColorAvar, ColorDisable, ColorBlink, Vid, Scale, Menu, Points, PassТС в вызванной ячейке не изменяет состояния перечисленных переменных в разделе Picture после завершения подпрограммы, но не наоборот. Действующие значения всех внутренних переменных из Picture передаются в вызванную подпрограмму без изменений.

15.26. Int - перечень целочисленных значений. Int предназначен для описания замен формальных параметров вызываемой библиотечной ячейки фактическими значениями.

Синтаксис команды:

Int [1-е число],[2-е число] ... - перечень чисел

Типы параметров: Int *Int,Int*, ...

Пример: Int 1,98

15.27. Float - перечень чисел с плавающей точкой. Float предназначен для описания замен формальных параметров вызываемой библиотечной ячейки фактическими значениями.

Синтаксис команды:

Float [1-е число],[2-е число] ... - перечень чисел

Типы параметров: Float *Float,Float*, ...

Пример: Float 12.34,987.654

15.28. Clr - перечень цветов. Clr предназначен для описания замен формальных параметров вызываемой библиотечной ячейки фактическими значениями.

Синтаксис команды:

Clr [1-й цвет],[2-й цвет] ... - перечень цветов

Типы параметров: Clr *Clr,Clr*, ...

Пример: Clr clRed,fff

15.29. Point - перечень координат точек. Point предназначен для описания замен формальных параметров вызываемой библиотечной ячейки фактическими значениями.

Синтаксис команды:

Point [1-я точка],[2-я точка] ... - перечень точек

Типы параметров: Point *Point,Point*, ...

Пример: Point 12/34,98/54

15.30. String - перечень текстовых строк. String предназначен для описания замен формальных параметров вызываемой библиотечной ячейки фактическими значениями.

Синтаксис команды:

String [1-й текст],[2-й текст] ... - перечень строк

Типы параметров: String *String,String*,...

Каждый текст может состоять из множества строк, разделяемых символами “\n” (без кавычек). Число строк в тексте – не более 10-и.

Пример: String “Правый\nугол”,”Левый\nугол”

15.31. Ident - перечень идентификаторов. Ident предназначен для описания замен формальных параметров вызываемой библиотечной ячейки фактическими значениями.

Синтаксис команды:

Ident [1-й идент.],[2-й идент.] ... - перечень идентификаторов

Типы параметров: Ident *Ident,Ident*,...

Пример: Ident name

15.32. SignTC - перечень сигналов таблицы TC (фактические сигналы).
Перечень сигналов предназначен для замены формальных сигналов вызываемой библиотечной ячейки фактическими сигналами.

Синтаксис команды:

SignTC [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров: SignTC *SignTC,SignTC*,...

Структура имен сигналов имеет вид:

(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Перечисленные в команде сигналы должны быть описаны в таблице TC базы данных настроек АРМ. Работа команд зависит от описаний сигналов в таблице TC.

Для состояния связи с объектом определены зарезервированные имена телесигналов Link с номером объекта. Сигналы Link не описывают в таблице TC.

Вместо имен сигналов допускается применять зарезервированное слово “null” или указать значение сигнала: 0 или 1.

Примеры:

SignTC 1_-1_62,1_1_33,Link22

SignTC 1,0,null // перечень значений телесигналов и указание на отсутствие сигнала

15.33. SignTI - перечень сигналов таблицы TI (фактические сигналы)

Типы параметров: SignTI *SignTI,SignTI*,...

Синтаксис и структура команды SignTI идентичны описанным для команды SignTC за исключением имен состояния связи Link . Перечисленные в команде сигналы должны быть описаны в таблице TI базы данных настроек АРМ. Работа команд зависит от описаний сигналов в таблице TI

Вместо имен сигналов допускается применять зарезервированное слово “null” или указать значение сигнала.

Пример: SignTI 1.25,-2.75 // перечень значений телеизмерений

Еще пример:

SignTC 1_1_62,1_1_33,null // перечень сигналов таблицы TC (фактические сигналы)

SignTI 1_1_100,1_2_77,null // перечень сигналов таблицы TI (фактические сигналы)

CallCell 10/20,1.5,FD5,“ФидерA5”

Команды Menu и Mouse определяют область и реакцию системы при нажатии правой и левой кнопок мыши соответственно. Для выполнения действия курсор мыши должен находиться в пределах заданной области. Команда Menu должна предшествовать команде Mouse.

15.34. SignTY - перечень сигналов телеуправлений из таблицы TY (фактические сигналы).

Перечень сигналов предназначен для замены формальных сигналов вызываемой библиотечной ячейки фактическими сигналами.

Синтаксис команды:

SignTY [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров: SignTY *SignTY,SignTY*,...

Структура имен сигналов имеет вид:

(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Перечисленные в команде сигналы должны быть описаны в таблице TY базы данных настроек АРМ. Работа команд зависит от описаний сигналов в таблице TY.

Вместо имен сигналов допускается применять зарезервированное слово “null” или указать значение сигнала: 0 или 1.

Пример:

SignTY 1_2_33

15.35. SignCN - перечень констант из таблицы Const (фактические сигналы)

Типы параметров: SignCN *SignCN,SignCN*,...

Синтаксис и структура команды SignCN идентичны описанным для команды SignTI.

Перечисленные в команде сигналы должны быть описаны в таблице Const базы данных настроек АРМ. Работа команд зависит от описаний сигналов в таблице Const

Вместо имен сигналов допускается применять зарезервированное слово “null” или указать значение сигнала.

Пример: SignCN 1_2_10,1_2_77,null // перечень сигналов таблицы Const (фактические сигналы)

15.36. Mouse - определяет область и типы реакции при нажатии левой кнопки мыши. Выбор действия возможен из следующего перечня:

0. Нет действий
1. Перейти к экранной форме объекта по имени файла
2. Телеуправление

3. Изменение значения ТС
4. Изменение значения ТІ
5. Макрокоманда
6. Вывод Метод с текстом
7. Изменение значения Const
8. Управление через SendТС
9. Закрывать экранную форму
10. Запрос данных для Альфа
11. Блокировка ТС
12. Блокировка ТІ

Синтаксис команды Mouse:

Mouse (x1)/(y1),(x2)/(y2),(номер действия),[имя],[значение]

Типы параметров: Mouse Point, Point, Int, Смотри таб. 1, Int или float

Где: (x1)/(y1),(x2)/(y2) - координаты углов области клика

Имя – зависит от значения третьего параметра “номер действия”. Правила записи имен для различных номеров действий приведены в таблице 1.

0) Действие "Нет действий" предназначено для описания реакции на правую кнопку мыши без побочных эффектов от левой кнопки. Целесообразно поместить в конце раздела Picture, указав в качестве области клика всю экранную форму.

1) Действие “Перейти к экранной форме объекта по имени файла”. Выводит экранную форму в одно из двух независимых окон отображения экранных форм. Вывод формы в первое окно производится, если NumAbon для выводимого файла в таблице Graphics имеет неотрицательное число, и во второе окно при отрицательном значении. Предусмотрен механизм передачи параметров в вызываемую экранную форму посредством команд Header, OutInt, OutString, и т.д.

Если имя из команды Mouse является ссылкой на файл одного из листов текущей (видимой) ЭФ, то экранная форма выводится в это же окно, даже если ЭФ была внедренной. Напоминаем, что листы ЭФ являются отдельными файлами и должны иметь в строках Base одинаковый номер абонента, но разные номера листов.

2) Действие “Телеуправление” отправляет команду через сервер соответствующему контроллеру. Идентификатор телеуправления указан в поле “Имя”. Контроллер исполняет полученную команду. Значение телеуправления (0 или 1) должно быть указано в поле “Значение” и является обязательным. Завершив исполнение, контроллер отправляет в сервер, а далее и в АРМ, исполненное телеуправление (эхо-сигнал).

3), 4), 7) Действия “Изменение телесигнала”, “Изменение телеизмерения” и “Изменение константы” исполняют команду смены значения параметра, идентификатор которого указан в поле “Имя”. Команда через сервер отправляется на соответствующий контроллер, где и исполняется, изменяя параметр. После исполнения, новое значение параметра (телесигнала, телеизмерения или константы) отправляется серверу и далее в АРМ (эхо-сигнал).

Поле “Значение” определяет число, на которое будет изменен параметр. Не является обязательным. Если поле не задано, то перед исполнением команды выводится диалоговое окно для ручного ввода значения. Действие “Изменение телесигнала” может принимать значения : 0 или 1. Значение для действий “Изменение телеизмерения” и “Изменение константы” является целым или действительным числом.

5) Действие “Макрокоманда“ в качестве параметра содержит номер макрокоманды из таблицы Macro. Напоминаем, что каждая макрокоманда содержит перечень команд в таблице Commands. При запуске Mouse с данным действием все предписанные команды будут выполнены, а ход выполнения команд отобразится в специальном окне.

6) Действие “Вывод Мемо“ зависит от типа кодировки файла. Имя файла задается в параметре ‘имя’. Кодировка определяется расширением в имени файла. Приводит к появлению окна в стиле “Мемо” или вызывает приложение стороннего разработчика для отображения данных из файла. Допускаются следующие виды кодировок:

■ Кодировка текста Windows. Расширение файла - txt. Содержат только текст.

Тексты из файлов выводятся на экран в окно простейшего вида (стиль “Мемо”).

■ Кодировка в формате rtf. Расширение файла - rtf.

Данные из файлов выводятся на экран в окно простейшего вида (стиль “Мемо”), но обеспечивается поддержка стандарта MS Word (разметка текста, таблицы, изображения и т.д.).

■ Кодировка в формате MS Word. Расширение файла – doc или docx.

■ Кодировка в формате MS Excel. Расширение файла – xls или.xlsx.

■ Кодировка в формате Adobe. Расширение файла - pdf.

Данные из файлов с расширениями doc и xls вызывают приложения из пакета MS Office (MS Word и MS Excel соответственно).

8) Действие "Управление через SendTC" не требует задания значения. По данному действию на экран выводится окно, в состав которого входят перечни реакций из таблиц SendTC и CheckTC для указанного в поле “Имя” базового телесигнала.

9) Действие “Закрыть экранную форму”. При клике по указанной области закрывает текущую экранную форму (ЭФ). Пример: Mouse 10/20.110/40,9. Имеет важную особенность. Если расположить эту команду вслед за командой управления и задать им одинаковые области, то при клике по данной области сначала произойдет запуск команды управления с последующим закрытием окна ЭФ. В число поддерживаемых команд управления входят:

2 - Телеуправление,

3 - Изменение телесигнала,

4 - Изменение телеизмерения,

5 – Макрокоманда,

7 - Изменение константы.

Пример программы, поддерживающей данную особенность:

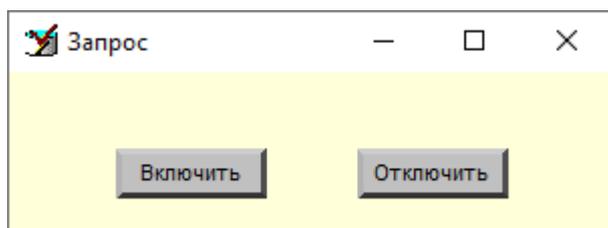
```
Base -1,1,"Запрос"  
{Size 300/80,Simple  
  Ground d9ffff  
}  
Input  
{SignTC tc1  
  SignTY ty1,ty2 //On,Off  
}  
Picture  
{  
  Vid 3D  
  Pen 3  
  Brush clLtGray  
  Font 8,Arial
```

```

Color clGray
PassTC tc1,0 //Блокируем по нулю
Mouse 55/40,125/60,2,ty1,1
Mouse 55/40,125/60,9 //Close
Button 55/40,125/60,clBlack,"Включить"
PassTC tc1,1 //Блокируем по единице
Mouse 175/40,245/60,2,ty2,1
Mouse 175/40,245/60,9 //Close
Button 175/40,245/60,clBlack,"Отключить"
PassTC //Не забывайте отключать блокировку
}

```

Вид программы:



10) Действие "Запрос данных для Альфа" вызывает окно "Запрос данных для Альфа ЦЕНТР". Окно содержит единственную строку в таблице счетчиков. Строка параметров команды Mouse должна иметь в качестве четвертого параметра регистрационный номер запрашиваемого счетчика. Пятый параметр отсутствует.

Внимание.

Каждому счетчику производитель присваивает уникальный серийный номер счетчика. Помимо серийного, в БД АРМ и серверов каждый счетчик имеет регистрационный номер счетчика. По мере износа, счетчики меняют. При этом серийный номер меняется, а регистрационный остается неизменным. Раз в сутки автоматически запускается процедура сверки номеров счетчиков в БД АРМ и серверов. Сверка серийных номеров производится по строкам, имеющим одинаковые регистрационные номера счетчиков в разных БД. По результатам замеченных изменений серийных номеров счетчиков БД АРМ корректируется.

11), 12) Действия: "Блокировка ТС", "Блокировка ТГ". Параметрами команды Mouse для этих действий являются: идентификатор сигнала и значение сигнала. Действия предназначены для блокировки данного сигнала и задания ему предписанного значения. Блокировка означает, что поступающие события от внешних источников данных (серверов, контроллеров) не смогут изменить внутреннее значение данного сигнала (в рамках АРМ). Переход к нормальному режиму реакции на внешние события возможен только после разблокировки сигнала. Для этого используйте пункт меню "Настройки/Редактор заблокированных сигналов". Идентификатор сигнала определяется в поле "Имя". Поле "Значение" задает число, на которое будет изменен параметр. Не является обязательным. Если поле не задано, то перед исполнением команды выводится диалоговое окно для ручного ввода значения. Перечень заблокированных сигналов с заданными параметрами сохраняются в таблице Calc, что позволяет восстановить блокировки после перезагрузки АРМ.

Таблица 1. Правила составления параметра 'имя' в команде Mouse.

Номер действия	Имя действия	Синтаксис параметра 'имя'	Тип параметра	Пример
0	Нет действий			
1	Перейти к экранной форме	Имя фала без расширения из таблицы Graphics	Ident	Test1
2	Телеуправление	(N абонента)_(N группы)_(N парам.) из таблицы TY	SignTY	1_2_33
3	Изменение значения ТС	(N абонента)_(N группы)_(N парам.) из таблицы ТС	SignТС	1_2_33
4	Изменение значения ТІ	(N абонента)_(N группы)_(N парам.) из таблицы ТІ	SignТІ	1_2_33
5	Макрокоманда	N макрокоманды из таблицы Macro	Int	1
6	Вывод Мемо	Полное имя файла (с путем и расширением)	Ident	Help.txt Help.rtf
7	Изменение значения Const	(N абонента)_(N группы)_(N парам.) из таблицы Const	SignCN	1_2_33
8	Управление через SendТС	Базовый (N абонента)_(N группы)_(N парам.) из таблицы SendТС	SignТС	1_2_33
9	Закрывать экранную форму			
10	Запрос данных для Альфа	Регистрационный № счетчика	Int	1203456
11	Блокировка ТС	(N абонента)_(N группы)_(N парам.) из таблицы ТС	SignТС	1_2_33
12	Блокировка ТІ	(N абонента)_(N группы)_(N парам.) из таблицы ТІ	SignТІ	1_2_33

Команда Mouse реагирует и на правую кнопку мыши, если ей предшествует команда Menu.

Примечание.

При наличии наложений областей клика от разных команд Mouse и при нажатии кнопки в спорной области работает следующее правило - исполняется первая команда из числа спорных по порядку описания в разделе Picture. Если команда Mouse расположена в подпрограмме, то ее порядком описания является местоположение вызывающей команды CallCell.

15.37. Menu - определяет перечень пунктов контекстного меню. Контекстное меню появляется на экране при нажатии правой кнопки мыши при условии нахождения курсора мыши в пределах заданной области. Области задаются последующими командами Mouse. Фактически, контекстное меню от команды Menu приписывается всем последующим командам Mouse. Состав контекстного меню задается набором параметров команды Menu. Выбор пунктов меню возможен из следующего перечня:

1. Таблица телесигналов экранной формы
2. Таблица телеизмерений экранной формы
3. Графики экранной формы
4. Квитирование
5. Телеуправление
6. Запрос среза КП
7. Изменение параметров (сигналов)
8. Свойства
9. Таблица аварий экранной формы
10. Таблица предупреждений экранной формы
11. Таблица телесигналов объекта
12. Таблица телеизмерений объекта
13. Графики объекта
14. Таблица аварий объекта
15. Таблица предупреждений объекта
16. Блокнот
17. Запрос данных для Альфа
18. Графики телесигналов экранной формы
19. Графики телеизмерений экранной формы
20. Осциллограммы
21. Хронологическая таблица экранной формы
22. Таблица текущих значений экранной формы

Если выбранный пункт меню имеет вид: "... экранной формы", то при выводе таблиц в них включаются только те сигналы, аварии или предупреждения, которые относятся к ячейкам, расположенным под курсором мыши в момент ее нажатия. Если под курсором нет ячеек, то в таблицы включаются все сигналы, аварии или предупреждения, которые относятся к данной экранной форме (упоминались при описании формы).

Если пункт меню имеет вид: "... объекта", то в таблицы включаются все сигналы, аварии или предупреждения объекта, которому принадлежит экранная форма.

Пункт меню "Блокнот" выводит на экран окно с текстом. Содержимое текста является контекстно зависимым, т.е. относится к той экранной форме и ячейке, на которой была нажата правая кнопка мыши. Текст можно изменять. При закрытии блокнота текст будет сохранен с учетом всех внесенных изменений.

Синтаксис команды:

Menu (Номер первого пункта контекстного меню),(Номер второго пункта контекстного меню),...

Типы параметров: Menu *Int,Int, ...*

Пример:

Menu 1,3,5

Mouse 5/5,50/50,0 //Нет действий

Mouse 5/5,50/50,1,test1 //Переход по имени файла

Mouse 5/5,50/50,2,1_3_150,1 //Телеуправление

Mouse 5/5,50/50,3,1_2_100 //Телесигнал

Mouse 5/5,50/50,4,3 //Макрокоманда

Для вывода таблицы текущих значений по пункту меню 22 из экранной формы необходимо:

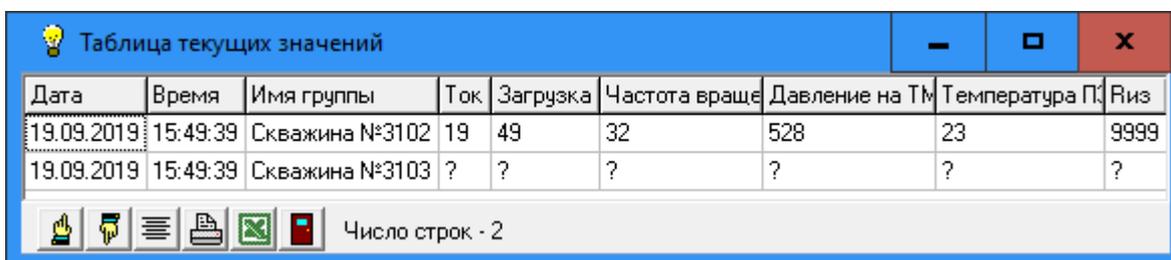
1) Заполнить таблицы настройки вывода текущих значений: CurrentGroups, CurrentPar. См. документ "База данных настроек АРМ" (файл ARMdb.doc).

2) Написать код в программе экранной формы на языке “АРМ-графика”.

Пример фрагмента программы экранной формы для вывода таблицы текущих значений:

```
...
int i1,i2 //Номера групп фактические
CallCell 30/40,CurTab,"Текущие значения"
...
Library
{
  {Cell CurTab
    Int i1,i2 //Номера групп формальные
    Vid 3d
    Pen 3
    Brush clLtGray
    Color clLtGray
    Button 0/0,70/20,clBlack,"Текущие значения"
    Menu 22
    Mouse 0/0,70/20,0
  }
}
```

Суть фрагмента состоит в передаче в подпрограмму “CurTab” номеров групп, прописанных в таблице “CurrentGroups”, через фактические и формальные параметры типа “Int”. Команда “Mouse” с нулевым действием реагирует на нажатие ПКМ и выводит меню, состоящее из пункта: “Таблица текущих значений экранной формы”, что обеспечивает команда “Menu 22”. Нажатие ЛКМ на этот пункт меню приведет к выводу таблицы текущих значений сигналов указанных групп. Пример выводимой таблицы:



Дата	Время	Имя группы	Ток	Загрузка	Частота враще	Давление на ТМ	Температура П	Физ
19.09.2019	15:49:39	Скважина №3102	19	49	32	528	23	9999
19.09.2019	15:49:39	Скважина №3103	?	?	?	?	?	?

15.38. Button – определяет визуальный образ стандартной кнопки для одной из предшествующих команд Mouse.

Синтаксис команды:

Button (x1)/(y1),(x2)/(y2) ,[цвет текста],<текст>

Типы параметров: Button *Point,Point,Clr,String*

Где: (x1)/(y1),(x2)/(y2) - координаты углов области кнопки;

текст – имя кнопки, отображаемое по ее центру. Текст может состоять из множества строк, разделяемых символами “\n” (без кавычек). Число строк в тексте – не более 10-и. Координаты углов области кнопки должны совпадать с координатами углов области клика соответствующей команды Mouse. Цвет кнопки и ее обрамления задают предшествующие команды Brush и Color, а ширину обрамления и вид кнопки - команды Pen и Vid.

Параметры фонта задают предшествующие команды Font, FontStyle. Угол поворота текста предварительно задается командой Angle. Допускаются углы поворота: 0 или 90.

Расположением текста управляет команда TrimButton. По умолчанию текст размещается по центру кнопки.

Пример:

Mouse 10/10,110/50,1,test1

Brush clLtGray

Color clLtGray

Button 10/10,110/50,clBlack,"Перейти к 'Паклиновская'"

15.39. PassTC – задает правило срабатывания команды Mouse, если в ней задано одно из следующих действий:

1 – Перейти к экранной форме (см. Примечание)

2 – Телеуправление

3 - Изменение значения TC

4 - Изменение значения TI

5 – Макрокоманда

7 - Изменение значения Const

8 - Управление через SendTC

11 – Блокировка телесигнала

12 – Блокировка телеизмерения

При наличии других действий команда PassTC не оказывает влияния на выполнение команды Mouse.

Синтаксис команды:

PassTC [имя телесигнала],[значение телесигнала)

Типы параметров: PassTC *SignTC,Int*

Правило записи имени телесигнала имеет вид:

(N абонента)_(N группы)_(N параметра) из таблицы TC

Значение – число: 0 или 1.

Команда PassTC должна предшествовать команде Mouse. Команда PassTC без параметров отменяет ранее определенные параметры. Команда Mouse будет исполняться после нажатия левой кнопки мыши в ее области, если текущее значение заданного в PassTC телесигнала совпадает с указанным в поле “значение телесигнала”, иначе, исполнение не производится.

Примечание.

Ранее команда PassTC не влияла на выполнение команды Mouse для действия под номером 1. Для сохранения работоспособности задела в конфигуратор АРМ (Arm.cfg) введена команда PassTC, принимающая значения: “Yes”, “No”. При задании “No” управление выводом экранных форм отменяется (работаем по-старому).

Пример:

PassTC 1_2_100,1 //Условие срабатывания последующего вызова телеуправления

Mouse 5/5,50/50,2,1_3_150,1 //Телеуправление

PassTC //Сброс

Mouse 5/5,50/50,4,3 //Макрокоманда – срабатывание без условий

В разделе Picture команда PassTC, однажды заданная с параметрами, будет воздействовать на все последующие команды Mouse из выше приведенного списка. Во избежание ошибок не забывайте сбрасывать команду PassTC после ее использования, записывая без параметров. При выполнении команды CallCell текущее состояние команды PassTC (из

Picture) будет передано в CallCell без изменений. Однако, изменения PassTC в вызванной CallCell не изменят состояния PassTC в разделе Picture после завершения CallCell.

15.40. Header – задает заголовок окна экранной формы, описание вызова которой определяется в последующей команде Mouse. Команда Mouse должна иметь номер действия – 1 (Перейти к картинке). Синтаксис команды:

Header <имя окна>

Типы параметров: Header *String*

Значение параметра команды Header гасится после команды Mouse.

Пример:

Header “Управление вентилятором”

15.41. OutInt – передача целых чисел в вызываемую программу с разделом Input. Должна располагаться перед командой Mouse и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutInt (1-е целое), (2-е целое),... - перечень целых чисел

Типы параметров: OutInt *Int,Int,...*

Пример:

OutInt 101,102

15.42. OutFloat – передача чисел с плавающей точкой вызываемую программу с разделом Input. Должна располагаться перед командой Mouse и гасится (теряют свое содержимое) после нее. Синтаксис команды:

OutFloat (1-е числj с плавающей точкой), (2-е числj с плавающей точкой),...

Типы параметров: OutFloat *Float,Float,...*

Пример:

OutFloat 12.34,56.78

15.43. OutClr - передача перечня цветов в вызываемую программу с разделом Input. Должна располагаться перед командой Mouse и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutClr [1-й цвет],[2-й цвет], ...

Типы параметров: OutClr *Clr,Clr,...*

Пример:

OutClr clGreen,clRed

15.44. OutPoint – передача координат точек в вызываемую программу с разделом Input. Должна располагаться перед командой Mouse и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutPoint (x1)/(y1),(x2)/(y2) ... - перечень координат точек.

Типы параметров: OutPoint *Point,Point,...*

Пример:

OutPoint 12/34,56/78

15.45. OutString – передача текстовых строк в вызываемую программу с разделом Input. Должна располагаться перед командой Mouse и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutString <1-й текст>,<2-й текст>, ...

Типы параметров: OutString *String,String,...*

Пример:

OutString “Строка 1”,”Строка 2”

15.46. OutIdent – передача идентификаторов в вызываемую программу с разделом Input. Должна располагаться перед командой Mouse и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutIdent [1-й идентификатор], [2-й идентификатор],...

OutIdent *Ident,Ident,...*

Пример:

OutIdent 2d,3d

15.47. OutSignTC – передача перечня телесигналов из таблицы TC в вызываемую программу с разделом Input. Должна располагаться перед командой Mouse и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutSignTC [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров: OutSignTC *SignTC,SignTC,...*

Структура имен сигналов имеет вид:

(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Перечисленные в команде сигналы должны быть описаны в базе данных настроек АРМ в таблице TC. Для состояния связи с объектом определены зарезервированные имена телесигналов Link с номером объекта. Сигналы Link не описывают в таблице TC. Можно использовать переменные типа SignTC из раздела Input.

Вместо имен сигналов допускается применять зарезервированное слово “null” (сигнал отсутствует) или указать значение сигнала: 0 или 1. Используются для резервирования позиции в OutSignTC для временно не заданного в таблице TC сигнала.

Пример:

OutSignTC 1_2_3,1_2_4

15.48. OutSignTI – передача перечня телеизмерений из таблицы TI вызываемую программу с разделом Input. Должна располагаться перед командой Mouse и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutSignTI [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров: OutSignTI *SignTI,SignTI,...*

Структура имен сигналов имеет вид:

(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Перечисленные в команде сигналы должны быть описаны в базе данных настроек АРМ в таблице TI. Можно использовать переменные типа SignTI из раздела Input.

Вместо имен сигналов допускается применять зарезервированное слово “null” (сигнал отсутствует) или указать значение сигнала в формате числа с плавающей точкой.

Используются для резервирования позиции в OutSignTI для временно не заданного в таблице TI сигнала.

Пример:

OutSignTI 1_2_3,1_2_4

15.49. OutSignTY – передача перечня телеуправлений из таблицы TY вызываемую программу с разделом Input. Должна располагаться перед командой Mouse и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutSignTY [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров: OutSignTY *SignTY,SignTY*,...

Структура имен сигналов имеет вид:

(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Перечисленные в команде сигналы должны быть описаны в базе данных настроек АРМ в таблице TY. Можно использовать переменные типа SignTY из раздела Input.

Вместо имен сигналов допускается применять зарезервированное слово “null” (сигнал отсутствует) или указать значение сигнала: 0 или 1. Используются для резервирования позиции в OutSignTY для временно не заданного в таблице TY сигнала.

Пример:

OutSignTY 1_2_3,1_2_4

15.50. OutSignCN – передача перечня констант из таблицы Const вызываемую программу с разделом Input. Должна располагаться перед командой Mouse и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutSignCN [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров: OutSignCN *SignCN,SignCN*,...

Структура имен сигналов имеет вид:

(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Перечисленные в команде сигналы должны быть описаны в базе данных настроек АРМ в таблице CN. Можно использовать переменные типа SignCN из раздела Input.

Вместо имен сигналов допускается применять зарезервированное слово “null” (сигнал отсутствует) или указать значение сигнала в формате числа с плавающей точкой.

Используются для резервирования позиции в OutSignCN для временно не заданного в таблице CN сигнала.

Пример:

OutSignCN 1_2_3,1_2_4

15.51. MouseTC – выводит на экран одну из двух перечисленных экранных форм при нажатии левой кнопки мыши в заданной области. Выбор экранной формы определяет состояние телесигнала, который является одним из параметров команды. При нулевом

значении телесигнала на экран выводится первая экранная форма, а при единичном – вторая.

Синтаксис команды MouseTC:

MouseTC (x1)/(y1),(x2)/(y2), [телесигнал],[имя первой экранной формы], [имя второй экранной формы]

Типы параметров: Mouse *Point,Point,SignTC,Ident,Ident*

Где: (x1)/(y1),(x2)/(y2) - координаты углов области клика.

В качестве имени телесигнала допускается использовать фактическое имя (например: 1_2_3), имя переменной или имя формального параметра типа SignTC в подпрограммах. Имя экранной формы является именем файла без расширения из таблицы Graphics. Допускается использовать имя переменной или имя формального параметра типа Ident в подпрограммах.

Предусмотрен механизм передачи параметров в вызываемую экранную форму. Комплект команд Header, OutInt, OutString, и т.д. определяет параметры для первой экранной формы, а комплект команд Header1, OutInt1, OutString1, и т.д. - для второй экранной формы.

Команда MouseTC реагирует и на правую кнопку мыши, если ей предшествует команда Menu.

Пример:

```
//Передаваемые параметры в KotelOn
Header “Включить котел”
OutTY 1_2_100
//Передаваемые параметры в KotelOff
Header1 “Отключить котел”
OutTY1 1_2_101
//Состояния ТС (1_2_3): 0 – котел отключен, 1 – котел включен
MouseTC 100/10,200/30,1_2_3,KotelOn,KotelOff
```

Пример использования команды MouseTC для организации окна с запросом пользователя на выполнение действия. Верхняя строка окна сообщает о текущем состоянии предписанного телесигнала (Включен, Отключен). Следующая строка запрашивает: хотите ли вы совершить переключение текущего состояния телесигнала. Примечание: переключения совершаются посылкой телеуправлений. Далее, следуют две кнопки с именами “Да”, “Нет”.

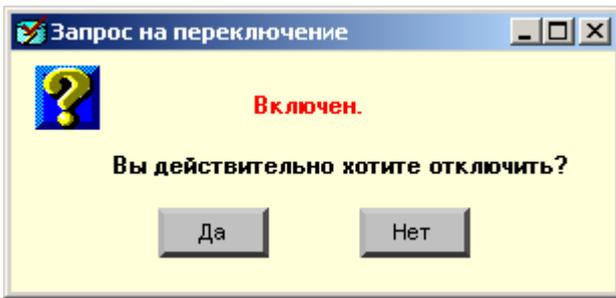
При нажатии кнопки “Да” по команде MouseTC вызывается одна из пары экранных форм в зависимости от состояния предписанного телесигнала:

- 1). Если телесигнал в состоянии “Включен” – вызывается экранная форма на отключение.
- 2). Если телесигнал в состоянии “Отключен” – вызывается экранная форма на включение.

Каждой вызываемой экранной форме передается свой комплект параметров. При вызове исходного окна в него передается сумма параметров для обеих вызываемых экранных форм.

При нажатии кнопки “Нет” выполняется закрытие экранной формы.

Окно имеет вид:



Вызов окна с запросом выполняется из основной программы посредством организации кнопки с именем “Переключить” и команды Mouse. Текст основной программы:

```
Base ...
Picture
{
...
OutIdent Vkl,Otkl //имена вызываемых экранных форм в программе YesNo
OutSignTC 4_2_1
OutSignTY 4_2_150,4_2_151 //TY “Включить”,“Отключить” для форм Vkl,Otkl
Mouse 140/320,240/340,1,YesNo
Button 140/320,240/340,clBlack,"Переключить"
...
}
```

Текст вызываемой программы с запросом:

```
Base -1,1,"Запрос на переключение" //-1 – вызов окна в доп. ЭФ без закрытия исходной
{Size 300/120,Simple}
Input
{
ident fvkl,fotkl //имена вызываемых экранных форм
signtc sw //TC состояния
signty ty_vkl,ty_otkl //TY “Включить”,“Отключить”
}
Picture
{
// Закраска фона
Vid 2D
Color d9ffff
Brush d9ffff
Pen 0
Rect 0/0,300/120
Image 12/7,44/39,"qmark.bmp" //картинка с вопросительным знаком
// Тексты: состояние, запрос
SignTC sw
CallCell 50/20,PrnTxt,"Тексты"
// Кнопки
Vid 3D
Pen 3
Brush clLtGray
Color clDkGray
OutSignTY ty_vkl
```

```

OutSignTY1 ty_otkl
MouseTC 75/80,125/100,sw,fvkl,fotkl //вызов экранной формы
Button 75/80,125/100,clBlack,"Да"
Mouse 175/80,225/100,9 //закрытие экранной формы
Button 175/80,225/100,clBlack,"Нет"
}
Library
{
Cell PrnTxt
{
SignTC tc
// 1. Строка состояния
Colors clGreen,clRed
FontStyle fsBold
Texts "Отключен.", "Включен."
TextTC tc,70/0
// 2. Строка запроса
Colors clBlack,clBlack
Texts "Вы действительно хотите включить?","Вы действительно хотите отключить?"
TextTC tc,0/30
}
}
}

```

В нашем примере вызванная экранная форма (vkl или otkl) должна иметь раздел Input с типом signty, кнопку и команду Mouse для запуска переданного телеуправления. Номер абонента в разделе Base должен быть отрицательным.

15.52. Header1 – задает заголовок окна второй экранной формы, описание вызова которой определяется в последующей команде MouseTC. Синтаксис команды:

Header1 <имя окна>

Типы параметров: Header1 *String*

Значение параметра команды Header1 гасится после команды MouseTC.

Пример:

Header1 “Управление вентилятором”

15.53. OutInt1 – передача целых чисел при вызове второй экранной формы командой MouseTC. Экранная форма должна иметь раздел Input с командой Int. Число параметров в OutInt1 и Int должно совпадать.

Команда OutInt1 должна располагаться перед командой MouseTC и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutInt1 (1-е целое), (2-е целое),... - перечень целых чисел

Типы параметров: OutInt1 *Int,Int,...*

Пример:

OutInt1 101,102

15.54. OutFloat1 – передача чисел с плавающей точкой при вызове второй экранной формы командой MouseTC. Экранная форма должна иметь раздел Input с командой Float. Число параметров в OutFloat1 и Float должно совпадать.

Команда OutFloat1 должна располагаться перед командой MouseTC и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutFloat1 (1-е числј с плавающей точкой), (2-е числј с плавающей точкой),...

Типы параметров: OutFloat1 *Float,Float,...*

Пример:

OutFloat1 12.34,56.78

15.55. OutClr1 - передача перечня цветов при вызове второй экранной формы командой MouseTC. Экранная форма должна иметь раздел Input с командой Clr. Число параметров в OutClr1 и Clr должно совпадать.

Команда OutClr1 должна располагаться перед командой MouseTC и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutClr1 [1-й цвет],[2-й цвет], ...

Типы параметров: OutClr1 *Clr,Clr*,...

Пример:

OutClr1 clGreen,clRed

15.56. OutPoint1 – передача координат точек при вызове второй экранной формы командой MouseTC. Экранная форма должна иметь раздел Input с командой Point. Число параметров в OutPoint1 и Point должно совпадать.

Команда OutPoint1 должна располагаться перед командой MouseTC и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutPoint1 (x1)/(y1),(x2)/(y2) ... - перечень координат точек.

Типы параметров: OutPoint1 *Point,Point*,...

Пример:

OutPoint1 12/34,56/78

15.57. OutString1 – передача текстовых строк при вызове второй экранной формы командой MouseTC. Экранная форма должна иметь раздел Input с командой String. Число параметров в OutString1 и String должно совпадать.

Команда OutString1 должна располагаться перед командой MouseTC и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutString1 <1-й текст>,<2-й текст>, ...

Типы параметров: OutString1 *String,String*,...

Пример:

OutString1 “Строка 1”,”Строка 2”

15.58. OutIdent1 – передача идентификаторов при вызове второй экранной формы командой MouseTC. Экранная форма должна иметь раздел Input с командой Ident. Число параметров в OutIdent1 и Ident должно совпадать.

Команда OutIdent1 должна располагаться перед командой MouseTC и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutIdent1 [1-й идентификатор], [2-й идентификатор],...

OutIdent1 *Ident,Ident*,...

Пример:

OutIdent1 2d,3d

15.59. OutSignTC1 – передача перечня телесигналов из таблицы ТС при вызове второй экранной формы командой MouseTC. Экранная форма должна иметь раздел Input с командой SignTC. Число параметров в OutSignTC1 и SignTC должно совпадать.

Команда OutSignTC1 должна располагаться перед командой MouseTC и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutSignTC1 [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров: OutSignTC1 *SignTC,SignTC*,...

Структура имен сигналов имеет вид:

(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Пример:

OutSignTC1 1_2_3,1_2_4

15.60. OutSignTI1 – передача перечня телеизмерений из таблицы TI при вызове второй экранной формы командой MouseTC. Экранная форма должна иметь раздел Input с командой SignTI. Число параметров в OutSignTI1 и SignTI должно совпадать.

Команда OutSignTI1 должна располагаться перед командой MouseTC и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutSignTI1 [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров: OutSignTI1 *SignTI,SignTI*,...

Структура имен сигналов имеет вид:

(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Пример:

OutSignTI1 1_2_3,1_2_4

15.61. OutSignTY1 – передача перечня телеуправлений из таблицы TY при вызове второй экранной формы командой MouseTC. Экранная форма должна иметь раздел Input с командой SignTY. Число параметров в OutSignTY1 и SignTY должно совпадать.

Команда OutSignTY1 должна располагаться перед командой MouseTC и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutSignTY1 [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров: OutSignTY1 *SignTY,SignTY*,...

Структура имен сигналов имеет вид:

(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Пример:

OutSignTY1 1_2_3,1_2_4

15.62. OutSignCN1 – передача перечня констант из таблицы Const при вызове второй экранной формы командой MouseTC. Экранная форма должна иметь раздел Input с командой SignCN. Число параметров в OutSignCN1 и SignCN должно совпадать.

Команда OutSignCN1 должна располагаться перед командой MouseTC и гасится (теряет свое содержимое) после нее. Синтаксис команды:

OutSignCN1 [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров: OutSignCN1 *SignCN,SignCN*,...

Структура имен сигналов имеет вид:
(NumAbon)_(NomGr)_(NomPar).

Где:

NumAbon – номер абонента;

NomGr – номер группы;

NomPar – номер параметра (сигнала).

Пример:

OutSignCN1 1_2_3,1_2_4

Пример описания раздела Picture:

Picture

{ColorAvar cWhite //Цвет блинкера

Menu 1,3,5 //Перечень пунктов контекстного меню

Mouse 5/5,50/50,1,test1 //Для левой кнопки задан переход к картинке test1

Brush cLtGray

Color cLtGray

Button 5/5,50/50,cBlack,"Вызов объекта test1"

Pen 5 // Ширина всех последующих примитивов черчения до новой команды Pen

Color cBlack // Цвет всех последующих примитивов черчения до новой команды Color

Brush cBlue

Points 20/40,100/40

Polyline // прорисовка полигона (линии) для двух точек

Rect 30/30,50/50 // Коорд.углов. Ширина и цвет контура в соответствии с Pen и Color

Brush cRed

Rect 120/20,140/40 // Закрашенный Rect. Координаты углов.

Ellipse 50/50,20,20 // координаты центра и размеры полуосей

Font 8,Sans Serif //фонт для последующего применения в командах text

Text 10/20,"Котел N 1" // цвет фонта в соответствии в командой Color

Image 10/20,100/200,"MyImage.bmp" // Коорд.углов области рисования. Имя файла.

SignTC 1_-1_62,1_1_33 // перечень сигналов таблицы TC (фактические сигналы)

SignTI 1_-1_100,1_2_77 // перечень сигналов таблицы TI (фактические сигналы)

CallCell 10/20,FD5,"ФидерA5"

...

}

15.63. Opt - Настройка команд LevelTI, MeterTI и ArrowTI.

Синтаксис команды:

Opt [Состав текста],[Видимость шкалы],[Ориентация]

Типы параметров: Opt *Ident*, *Ident*, *Ident*

Команда Opt может иметь от одного до трех параметров. Порядок их следования определен синтаксисом команды. Перед командами ArrowTI и MeterTI команда Opt не может иметь более двух параметров, т.к. параметр "Ориентация" используется только в команде LevelTI.

Параметр "Состав текста" может принимать один из следующих идентификаторов:

oNo – текст отсутствует.

oPercent – текстом является процентное отношение значения текущего сигнала к максимально допустимому его значению в заданном диапазоне. Диапазон возможных значений сигнала задается в полях MinVal, MaxVal таблицы TI БД настроек АРМ.

oValue - текстом является значение текущего сигнала.
oValueUnit - текстом является значение текущего сигнала с его единицей измерения.
oMiddlePercent – текстом является процентное отношение значения текущего сигнала к максимально допустимому его значению в заданном диапазоне. Диапазон возможных значений сигнала задается в полях MinVal, MaxVal таблицы TI БД настроек АРМ. Текст располагается по высоте в середине сегмента. Использует команда ArrowTI.

oMiddleValue - текстом является значение текущего сигнала. Текст располагается по высоте в середине сегмента. Использует команда ArrowTI.

oMiddleValueUnit - текстом является значение текущего сигнала с его единицей измерения. Текст располагается по высоте в середине сегмента. Использует команда ArrowTI.

Параметры шрифта текста определяют команды Font и FontStyle.

Параметр “Видимость шкалы” допускает следующие значения:

oScaleOff – шкала отсутствует; в команде MeterTI ход стрелки – 90 градусов.

oScaleOn – выполняется прорисовка шкалы; в команде MeterTI ход стрелки – 90 градусов.

oRoundScaleOff – шкала выключена; в команде MeterTI ход стрелки – 270 градусов.

oRoundScaleOn – шкала включена; в команде MeterTI ход стрелки – 270 градусов.

Параметр “Ориентация” используется командой LevelTI и может иметь одно из следующих значений:

oAny – уровень ориентируется вдоль длинной стороны прямоугольника, заданного командой LevelTI.

oUp – уровень ориентируется снизу вверх

oRight – уровень ориентируется слева направо

При отсутствии команды Opt считается, что задано

Opt oPercent,oScaleOff,oAny.

Команда Opt, также, позволяет управлять видом и толщиной стрелки посредством следующих идентификаторов:

oArrowX, где X является числом и задает толщину стрелки. Например: oArrow3.

oOutArrowX, где X является числом и задает толщину стрелки. При отсутствии данной команды стрелка рисуется от центра вращения. При наличии данной команды рисуется только верхняя часть стрелки, длина которой составляет 20 процентов от полной длины стрелки.

Пример:

Opt oValueUnit

15.64. Lines.

Определяет правила рисования полилинии в командах: Polyline, PolylineTC, SPolylineTC, PolylineTI и располагается перед ними. Полилинией именуется кусочно-линейная кривая, координаты которой задаются в команде Points.

Команда Lines воздействует на прорисовку полилинии, если будут выполнены следующие условия:

1) Все отрезки полилинии параллельны осям координат.

2) Команда Vid имеет параметр 3D или Brd.

3) Файл конфигурации содержит команду volume=yes.

При невыполнении этих условий полилиния рисуется в режиме 2D без учета команды Lines.

Синтаксис команды:

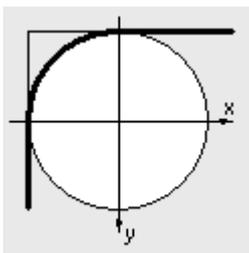
Lines (x1)/(y1),(Врезка начало),(Врезка конец)

Типы параметров: Lines *Point,Int,Int*

Где:

1) x1, y1 – размеры осей эллипса. Задают сглаживание углов полилинии. Кривая скругленных углов совпадает с кривой эллипса в точках смены направления полилинии (кусочно-линейной кривой). Центр эллипса располагается внутри меньшего угла, образованного смежными отрезками полилинии.

Пример: Lines 10/10,0,0



На рисунке приведен пример сглаживания на основе заданного эллипса.

Если оси эллипса меньше толщины полилинии, то внешние стороны сглаживаются, а внутренние сопрягаются под прямым углом.

Пример.

Color clBlue

Pen 20

Vid 3D

Lines 20/20,0,0

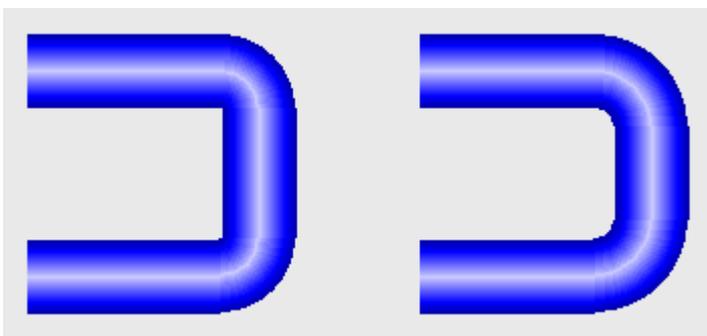
Points 417/90,477/90,477/144,417/144

Polyline

Lines 30/30,0,0

Points 515/90,575/90,575/144,515/144

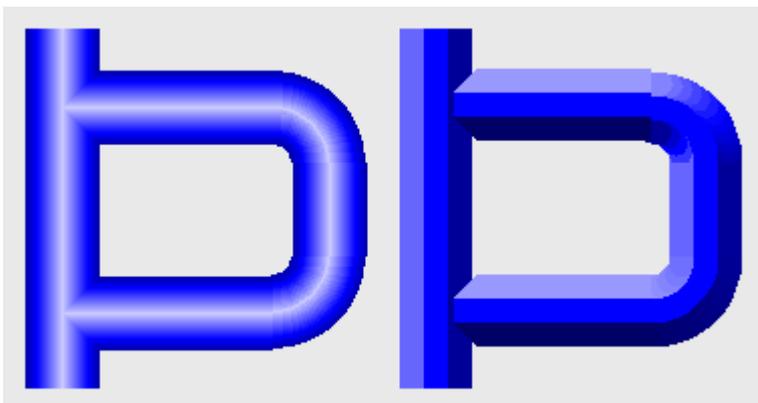
Polyline



2) Врезка начало и врезка конец. При нулевых значениях параметров врезок полилинии имеют ровные окончания (см. предыдущую картинку). Ненулевое значение параметра задает глубину проникновения начала (конца) полилинии в другую соприкасающуюся полилинию. Проникновение рисуется в виде треугольника. Если ширины соприкасающихся полилиний одинаковы, то значение врезки следует задать равным половине ширины.

Пример.

```
Color cBlue
Pen 20
Vid 3D
Lines 30/30,0,0
Points 407/70,407/164
Polyline
Lines 30/30,10,10
Points 417/90,477/90,477/144,417/144
Polyline
Vid Brd
Lines 30/30,0,0
Points 505/70,505/164
Polyline
Lines 30/30,10,10
Points 515/90,575/90,575/144,515/144
Polyline
```



15.65. TrimButton – управляет расположением текста в командах: Button, ButtonTC, ButtonTI, ButtonTCTI, SButtonTC, RoundButtonTC, SRoundButtonTC, RoundButtonTI, RoundButtonTCTI. Определяет подгонку (выравнивание) строк однострочных и многострочных текстов при их выводе на поверхности кнопки. Напомним, что строки в тексте разделяются парой символов: “\n” (без кавычек).

Синтаксис команды: TrimButton (отступ)[значение]

Типы параметров: TrimButton [Int][Ident]

Отступ – необязательный параметр. Используется со следующими правилами подгонки текста: trR, trL (см. ниже). Задаёт для выводимых текстов отступы от краёв кнопки. Отступ задаётся в проектных единицах экранной формы. По умолчанию (при его отсутствии) отступ равен нулю. Для trR отступ отсчитывается от правого края кнопки. Для trL отступ отсчитывается от левого края кнопки.

Принимает значения:

trR – строки текста подгоняются по правому краю кнопки;

trC – строки текста подгоняются по центру кнопки;

trL – строки текста подгоняются по левому краю кнопки.

По умолчанию имеет значение trC.

Пример:
TrimButton 4trR //отступ на 4 проектных единицы от правого (верхнего) края
Angle 90
Text “Выравнивание двух строк\ппо правому краю кнопки” //т.е. верхнему с учетом поворота

16. Раздел **Library**.

Раздел начинается с ключевого слова **Library** без параметров. Раздел состоит из набора подразделов **Cell**. Подраздел **Cell** начинается с ключевого слова **Cell**. Все входящие в него команды обрамляются открывающей и закрывающей скобками. Первыми командами подраздела **Cell** являются описания формальных параметров ячейки.

Формальные параметры ячейки задаются командами **Int**, **Float**, **Clr**, **Point**, **String**, **Ident**, **SignTC**, **SignTI**, **SignTY**, **SignCN**. Состав и перечень команд определяется индивидуально для каждой ячейки и зависит от решаемой задачи. Фактические параметры описываются в разделе **Picture** тем же набором команд перед командой обращения к ячейке **CallCell**. Количество формальных параметров ячейки и фактических из раздела **Picture** должны совпадать для каждой использованной команды.

16.1. **Int** - перечень формальных имен для целочисленных значений.

Синтаксис команды **Int**:
Int [1-е число],[2-е число] ... - перечень имен чисел
Типы параметров: **Int** *Int,Int*, ...

Имена чисел – произвольные идентификаторы, смысловую нагрузку которых составитель библиотеки определяет по собственному усмотрению.
Пример: **Int** nul,one

16.2. **Float** - перечень формальных имен для чисел с плавающей точкой.

Синтаксис команды:
Float [1-е число],[2-е число] ... - перечень имен чисел
Типы параметров: **Float** *Float,Float*, ...

Пример: **Float** Pmax,Pmin

16.3. **Clr** - перечень формальных имен цветов.

Синтаксис команды:
Clr [1-й цвет],[2-й цвет] ... - перечень имен цветов
Типы параметров: **Clr** *Clr,Clr*, ...

Пример: **Clr** Avar

16.4. **Point** - перечень формальных имен для координат точек.

Синтаксис команды:
Point [1-я точка],[2-я точка] ... - перечень имен точек

Типы параметров: Point *Point,Point*,...

Пример: Point pt1,pt2

16.5. String - перечень формальных имен для текстовых строк.

Синтаксис команды:

String [1-й текст],[2-й текст] ... - перечень имен строк

Типы параметров: String *String,String*,...

Пример: String name

16.6. Ident- перечень формальных имен идентификаторов.

Синтаксис команды:

Ident [1-й идент.],[2-й идент.] ... - перечень идентификаторов

Типы параметров: Ident *Ident,Ident*,...

Пример: Ident pictureName

16.7. SignTC - перечень формальных имен для телесигналов.

Синтаксис команды SignTC:

SignTC [1-й сигнал],[2-й сигнал] ... - перечень имен телесигналов

Типы параметров: SignTC *SignTC,SignTC*,...

Имена телесигналов – произвольные идентификаторы, смысловую нагрузку которых составитель библиотеки определяет по собственному усмотрению.

Пример: SignTC sw,in,out

16.8. SignTI - перечень формальных имен для сигналов телеизмерений.

Синтаксис и структура команды SignTI:

SignTI [1-й сигнал],[2-й сигнал] ... - перечень имен телеизмерений

Типы параметров: SignTI *SignTI,SignTI*,...

Пример: SignTI level,I,U

16.9. SignTY - перечень формальных имен для телеуправлений.

Синтаксис команды SignTY:

SignTY [1-й сигнал],[2-й сигнал] ... - перечень имен телеуправлений

Типы параметров: SignTY *SignTY,SignTY*,...

Имена телеуправлений – произвольные идентификаторы, смысловую нагрузку которых составитель библиотеки определяет по собственному усмотрению.

Пример: SignTY on,off

16.10. SignCN - перечень формальных имен для констант.

Синтаксис и структура команды SignCN:

SignCN [1-й сигнал],[2-й сигнал] ... - перечень имен констант

Типы параметров: *SignCN SignCN,SignCN,...*

Пример: *SignCN top,low*

Для оформления картинки ячейки используется набор элементарных графических команд, перечисленных в разделе *Picture*. В дополнение к набору элементарных команд используются специальные команды.

16.11. Attr - задает текстовые данные, описывающие неизменные параметры библиотечной ячейки (название агрегата, мощность, конструктивный тип ...). Данные из *Attr* используются во всплывающей подсказке и при опросе свойств ячейки, в формах управления ячейкой и т.д. Для одной ячейки допускается не более девяти команд *Attr*.

Синтаксис команды: *Attr* <произвольный текст>

Типы параметров: *Attr String*

Пример: *Attr “Фидер ФД5”*.

16.12. FreshTC, FreshTI

Программа вывода и мониторинга экранной формы (ЭФ) работает в режиме реального времени, т.е. при поступлении сигнала изменяются соответствующие сигналу элементы ЭФ в соответствии с описанием программы на языке АРМ-графика. При составлении сложных рисунков из специальных команд иногда возникает необходимость совмещения их изображений на картинке. Учитывая событийный характер поведения программы, в этом случае происходит затирание ранее отображенных элементов ЭФ. Восстановить затертые элементы можно путем имитации поступления сигналов в программу вывода для их восстановления. Описание имитируемых сигналов производится командами *FreshTC* и *FreshTI*. Имитированные сигналы не появляются в остальной части программного комплекса АРМ.

В кратком изложении правило расстановки *FreshTC* и *FreshTI* звучит так: *FreshTC* и *FreshTI* должны располагаться перед специальными командами рисования, которые являются виновниками затирания изображений других специальных команд. В качестве параметров команд *FreshTC* и *FreshTI* задаются имена сигналов от затираемых команд.

Команды *FreshTC* и *FreshTI* располагаются в подпрограммах. Они содержат перечень фактических имен телесигналов или телеизмерений и указывают последующим в текущей подпрограмме специальным командам о необходимости имитации перечисленных сигналов. В командах *FreshTC* и *FreshTI* допускается использовать переменные, которым необходимо присвоить значения до вызывающей *CallCell*. Типами переменных являются *SignTC SignTI*. В командах *FreshTC* и *FreshTI* не допускается использовать формальные имена сигналов. Действие команд *FreshTC* и *FreshTI* можно прервать, не дожидаясь конца подпрограммы, если ввести в текст подпрограммы строку *FreshTC* или *FreshTI* без параметров.

Имитация состоит из двух фаз.

1) На вход поступил сигнал. Производится перерисовка тех частей ЭФ, которые ранее были нарисованы специальными командами, связанными с поступившим сигналом. Если перед рисуемой командой имеется команда *FreshTC* или *FreshTI*, то приписанные к ним сигналы сохраняются в массиве имитируемых сигналов без повторений.

2) При наличии имитируемых сигналов, они поочередно подаются на вход программы вывода (рисования) ЭФ. Программа исполнит (перерисует) все специальные команды, которым предписаны сигналы из числа имитируемых сигналов.

Для предписания сигнала специальной команде используется команда `Signals` (команды `TableTC`, `SPolylineTC` и т.д.). В некоторых командах предписанный сигнал является параметром команды (`LevelTI` и т.д.).

Замечание.

Восстанавливаемая часть экранной формы через механизм имитации сигналов должна находиться в другой подпрограмме, а не в той, где расположены иницилирующие ее команды `FreshTC` и `FreshTI`.

Синтаксис команд:

`FreshTC` [1-й сигнал],[2-й сигнал] ... - перечень сигналов

`FreshTI` [1-й сигнал],[2-й сигнал] ... - перечень сигналов

Типы параметров:

`FreshTC` *SignTC,SignTC,...*

`FreshTI` *SignTI,SignTI,...*

Структура имен сигналов совпадает с описанием фактических параметров для команд `SignTC`, `SignTI`.

Алгоритм восстановления затертых элементов зависит от наличия или отсутствия команд `SignalBlinks` и `StateBlinks` в подпрограмме рядом с командами `FreshTC` и `FreshTI` до специальной команды, которая может затереть рисунок другой команды.

16.12.1. Команды `SignalBlinks` и `StateBlinks` отсутствуют.

Имитация выполняется один раз сразу после перерисовки поступившей команды.

Пример:

`FreshTC` 1_1_22,1_1_23

`RectTC` clRed,clGreen,sw,10/20,10/40

`FreshTC`

Еще пример размещения рисунка ячейки `OverMoto` (закрашенный треугольник) всегда над рисунком ячейки `Moto` (квадрат с кругом):

`Picture`

{

...

`SignTC` 1_3_70,1_3_71

`CallCell` 15/150,Moto,"Группа Двигатель"

`SignTC` 1_1_70,1_1_71,1_1_72

`CallCell` 15/150,OverMoto,"Группа "

...

}

`Library`

{

`Cell` Moto

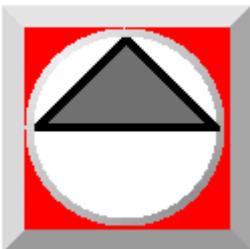
{

```

SignTC s1,s2
Color clLtGray
ColorAvar clLtGray
ColorBlink clBlack
ColorDisable clLtGray
Colors clRed,clWhite,clDkGray
Pen 3
Signals s1,s2
States 1,1
SRectTC rgLast,0/0,30/30
Colors clWhite,clWhite,clWhite,clWhite
FreshTC 1_1_70,1_1_71,1_1_72
Pen 1
SEllipseTC rgLast,2/2,28/28
}
Cell OverMoto
{
SignTC s1,s2,s3
Color clBlack
ColorAvar clLtGray
ColorBlink clBlack
ColorDisable clLtGray
Colors clGreen, clRed,clYellow,clDkGray
Pen 1
Points 3/15,15/3,27/15
Signals s1,s2,s3
States 1,1,1
SPolygonTC rgLast
}
...
}

```

На экранной форме пример имеет вид:



16.12.2. Команды SignalBlinks и StateBlinks присутствуют.

Их присутствие означает мигание последующих за ними в подпрограмме специальных команд рисования при наступлении предписанных состояний телесигналов. Механизм мигания применяют для привлечения внимания к произошедшим событиям. Рассмотрим пример. Пусть при изменении состояния выключателя на ЭФ вокруг него нужно запрограммировать мигающую рамку. Рамка рисуется каждую секунду, чередуя то цвет фона (ColorBlink), чтобы скрыть рамку, то выбранный цвет из Colors в соответствии с алгоритмом работы примененной в примере команды SPolylineTC. Заметим, что рамка затирает подходы к выключателю. Добавим команду FreshTC для восстановления под рамкой затертых частей рисунков и зададим ей в качестве параметров телесигналы,

связанные с командами рисования подходов. Каждый раз при перерисовке мигающего прямоугольника цветом фона автоматически запускается механизм имитации с целью восстановления затертых рамкой частей рисунка.

Формальное описание. На вход программы вывода и мониторинга ЭФ поступил сигнал.

1) Накапливаем перечень сигналов для имитации (см. выше). Запасаем команды рисования, связанные с поступившим сигналом.

2) Имитируем накопленные сигналы для восстановления затертых частей рисунка (см. выше).

3) Запускаем секундный таймер. При четном числе секунд исполняем запасенные команды рисования с основными цветами.

При нечетном числе секунд исполняем запасенные команды рисования с цветами фона (для затирания), а также имитируем накопленные сигналы для восстановления затертых частей рисунка.

Пример мигания:

```
Base 3,1,"7ср РУ-6кВ КНС-1" //N абонента, N листа, Имя листа
```

```
{Size 629/429,Origin
```

```
Ground e9e9e9
```

```
}
```

```
Var
```

```
{SignTC frh1,frh2
```

```
}
```

```
Picture
```

```
{
```

```
Vid 2D
```

```
//Выключатель
```

```
frh1=3_1_15 //низ
```

```
frh2=3_1_1 //верх
```

```
SignTC 3_1_2
```

```
CallCell 40/60,kvadrat_1,""
```

```
//Подходы к выключателю. Д.б. после
```

```
SignTC 3_1_1,3_1_1
```

```
CallCell 56/40,fiderUp,""
```

```
SignTC 3_1_15
```

```
CallCell 56/92,sekcijaDn,""
```

```
}
```

```
Library
```

```
{
```

```
Cell fiderUp
```

```
{
```

```
SignTC c1,c2
```

```
Signals c1,c2
```

```
States 1,1
```

```
Colors clBlack,clLtGray,clDkGray
```

```
Pen 2
```

```
Points 0/0,0/20
```

```
SPolylineTC rgFirst
```

```
}
```

```
Cell kvadrat_1
```

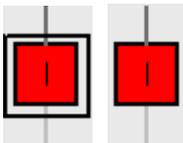
```
{
```

```

SignTC on
ColorDisable clLtGray
Colors clGreen,clRed
Colors clBlack,clBlack,clBlack
Font 14,MS Sans Serif
Texts "—", "I", "?"
Pen 2
Color clBlack
ButtonTC on,0/0,32/32
ColorDisable clDkGray
Signals on,on
States 0,1
ColorBlink clBlack
SignalBlinks on,on
StateBlinks 0,1
Colors e9e9e9,e9e9e9
ColorAvar clBlack
FreshTC frh1,frh2
Points -5/-5,37/-5,37/37,-5/37,-5/-5
SPolylineTC rgFirst
// Взамен предыдущих трех команд
// Можно обойтись без FreshTC, рисуя две мигающие половинки, не пересекая подходы
// Points 14/-5,-5/-5,-5/37,14/37
// SPolylineTC rgFirst
// Points 18/-5,37/-5,37/37,18/37
// SPolylineTC rgFirst
}
Cell sekciyaDn
{
SignTC c1
Color clBlack
ColorDisable clDkGray
Pen 2
Points 0/0,0/20
PolylineTC clLtGray,clBlack,c1
}
}

```

Вид примера на ЭФ. Происходит чередование картинок через каждую секунду.



16.13. TableTI – рисует таблицу текущих значений телеизмерений с именами столбцов и строк. Столбцы и строки образуют набор клеток таблицы. Каждой клетке соответствует свой сигнал, значение которого выводится в данной клетке. Слева от рядов клеток таблицы располагаются имена строк. Над столбцами клеток располагаются имена столбцов. Набор сигналов телеизмерений выбирается из перечня формальных сигналов ячейки (команда SignTI) и включается в перечень параметров команды **Signals**. Команда Signals должна предшествовать команде TableTI.

Синтаксис команды:

TableTI (число рядов),(число столбцов) ,[цвет текста],[x1]/(y1),(x2)/(y2)

Типы параметров: TableTI *Int,Int,Clr,Point,Point*

Где: (x1)/(y1),(x2)/(y2) - координаты прямоугольника для размещения таблицы.

Команде TableTI должны предшествовать команды Signals и Texts. Предварительно заданная команда Vid определяет вид контура и клеток. Команда Format задает правила вывода значений сигналов в клетках таблицы.

Синтаксис команды **Signals**:

Signals [сигнал 1],[сигнал 2], ...

Типы параметров: Signals *Ident,Ident,...*

Число сигналов= число рядов*число столбцов.

Перечисляются сигналы первого ряда клеток таблицы, затем второго и т.д.

Напоминаем, что для записи большого числа параметров следует использовать признак переноса (символ +), после которого продолжайте ввод с новой строки.

В процессе работы клетки таблицы остаются пустыми для недостоверных сигналов.

Синтаксис команды **Texts**:

Texts <текст для первого ряда>,< текст для второго ряда >, ...

,<текст первой колонки>,<текст второй колонки>,...

Типы параметров: Texts *String,String,...*

Число текстов= число рядов+число столбцов

Если число текстов равно числу рядов, то колонки не имеют текстовых пояснений (ряд под тексты столбцов отсутствует).

Если в команде Texts все имена строк пустые тексты вида “”, то столбец с именами строк не выводится.

Если в команде Texts все имена строк пустые тексты вида “”, а число текстов равно числу рядов, то имеем таблицу, состоящую только из клеток со значениями.

При наличии столбца с именами строк его ширина определяется по ширине самой длинной строки. Ширины остальных столбцов одинаковы и вычисляются по ширине таблицы за вычетом ширины столбца с именами строк.

Цвет фона таблицы задается предшествующей командой Brush. Ширина и цвет контура и разграничительных линий предварительно определяются командами Pen, Color. Команда Format определяет формат выводимых значений и подгонку имен строк и столбцов (влево, по центру, вправо). Вид таблицы определяет команда Vid. При наличии команды Vid Brd рекомендуется задать Pen не менее 3.

Если необходимо иметь в таблице незаполненные клетки, то при вызове CallCell задайте среди фактических параметров в команде SignTI слова “null” для соответствующих сигналов.

Команда Fon позволяет менять цвет клетки в зависимости от состояния сигнала. Может отсутствовать, если подкраска цветом состояния сигналов не требуется. Команда Fon должна предшествовать команде TableTI. В команде Fon каждому интервалу задается цвет отображения состояния сигнала. После поступления в систему очередного телеизмерения производится вычисление номера соответствующего ему интервала по таблице 2 (См. П.16.15). Номер интервала определяет номер параметра в команде Fon,

который и задает цвет клетки таблицы. Число интервалов равно семи. Если поступившее телеизмерение недостоверно - цвет ячейки определяет предшествующая команда ColorDisable. Правила применения команды Fon см. ниже при описании команды Columns.

Синтаксис команды Fon:

Fon [цвет первого интервала],[цвет второго интервала], ..., [цвет седьмого интервала]

Типы параметров: Fon *Clr,Clr*,...

Команда Colors позволяет менять цвет текстовой строки клетки в зависимости от состояния сигнала. Может отсутствовать, если мониторинг состояния сигналов не требуется. Команда Colors должна предшествовать команде TableTI. В команде Colors каждому интервалу задается свой цвет текстовой строки клетки. После поступления в систему очередного телеизмерения производится вычисление номера соответствующего ему интервала. Правила применения команды Colors см. ниже при описании команды Columns.

Синтаксис команды Colors:

Colors [цвет первого уровня],[цвет второго уровня], ..., [цвет седьмого уровня]

Типы параметров: Colors *Clr,Clr*,...

Команда Columns настраивает столбцы таблиц на вывод имени, времени измерения, единиц измерения или значения сигнала. При ее отсутствии выводятся значения сигналов. Позволяет задать ширины столбцов и определить правила раскраски текстовых строк и фона ячеек.

Синтаксис команды Columns:

Columns [знак][тип информации в столбце 1][ширина столбца 1], [знак][тип информации в столбце 2][ширина столбца 2]...

Типы параметров: Columns [*Ident*]*Ident*[*Int*] [*Ident*],*Ident*[*Int*],...

Знак – необязательный параметр (тип - Ident). Может являться с одним из следующих символов:

- 1) Знак отсутствует или команда Columns отсутствует. Текст во всех клетках таблицы имеет цвет, заданный третьим параметром команды TableTI. Цвет фона клетки выбирается из предшествующей команды Fon в зависимости от номера интервала, приписанного клетке телеизмерения.
- 2) '-' (минус). Означает отказ от раскраски строк и фона клеток столбца по состоянию телеизмерений. Их цвета остаются исходными.
- 3) '+' (плюс). Цвет текста определяет предшествующая команда Colors по номеру интервала. Интервал вычисляется по состоянию телеизмерения. Цвет клетки не меняет свою фоновую окраску, т.е. имеет цвет, заданный командой Brush.
- 4) '*' (звезда). Цвет текста и фона клетки определяют предшествующие команды Colors и Fon по номеру интервала. Интервал вычисляется по состоянию флагов телеизмерения, приписанного данной клетке.

Тип информации в столбце – обязательный параметр (тип Ident). Является одним из следующих идентификаторов:

- D или Date - выводит даты измерения сигналов
- T или Time - выводит времена измерения сигналов
- DT или DateTime - выводит даты и времена измерения сигналов
- V или Value – выводит значения сигналов
- N или Name - выводит имена сигналов, заданные в таблице сигналов БД настроек АРМ (имена таблиц: ТС, ТI или Const).

- U или Unit - выводит единицы измерения значений сигналов, заданные в таблице телеизмерений БД настроек АРМ.

Ширина столбца - необязательный член параметра (тип - Int). Является целым числом. При его отсутствии производится автоматический расчет ширины столбца на основании выводимой информации.

Команда Columns без параметров отключает действие предыдущей команды Columns.

Пример, в котором первая клетка таблицы остается незаполненной, а аварийные и предаварийные сигналы окрашивают клетки таблицы в красный и желтый цвета соответственно:

```
Picture
{
...
SignTI null,1_2_3
CallCell 100/200,Tab,"Таблица"
...
}
Library
{
...
Cell Tab
{SignTI U,I
Signals U,I
Texts "Ввод 1 ", "U", "I"
Brush clLtGray
Color clGray
ColorDisable clLtGray
Pen 2
Format 8,2,trR //подгонка текстов по правым краям клеток
Fon clRed,clYellow, clLtGray, clLtGray, clLtGray,clYellow,clRed
TableTI 1,2,clBlack,10/20,100/40
}
...
}
```

16.14. TableTC – рисует таблицу состояний телесигналов с именами столбцов и строк. Правила функционирования и синтаксис, в основном, совпадают с описанными для команды TableTI. Отличие состоит в том, что вместо значений сигналов в клетках таблицы выводятся круги, прямоугольники или тексты о состоянии сигналов. Число параметров на единицу больше. Окраска каждого круга/прямоугольника/текста определяется состоянием телесигнала и предписанными значениями в предшествующей команде Colors.

В процессе работы круги/прямоугольники недостоверных сигналов окрашиваются в цвет предшествующей команды ColorDisable.

Синтаксис команды:

TableTC (число рядов),(число столбцов) ,[цвет текста],[x1]/(y1),(x2)/(y2),[тип фигуры]
Типы параметров: TableTC *Int,Int,Clr,Point,Point,Ident*

Тип фигуры определяет способ графического отображения сигналов в клетках таблицы.

Принимает следующие значения:

tbCirc – круг;

tbRect – прямоугольник.

tbText – имя состояния сигнала. Если значение сигнала равно нулю, то текст с именем состояния берется для данного сигнала из столбца TextNull таблицы ТС БД настроек АРМ. Если значение сигнала равно единице, то из столбца TextOne таблицы ТС.

Синтаксис команды **Colors**:

Colors [цвет нулевого сигнала],[цвет единичного сигнала]

Типы параметров: Colors *Clr,Clr*

Если необходимо иметь в таблице незаполненные клетки, то при вызове CallCell задайте среди фактических параметров в команде SignТС слова “null” для соответствующих сигналов. Команда Format определяет подгонку имен строк и столбцов (влево, по центру, вправо). Вид таблицы определяет команда Vid. При наличии команды Vid Brd рекомендуется задать Rep не менее 3.

Команда Columns настраивает столбцы таблиц на вывод имен, времен измерений или значений сигналов (в соответствии с заданным типом фигуры). При отсутствии команды Columns выводятся значения сигналов. Позволяет задать ширины столбцов.

Синтаксис команды **Columns**:

Columns [знак][тип информации в столбце 1][ширина столбца 1], [знак][тип информации столбце 2] [ширина столбца 2]...

Типы параметров: Columns [*Ident*]*Ident*[*Int*], [*Ident*]*Ident*[*Int*],...

Знак – необязательный параметр (тип - Ident). Может являться с одним из следующих символов:

- 1) Знак отсутствует или команда Columns отсутствует. Текст в клетке имеет цвет, заданный третьим параметром команды TableТС. Цвет фона клетки выбирается из предшествующей команды Colors в зависимости от номера интервала. Номер интервала вычисляется по значению приписанного ячейке телесигнала.
- 2) ‘-’ (минус). Означает отказ от раскраски строк и фона клеток столбца по состоянию сигналов, прописанных в команде Signals. Их цвета остаются исходными.
- 3) ‘+’ (плюс). Цвет текста определяет предшествующая команда Colors по номеру интервала. Интервал вычисляется по значению телесигнала (0, 1). Цвет клетки не меняет свою исходную окраску, т.е. имеет цвет, заданный командой Brush.

Тип информации в столбце – обязательный параметр (тип Ident). Является одним из следующих идентификаторов:

- D или Date - выводит даты измерения сигналов
- T или Time - выводит времена измерения сигналов
- DT или DateTime - выводит даты и времена измерения сигналов
- V или Value – выводит значения сигналов
- N или Name - выводит имена сигналов, заданные в таблице сигналов БД настроек АРМ.

Ширина столбца - необязательный член параметра (тип - Int). Является целым числом. При его отсутствии производится автоматический расчет ширины столбца на основании выводимой информации.

Команда Columns без параметров отключает действие предыдущей команды Columns.

Пример:

```
Cell TabТС_1-2  
{
```

```

SignTC Sw1,Sw2
Font 8,MS Sans Serif
FontStyle
Vid 3D
Signals Sw1,Sw2
Colors clRed,clGreen
ColorDisable clLtGray
Texts "Связь ", "Объект 1", "Объект 2"
Brush clLtGray
Color clLtGray
Pen 2
Format 8,2,trC //Центрируем строки
TableTC 1,2,clBlack,10/20,100/40,tbRect
}

```

Пример2:

В программе:

```

Color clBlack
Text 160/380,"ПС Паклиновская 110 кВ ТН-1"
SignTC 1_9_70
CallCell 160/395, TabTC_Text,"TCVal_Text"

```

В Library:

```

Cell TabTC_Text
{SignTC c1
Font 8,Tahoma
FontStyle
Brush clLtGray
Color clLtGray
Pen 5
Signals c1
Colors clGreen,clRed
ColorDisable clLtGray
Texts "Полож.ШР" //Имена столбцов не выводим
Vid Brd
Format 8,2,trC //Центрируем строки
TableTC 1,1,clBlack,0/0,150/26,tbText
}

```

На экранной форме:

ПС Паклиновская 110 кВ ТН-1



16.15. Сигналы “Телеизмерение”

Поступающие от сервера на вход АРМ телеизмерения содержат следующую информацию об их состоянии:

FL_MaxAvar - Превышена верхняя аварийная граница
 FL_MinAvar - Значение параметра меньше нижней аварийной границы
 FL_MaxPrAvar - Превышена верхняя предаварийная граница
 FL_MinPrAvar - Значение параметра меньше нижней предаварийной границы
 FL_MaxTechn - Превышена верхняя технологическая граница
 FL_MinTechn - Значение параметра меньше нижней технологической границы
 FL_ErrIzm - код АЦП ниже допустимого, 2 - выше допустимого, 3 - недопустимый адрес параметра
 FL_Apert - Превышена апертура
 FL_Period - Истек период контроля
 FL_Hand - Значение введено вручную
 FL_ReakCtl - Реакция на управление
 FL_Enable - Параметр стал доступен
 FL_Disable - Параметр недоступен - 1, Ошибка УСО - 2, недостоверен - 3
 FL_PeriodSum - Истек период суммирования
 Value - Значение параметра. Число с плавающей точкой.

Флаги FL_MaxAvar, FL_MinAvar, FL_MaxPrAvar, FL_MinPrAvar, FL_MaxTechn, FL_MinTechn используются для вычисления интервала, в котором находится текущее значение параметра. Только один из перечисленных флагов может иметь единичное значение. Остальные – нулевые. Сигналу присваивается номер интервала на числовой шкале возможных значений в соответствии с правилами, перечисленными в следующей таблице:

Таблица 2. Правила вычисления номера интервала телеизмерений.

Номер интервала	Условия	Наименование
1	FL_MinAvar=1	Авария по нижней границе
2	FL_MinPrAvar=1	Предавария по нижней границе
3	FL_MinTechn=1	Ниже технологической границы
4	Все флаги нулевые	Сигнал в пределах технологических границ
5	FL_MaxTechn=1	Выше технологической границы
6	FL_MaxPrAvar=1	Предавария по верхней границе
7	FL_MaxAvar=1	Авария по верхней границе

Флаги FL_ErrIzm, FL_Enable, FL_Disable определяют достоверность поступившего сигнала. Сигнал достоверен, если FL_ErrIzm=0 и FL_Enable=1 и FL_Disable=0.

16.16. Команды обработки телеизмерений

Все команды вида xxxxTI отображают значения и/или состояния телеизмерений. Имя телеизмерения задается в командах в качестве одного из параметров или перечисляются в команде Signals перед командами TableTI и TableTC. Именем сигнала может быть одно из формальных имен, описанное в предложении SignTI ячейки. В процессе работы окна графики формальные имена сигналов ячеек заменяются фактическими.

В команде Colors каждому интервалу задается цвет отображения состояния сигнала. После поступления в систему очередного телеизмерения производится вычисление

номера соответствующего ему интервала по таблице 2. Номер интервала определяет номер параметра в команде Colors, который и задает цвет для исполняемой команды. Команда Colors должна предшествовать командам: LevelTI, ArrowTI, MeterTI, PolylineTI, RectTI, RoundRectTI, EllipseTI, TextTI, MultilineTI, MultigonTI, MultiTextTI, ValueTI, TimeValueTI, TextValueTI. Если поступившее телеизмерение недостоверно - цвет рисунка определяет предшествующая команда ColorDisable. При исполнении команды MultiTextTI по номеру интервала определяют координату начала и текст из предшествующих команд Points и Texts, также имеющих по семь параметров – координат и текстов соответственно. Командам MultilineTI, MultigonTI, помимо Colors, должны предшествовать семь команд Points. Первая команда Points определяет выводимую на экран фигуру, если номер интервала равен единице. Вторая команда Points – для второго интервала и т.д. Допускаются отрицательные значения координат.

Синтаксис команды **Colors**:

Colors [цвет первого уровня],[цвет второго уровня], ..., [цвет седьмого уровня]

Типы параметров: Colors *Clr,Clr*,...

Для правильной работы команд LevelTI, MeterTI и ArrowTI используемые в них телеуправления должны иметь осмысленные значения в следующих полях таблицы TI БД настроек АРМ:

MinVal, MaxVal – предельные значения сигнала

MinEmerg,MaxEmerg – аварийные границы сигнала

MinSubEmerg,MaxSubEmerg – предаварийные границы сигнала

MinTec,MaxTec- технологические границы сигнала

На числовой шкале значения перечисленных полей располагаются в следующем порядке:

MinVal, MinEmerg, MinSubEmerg, MinTec, MaxTec, MaxSubEmerg, MaxEmerg, MaxVal.

Имеем семь интервалов, которые определяют цвет отображаемого уровня по значению поступившего телеизмерения.

Для работы команд GraphTI используемые в них телеуправления должны иметь значения в полях MinVal, MaxVal таблицы TI БД настроек АРМ. Поля MinVal, MaxVal используются для ограничения значений сигналов.

Использование телеизмерения в командах ButtonTC TI, RoundButtonTC TI специфично. В них предполагается, что телеизмерение содержит целочисленное значение в заданном диапазоне, начинающимся с нуля, и указывает – какой телесигнал из списка Signals отвечает за прорисовку кнопки по алгоритму работы команд ButtonTC, RoundButtonTC.

Использование телеизмерения в командах TimeValueTI и TextValueTI тоже имеют специфику. В команде TextValueTI предполагается, что телеизмерение содержит целочисленное значение в заданном диапазоне, начинающимся с нуля, и указывает – какой из текстов следует выбрать из команды Tests для последующего вывода на экран монитора. В команде TimeValueTI предполагается, что телеизмерение содержит целочисленное значение времени в секундах, по которому формируется стандартная строка времени в формате: чч:мм:сс, которая и выводится на экран монитора.

16.17. Телесигналы.

Поступающие от сервера на вход АРМ телесигналы содержат следующую информацию об их состоянии:

Value - Значение дискрета (0 - вход разомкнут, 1 - замкнут)

FL_Hand - значение введено вручную

FL_Izm - параметр изменил состояние

FL_Period - истек период контроля
FL_Enable - параметр стал доступен
FL_ReakCtl - реакция на управление
FL_Disable - Параметр недоступен - 1, Ошибка УСО - 2, недостоверен - 3

Флаги FL_Enable, FL_Disable определяют достоверность поступившего сигнала. Сигнал достоверен, если FL_Enable=1 и FL_Disable=0.

Все команды вида xxxxTC отображают состояния телесигналов. Именем сигнала в команде может быть одно из формальных имен, описанное в предложении SignTC ячейки. Перед командой TableTC формальные имена сигналов перечисляются в команде Signals. Цвет рисунка исполняемой команды определяет значение поступившего телесигнала. Цвета для нулевого и единичного значений телесигналов, в основном, перечисляют в параметрах команды. Если поступивший телесигнал недостоверен - цвет рисунка задает предшествующая команда ColorDisable.

16.18. Fon – сообщает последующим командам цвет фона (подложки), на котором рисуются элементы изображения. Действует в пределах ячейки до появления следующей команды Fon. В разделе Picture не использовать.

Синтаксис команды:

Fon [цвет фона1], [цвет фона2],... - множество параметров
Типы параметров: Fon *Clr,Clr*,...

По умолчанию фоном считается текущее значение команды Brush.

Команда Fon используется командами: TableTI, MultilineTC, MultigonTC, MultiTextTC, TextTC, MultilineTI, MultigonTI, MultiTextTI, TextTI, ValueTI, TimeValueTI, TextValueTI. Перечисленные команды объединяет возможность вывода переменных картинок или текстов в различных местах экрана в зависимости от состояния сигнала.

Для команд ValueTI, TextTC, TextTI задают команду Fon с одним параметром. Для команд MultilineTC, MultigonTC, MultiTextTC – с двумя параметрами. Для команд TableTI, MultilineTI, MultigonTI, MultiTextTI – с семью параметрами. Число параметров зависит от числа местоположений выводимых командой картинок или текстов. Для телеизмерений номер выбираемого параметра из команды Fon зависит от номера интервала, в который попадает текущее значение телеизмерения. Для телесигналов – нулевое значение определяет первый параметр, единичное значение определяет второй параметр. Если в команде Fon цвет фона задан неверно или по умолчанию цвет подложки в месте вывода не соответствует текущему значению Brush – произойдет искажение цвета выводимой картинки или текста.

Пример: Fon clWhite

16.19. LevelTI – рисует прямоугольник с полосой уровня внутри. Высота уровня зависит от текущего значения телеизмерения, заданного в параметрах команды. Ориентацию уровня (вертикальная, горизонтальная) определяет третий параметр команды Opt, которую следует располагать перед командой LevelTI. Параметр может содержать один из следующих идентификаторов:

oAny – ориентация определяется по соотношению сторон прямоугольника. Если высота прямоугольника меньше его ширины, то имеем горизонтальную ориентацию. В этом случае основанием уровня является левая сторона прямоугольника. При вертикальной ориентации основанием уровня является нижняя сторона прямоугольника.

oUp – ориентация снизу вверх.
oRight – ориентация слева направо.

Заданное в параметрах команды телеизмерение должно иметь осмысленный набор значений в следующих полях таблицы TI из БД настроек АРМ:

MinVal, MaxVal – предельные значения сигнала
MinEmerg, MaxEmerg – аварийные границы сигнала
MinSubEmerg, MaxSubEmerg – предаварийные границы сигнала
MinTec, MaxTec- технологические границы сигнала

На числовой шкале значения перечисленных полей располагаются в следующем порядке: MinVal, MinEmerg, MinSubEmerg, MinTec, MaxTec, MaxSubEmerg, MaxEmerg, MaxVal.

Имеем семь интервалов, которые определяют цвет отображаемого уровня по значению поступившего телеизмерения. Цвета интервалов перечисляются в команде Colors. С левого края прямоугольника (для вертикальной ориентации) или сверху (для горизонтальной ориентации) выводится шкала в виде цветовых полос, отображающих диапазоны аварийных, предаварийных и нормальных значений для заданного в команде сигнала телеизмерения. Видимостью шкалы управляет второй параметр команды Opt. Параметр может принимать одно из следующих значений:

oScaleOff – шкала выключена (невидима).
oScaleOn – шкала включена.

Синтаксис команды:

LevelTI [имя сигнала],(x1)/(y1),(x2)/(y2) - координаты прямоугольника.

Типы параметров: LevelTI *Ident,Point,Point*

Ширина контура прямоугольника, цвет контура прямоугольника и фоновый цвет заливки прямоугольника задаются командами Pen, Color и Brush. Команда Vid определяет вид контура. При наличии команды Vid Brd рекомендуется задать Pen не менее 3. Команда Colors должна иметь семь параметров.

В середине прямоугольника контура выводится текст. Состав текста определяет первый параметр команды Opt. Параметр может содержать один из следующих идентификаторов:

oNo – текст отсутствует.

oPercent – текстом является процентное отношение значения текущего сигнала к максимально допустимому его значению в заданном диапазоне. Диапазон возможных значений сигнала задается в полях MinVal, MaxVal таблицы TI БД настроек АРМ.

oValue - текстом является значение текущего сигнала.

oValueUnit - текстом является значение текущего сигнала с его единицей измерения.

Параметры шрифта текста определяют команды Font и FontStyle.

При отсутствии команды Opt считается, что задано: Opt oPercent,oScaleOff,oAny.

Пример:

Font 8,Arial

FontStyle

Pen 2

Color clLtGray

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

Brush fffdcc

Opt oValue, oScaleOn

LevelTI sw, 0/60,70/75

Вид команды LevelTI при попадании значения сигнала в интервал аварийных значений:

16.20. ArrowTI – рисует стрелку в сегменте, ограниченном снизу хордой. Дуга сегмента вписывается в прямоугольник с координатами $(x1)/(y1), (x2)/(y2)$, заданными в параметрах команды. Дуга имеет угол 180 градусов и расположена над хордой. Хордой сегмента является нижняя грань прямоугольника. Если высота прямоугольника равна половине ширины, то в качестве дуги имеем половинку окружности, иначе, половинку эллипса. Угол наклона стрелки зависит от значения сигнала телеизмерения (заданного в параметрах команды), и ограничивается диапазоном его изменений. Отсчет угла наклона – по часовой стрелке. Неподвижный конец стрелки располагается в середине хорды. Длина стрелки составляет радиус дуги. Цвет и ширина контура хорды указываются предшествующими командами Color и Pen. Фон хорды определяет команда Brush. Толщина стрелки по умолчанию равна единице.

Дуга сегмента разбивается на семь участков. Заданное в параметрах команды телеизмерение должно иметь осмысленный набор значений в следующих полях таблицы TI из БД настроек АРМ:

- MinVal, MaxVal – предельные значения сигнала
- MinEmerg, MaxEmerg – аварийные границы сигнала
- MinSubEmerg, MaxSubEmerg – предаварийные границы сигнала
- MinTec, MaxTec- технологические границы сигнала

На числовой шкале значения перечисленных полей располагаются в следующем порядке: MinVal, MinEmerg, MinSubEmerg, MinTec, MaxTec, MaxSubEmerg, MaxEmerg, MaxVal.

Имеем семь интервалов, которые определяют относительные протяженности участков дуги. Цвета участков перечисляются в команде Colors. Шкала интервалов в виде цветных полос рисуется на дуге сегмента. Видимостью шкалы управляет второй параметр команды Opt, которую следует располагать перед командой ArrowTI. Параметр может принимать одно из следующих значений:

- oScaleOff – шкала выключена (невидима).
- oScaleOn – шкала включена.

По умолчанию шкала выключена.

Синтаксис команды:

ArrowTI [цвет стрелки],[имя сигнала],[$x1)/(y1), (x2)/(y2)$)

Типы параметров: ArrowTI *Clr, Ident, Point, Point*

Команда Vid определяет вид контура. При наличии команды Vid Brd рекомендуется задать Pen не менее 3.

В сегменте выводится текст. Содержимое и местоположение текста определяет первый параметр предварительно заданной команды Opt. Параметр может содержать один из следующих идентификаторов:

- oNo – текст отсутствует.
- oPercent – текстом является процентное отношение значения текущего сигнала к максимально допустимому его значению в заданном диапазоне. Диапазон возможных значений сигнала задается в полях MinVal, MaxVal таблицы TI БД настроек АРМ. Текст располагается у основания сегмента.
- oValue - текстом является значение текущего сигнала. Текст располагается у основания сегмента.
- oValueUnit - текстом является значение текущего сигнала с его единицей измерения. Текст располагается по высоте в середине сегмента.
- oMiddlePercent – текстом является процентное отношение значения текущего сигнала к максимально допустимому его значению в заданном диапазоне. Диапазон возможных

значений сигнала задается в полях MinVal, MaxVal таблицы TI БД настроек АРМ. Текст располагается по высоте в середине сегмента.

oMiddleValue - текстом является значение текущего сигнала. Текст располагается по высоте в середине сегмента.

oMiddleValueUnit - текстом является значение текущего сигнала с его единицей измерения. Текст располагается по высоте в середине сегмента. Текст располагается в середине сегмента.

Команда Opt, также, позволяет управлять видом и толщиной стрелки посредством следующих идентификаторов:

oArrowX, где X является числом и задает толщину стрелки. Например: oArrow3.

oOutArrowX, где X является числом и задает толщину стрелки. При отсутствии данной команды стрелка рисуется от центра вращения, расположенного в середине хорды. При наличии данной команды рисуется только верхняя часть стрелки, длина которой составляет 20 процентов от полной длины стрелки.

Параметры шрифта текста определяют команды Font и FontStyle.

При отсутствии команды Opt считается, что задано oPercent, а толщина стрелки равна единице.

Пример:

Font 8,Arial

FontStyle

Pen 2

Color clLtGray

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

Brush fffdcc

Opt oMiddleValueUnit, oScaleOn

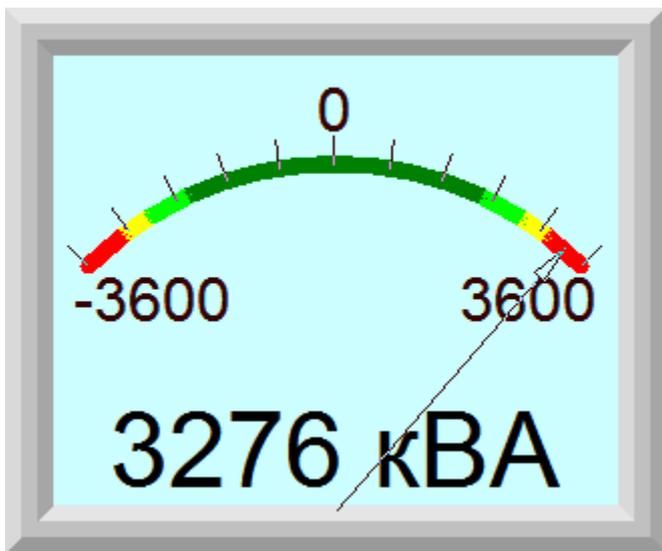
ArrowTI clBlack,sw,0/0,80/40

Вид команды ArrowTI:



16.21. MeterTI – имитирует стрелочный индикатор для заданного телеизмерения. Может быть настроен на один из следующих двух видов:

1) Ход стрелки – 90 градусов.



2) Ход стрелки – 270 градусов.

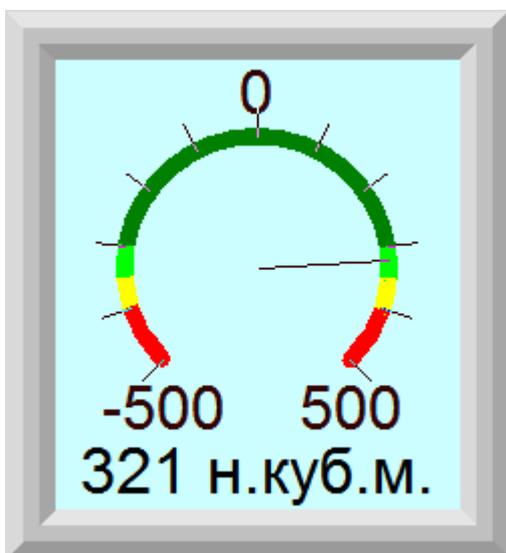


Рисунок индикатора вписывается в прямоугольник с координатами $(x1)/(y1),(x2)/(y2)$, заданными в параметрах команды. Имеет рамку, мерную шкалу в виде дуги с рисками, стрелку и строку с текущим показанием. Угол наклона стрелки зависит от значения сигнала телеизмерения (заданного в параметрах команды), и ограничивается диапазоном его изменений. Отсчет угла наклона – по часовой стрелке. Цвет и ширина рамки индикатора указываются предшествующими командами Color и Pen. По усмотрению пользователя можно указать вид мерной шкалы и, соответственно, диапазон хода стрелки индикатора. Допускаются два варианта:

- ход стрелки – 90 градусов
- ход стрелки – 270 градусов

Для первого случая рекомендуется задать отношение сторон прямоугольника кратным значению 40:33, а для второго – 60:66.

Фон рабочей поверхности индикатора определяет команда Brush. Толщина стрелки всегда 1. В зависимости от размера картинки индикатора стрелка рисуется либо в упрощенном виде (линия), либо в виде линии с наконечником (при достаточном разрешении).

Дуга мерной шкалы разбивается на семь участков. Заданное в параметрах команды телеизмерение должно иметь осмысленный набор значений в следующих полях таблицы TI из БД настроек АРМ:

MinVal, MaxVal – предельные значения сигнала

MinEmerg, MaxEmerg – аварийные границы сигнала

MinSubEmerg, MaxSubEmerg – предаварийные границы сигнала

MinTec, MaxTec- технологические границы сигнала

На мерной шкале значения перечисленных полей располагаются в следующем порядке:

MinVal, MinEmerg, MinSubEmerg, MinTec, MaxTec, MaxSubEmerg, MaxEmerg, MaxVal.

Имеем семь интервалов, которые определяют относительные протяженности участков дуги. Цвета участков перечисляются в команде Colors. Шкала интервалов в виде цветных полос рисуется на дуге сегмента. Видом шкалы управляет второй параметр команды Opt, которую следует располагать перед командой MeterTI. Параметр может принимать одно из следующих значений:

oScaleOff – шкала выключена, ход стрелки – 90 градусов.

oScaleOn – шкала включена, ход стрелки – 90 градусов.

oRoundScaleOff – шкала выключена, ход стрелки – 270 градусов.

oRoundScaleOn – шкала включена, ход стрелки – 270 градусов.

По умолчанию шкала выключена и рисуется сплошная дуга цвета Color без уточнения границ аварийных и т.д. значений. Вдоль шкалы выводятся тексты с минимальным, средним и максимальным значениями телеизмерения.

Синтаксис команды:

MeterTI [цвет стрелки],[имя сигнала],[x1]/(y1),(x2)/(y2)

Типы параметров: MeterTI *Clr, Ident, Point, Point*

Команда Vid определяет вид рамки (плоская, выпуклая). При наличии команды Vid Brd рекомендуется задать Rep не менее 3.

В видней части рабочей поверхности индикатора выводится текст. Содержимое текста определяет первый параметр предварительно заданной команды Opt. Параметр может содержать один из следующих идентификаторов:

oNo – текст отсутствует.

oPercent – текстом является процентное отношение значения текущего сигнала к максимально допустимому его значению в заданном диапазоне. Диапазон возможных значений сигнала задается в полях MinVal, MaxVal таблицы TI БД настроек АРМ.

oValue - текстом является значение текущего сигнала.

oValueUnit - текстом является значение текущего сигнала с его единицей измерения.

Параметры шрифта текста определяют команды Font и FontStyle. При отсутствии команды Opt считается, что задано oPercent.

Команда Opt, также, позволяет управлять видом и толщиной стрелки посредством следующих идентификаторов:

oArrowX, где X является числом и задает толщину стрелки. Например: oArrow3.

oOutArrowX, где X является числом и задает толщину стрелки. При отсутствии данной команды стрелка рисуется от центра вращения. При наличии данной команды рисуется только верхняя часть стрелки, длина которой составляет 20 процентов от полной длины стрелки.

Пример:

Cell Indicator
{SignTI ti

```

Font 8,Arial
FontStyle
Vid Brd
Pen 6
Color clLtGray
Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed
Brush fffdcc
Opt oValueUnit,oScaleOn
MeterTI clBlack,ti,0/0,80/66
}

```

16.22. PolylineTC – рисует кусочно-линейную кривую. Цвет закраски изменяется при изменении телесигнала (один из формальных параметров ячейки). Указываются цвета для нулевого и единичного значений сигнала. Если сигнал имеет статус аварии – происходит мигание до квитирования (чередование цвета полилинии с цветом команды ColorAvar один раз в секунду). Неопределенному состоянию сигнала соответствует цвет, заданный командой ColorDisable.

Ширина линий предварительно задается командой Pen.

Синтаксис команды:

PolylineTC [цвет нулевого сигнала],[цвет единичного сигнала],[имя сигнала]

Типы параметров: PolylineTC *Clr,Clr,Ident*

Команде PolylineTC должны предшествовать команды Vid, Lines, Pen, ColorAvar, ColorDisable, Points. Могут предшествовать команды ColorBlink, SignalBlinks и StateBlinks.

Синтаксис команды **Points**:

Points (x1)/(y1),(x2)/(y2) ... - перечень координат точек.

Типы параметров: Points *Point,Point,...*

Вид PolylineTC определяет команда Vid.

1) При задании Vid 2D или наличии в файле конфигурации предложения volume=no команда PolylineTC рисуется круглой маской, диаметр которой задает команда Pen. Углы наклонов составляющих ее отрезков произвольны.

2) При задании Vid 3D команда PolylineTC рисуется как объемная фигура с обрезанными концами.

3) При задании Vid Brd команда PolylineTC имеет вид рельефного бордюра с обрезанными концами.

В случаях 2) и 3) имеем следующие особенности:

- В файле конфигурации необходимо наличие команды volume=yes.

- Команда PolylineTC сглаживает углы между составляющими ее отрезками. Параметры сглаживания задает первый параметр команды Lines.

- Составляющие PolylineTC отрезки (кодируются в предшествующей команде Points) должны быть параллельны осям координат. При нарушении этого правила рисование происходит в режиме Vid 2D.

- Если в Points первая и последняя координаты совпадают, то происходит их сглаживание в соответствии с командой Lines и имеем замкнутую сглаженную фигуру во всех ее углах.

- Конфигурацию концов полилинии определяют второй и третий параметры команды Lines, которые задают глубину проникновения (врезки) начала (конца) полилинии в другую соприкасающуюся полилинию. При нулевых значениях параметров врезок полилинии имеют ровные окончания.

С целью поддержки задела допускается отключить прорисовку PolylineTC в режимах Vid 3D и Vid Brd, всегда рисуя в режиме Vid 2D вне зависимости от заданного Vid. Отключение производится в конфигуураторах графического редактора (Graph.exe) и АРМ (Arm.exe).

Пример:

```
ColorBlink clBlack
SignalBlinks blink //включает мигание при единичном значении телесигнала blink
StateBlinks 1
Points 10/20,10/40
PolylineTC clRed,clGreen,sw
SignalBlinks //отключает применение SignalBlinks для последующих команд
```

16.23. PolygonTC – рисует замкнутую закрашенную фигуру с кусочно-линейной границей. Синтаксис и поведение команды аналогичны команде PolylineTC. Команда Brush определяет цвет закрашки контура. Цвет и толщина контуров задаются командами Color, Pen. Отображает аварийное состояние. Могут предшествовать команды ColorBlink, SignalBlinks и StateBlinks. Неопределенному состоянию сигнала соответствует цвет, заданный командой ColorDisable.

Пример:

```
Pen 3
ColorAvar clBlue
Points 10/20,10/40,20/20
PolygonTC clRed,clGreen,sw
```

16.24. RectTC - рисует закрашенный прямоугольник. Ширину и цвет контура предварительно задают команды Pen, Color. Поведение команды идентично предыдущей. Команда Brush определяет цвет закрашки контура. Допускается использовать команду Vid. При наличии команды Vid Brd рекомендуется задать Pen не менее 3. Могут предшествовать команды ColorBlink, SignalBlinks и StateBlinks. Неопределенному состоянию сигнала соответствует цвет, заданный командой ColorDisable.

Синтаксис команды:

```
RectTC [цвет нулевого сигнала],[цвет единичного сигнала],[имя сигнала] ,
(x1)/(y1),(x2)/(y2) - координаты верхней левой и правой нижней точек прямоугольника.
Типы параметров: RectTC Clr,Clr,Ident,Point,Point
```

Отображает аварийное состояние.

Пример: RectTC clRed,clGreen,sw,10/20,10/40

16.25. RoundRectTC - рисует закрашенный прямоугольник со скругленными углами. Цвет закрашки скругленного прямоугольника изменяется при изменении телесигнала (один из формальных параметров ячейки). Указываются цвета для нулевого и единичного значений сигнала. Если сигнал имеет статус аварии – происходит мигание до квитирования (чередование цвета полилинии с цветом команды ColorAvar один раз в секунду). Ширину и цвет контура предварительно задают команды Pen, Color. Допускается использовать команду Vid. При наличии команды Vid Brd рекомендуется задать Pen не менее 3.

Могут предшествовать команды ColorBlink, SignalBlinks и StateBlinks. Неопределенному состоянию сигнала соответствует цвет, заданный командой ColorDisable.

Синтаксис команды:

RoundRectTC [цвет нулевого сигнала],[цвет единичного сигнала],[имя сигнала] ,
(x1)/(y1),(x2)/(y2),(x3)/(y3)

(x1)/(y1) - координата верхнего левого угла

(x2)/(y2) - координата правого нижнего угла прямоугольника.

Кривая скругленных углов совпадает с кривой эллипса шириной x3 и высотой y3.

Типы параметров: RoundRectTC *Clr,Clr,Ident,Point,Point,Point*

Отображает аварийное состояние.

Пример: RoundRectTC clRed,clGreen,sw,10/40,10/40,20/20

16.26. EllipseTC - рисует закрашенный эллипс. Поведение команды идентично предыдущей.

Синтаксис команды:

EllipseTC [цвет нулевого сигнала],[цвет единичного сигнала],[имя сигнала],
(x1)/(y1),(x2)/(y2) - координаты верхней левой и правой нижней точек охватывающего
прямоугольника.

Типы параметров: EllipseTC *Clr,Clr,Ident,Point,Point*

Отображает аварийное состояние. Могут предшествовать команды ColorBlink, SignalBlinks и StateBlinks. Неопределенному состоянию сигнала соответствует цвет, заданный командой ColorDisable.

Эллипс можно нарисовать объемным, если перед командой задать Pen=0 и Vid 3d, а в файл конфигурации внести предложение volume=yes.

Пример: EllipseTC clRed,clGreen,sw,10/20,30/40

16.27. GraphTI – прорисовка графиков сигналов телеизмерений в темпе их поступления в указанном окне.

Синтаксис команды:

GraphTI (диапазон в сек.),[цвет графика],[имя сигнала] (x1)/(y1),(x2)/(y2) - координаты
прямоугольника.

Типы параметров: GraphTI *Int,Clr,Ident,Point,Point*

Ширина контура, цвет контура и цвет фона предварительно задаются командами Pen, Color и Brush. Допускается использовать команду Vid. При наличии команды Vid Brd рекомендуется задать Pen не менее 3.

Пример:

GraphTI 50,clRed,power,10/20,100/40

16.28. PolylineTI – рисует кусочно-линейную кривую. Цвет зависит от значения сигнала телеизмерения (один из формальных параметров ячейки), набора интервалов в таблице TI и заданного диапазона его изменений. Каждый интервал имеет свою окраску. Перечень цветов задается командой Colors с семью параметрами. Неопределенному

состоянию сигнала соответствует цвет, заданный командой ColorDisable. Если сигнал имеет статус аварии – происходит мигание до квитирования (чередование цвета ячейки с цветом команды ColorAvar). Ширина линий предварительно задается командой Pen.

Синтаксис команды:

PolylineTI [имя сигнала]

Типы параметров: PolylineTI *Ident*

Команде PolylineTI должны предшествовать команды Vid, Lines, Pen, ColorAvar, Colors и Points.

Синтаксис команды Points:

Points (x1)/(y1),(x2)/(y2) ... - перечень координат точек.

Типы параметров: Points *Point,Point,...*

Вид PolylineTI определяет команда Vid.

1) При задании Vid 2D или наличии в файле конфигурации предложения volume=no команда PolylineTI рисуется круглой маской, диаметр которой задает команда Pen. Углы наклонов составляющих ее отрезков произвольны.

2) При задании Vid 3D команда PolylineTI рисуется как объемная фигура с обрезанными концами.

3) При задании Vid Brd команда PolylineTI имеет вид рельефного бордюра с обрезанными концами.

В случаях 2) и 3) имеем следующие особенности:

- В файле конфигурации необходимо наличие команды volume=yes.

- Команда PolylineTI сглаживает углы между составляющими ее отрезками. Параметры сглаживания задает первый параметр команды Lines.

- Составляющие PolylineTI отрезки (кодируются в предшествующей команде Points) должны быть параллельны осям координат. При нарушении этого правила рисование происходит в режиме Vid 2D.

- Если в Points первая и последняя координаты совпадают, то происходит их сглаживание в соответствии с командой Lines и имеем замкнутую сглаженную фигуру во всех ее углах.

- Конфигурацию концов PolylineTI определяют второй и третий параметры команды Lines, которые задают глубину проникновения (врезки) начала (конца) полилинии в другую соприкасающуюся полилинию. При нулевых значениях параметров врезок полилинии имеют ровные окончания.

С целью поддержки задела допускается отключить прорисовку PolylineTI в режимах Vid 3D и Vid Brd, всегда рисуя в режиме Vid 2D вне зависимости от заданного Vid.

Отключение производится в конфигуураторах графического редактора (Graph.exe) и АРМ (Arm.exe).

Пример:

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

Points 10/20,10/40

PolylineTI sw

16.29. PolygonTI – рисует замкнутую закрашенную фигуру с кусочно-линейной границей. Синтаксис и поведение команды аналогичны команде PolylineTI. Команда Brush определяет цвет закрашки контура. Цвет и толщина контуров задаются командами Color, Pen. Отображает аварийное состояние.

Пример:

Pen 3

ColorAvar clBlue

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

Points 10/20,10/40,20/20

PolygonTI clRed,clGreen,sw

16.30. RectTI - рисует закрашенный прямоугольник. Поведение команды идентично предыдущей. Предварительно заданная команда Vid определяет вид контура. При наличии команды Vid Brd рекомендуется задать Pen не менее 3.

Синтаксис команды:

RectTI [имя сигнала],(x1)/(y1),(x2)/(y2) - координаты прямоугольника.

Типы параметров: RectTI *Ident,Point,Point*

Команде RectTI должны предшествовать команды Pen, Color, Brush, ColorAvar, ColorDisable, Vid, Colors.

Pen и Color определяют ширину и цвет контура. Brush – цвет закрашки контура. Отображает аварийное состояние.

Пример:

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

RectTI U,10/20,10/40

16.31. RoundRectTI - рисует закрашенный прямоугольник со скругленными углами. Цвет скругленного прямоугольника зависит от значения сигнала телеизмерения (один из формальных параметров ячейки), набора интервалов в таблице TI и заданного диапазона его изменений. Каждый интервал имеет свою окраску. Перечень цветов задается командой Colors с семью параметрами. Неопределенному состоянию сигнала соответствует цвет, заданный командой ColorDisable. Если сигнал имеет статус аварии – происходит мигание до квитирования (чередование цвета ячейки с цветом команды ColorAvar). Ширину и цвет контура предварительно задают команды Pen, Color. Допускается использовать команду Vid. При наличии команды Vid Brd рекомендуется задать Pen не менее 3.

Синтаксис команды:

RoundRectTI [имя сигнала] , (x1)/(y1),(x2)/(y2),(x3)/(y3)

(x1)/(y1) - координата верхнего левого угла

(x2)/(y2) - координата правого нижнего угла прямоугольника.

Кривые скругленных углов совпадают с секторами кривой эллипса шириной x3 и высотой y3.

Типы параметров: RoundRectTI *Ident,Point,Point,Point*

Отображает аварийное состояние.

Пример: RoundRectTI sw,10/40,10/40,20/20

16.32. EllipseTI - рисует закрашенный эллипс. Поведение команды идентично предыдущей.

Синтаксис команды:

EllipseTI [имя сигнала],(x1)/(y1),(x2)/(y2) - координаты верхней левой и правой нижней точек охватывающего прямоугольника.

Типы параметров: EllipseTI *Ident,Point,Point*

Команде EllipseTI должны предшествовать команды Pen, Brush, Color, ColorAvar, ColorDisable, Colors, Vid.

Pen и Color определяют ширину и цвет контура. Brush определяет цвет закраски эллипса.

При наличии команды Vid Brd рекомендуется задать Pen не менее 3.

Отображает аварийное состояние.

Эллипс можно нарисовать объемным, если перед командой задать Pen=0 и Vid 3d, а в файл конфигурации внести предложение volume=yes.

Пример:

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

EllipseTI U,10/20,10/40

16.33. Особенности работы команд TextTC, TextTI, ValueTI, TimeValueTI, TextValueTI.

При исполнении команд TextTC, TextTI, ValueTI, TimeValueTI, TextValueTI цвет фона задает команда Fon с одним параметром. При ее отсутствии фоном для выводимого текста считается текущее значение команды Brush. Команда Angle определяет угол поворота текста. Отсчет угла - от оси X против часовой стрелки. Не допускается совмещать изображения текстов команд TextTC, TextTI, ValueTI, TimeValueTI, TextValueTI с рисунками других специальных команд, вид или цвет которых могут меняться в процессе функционирования системы.

16.34. TextTC – рисует переменный текст в зависимости от состояния телесигнала.

Синтаксис команды:

TextTC [имя сигнала] , (x1)/(y1)

Типы параметров: TextTC *Ident,Point*

Где: (x1)/(y1) – координата левой верхней точки выводимого текста относительно базовой координаты подпрограммы, в которую помещена данная команда.

Команде TextTC должны предшествовать команды Texts и Colors в каждой по два значения. Команде TextTC может предшествовать команда Fon с одним значением. При ее отсутствии фоном для выводимого текста считается текущее значение команды Brush.

Угол поворота текста предварительно задается командой Angle.

Синтаксис команды **Texts**:

Texts [текст нулевого сигнала],[текст единичного сигнала]

Типы параметров: *Texts String,String,...*

Текст может состоять из множества строк, разделяемых символами “\n” (без кавычек).
Число строк в тексте – не более 10-и. Можно задать способ подгонки строк командой Trim.

Синтаксис команды **Colors**:

Colors [цвет нулевого сигнала],[цвет единичного сигнала]

Типы параметров: *Colors Clr,Clr,...*

Параметры шрифта определяют команды *Font* и *FontStyle*.

Пример:

Texts “Фидер включен”,”Фидер выключен”

Colors clBlue,clRed

TextTC sw,5/10

16.35. TextTI - рисует переменный текст в зависимости от состояния сигнала телеизмерения.

Синтаксис команды:

TextTI [имя сигнала],[x1]/(y1)

Типы параметров: *TextTI Ident,Point*

Где: (x1)/(y1) – координата левой верхней точки выводимого текста относительно базовой координаты подпрограммы, в которую помещена данная команда.

Команде *TextTI* должны предшествовать команды *Texts* и *Colors* с семью параметрами.

Команде *TextTI* может предшествовать команда *Fon* с одним параметром. При ее отсутствии фоном для выводимого текста считается текущее значение команды *Brush*.
Угол поворота текста предварительно задается командой *Angle*.

Синтаксис команды **Texts**:

Texts [текст первого интервала],[текст второго интервала], ..., [текст седьмого уровня]

Типы параметров: *Texts String,String,...*

Текст может состоять из множества строк, разделяемых символами “\n” (без кавычек).
Число строк в тексте – не более 10-и. Можно задать способ подгонки строк командой Trim.

Синтаксис команды **Colors**:

Colors [цвет первого интервала],[цвет второго интервала], ..., [цвет седьмого интервала]

Типы параметров: *Colors Clr,Clr,...*

Параметры шрифта определяют команды *Font* и *FontStyle*.

Пример:

Texts “Авария”,”Предаварийное состояние”,”Внимание”,”Норма”,” Внимание ”,”

Предаварийное состояние ”,” Авария ”

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

Fon clWhite

TextTI U,5/10

16.36. ValueTI – выводит значение сигнала телеизмерения в виде текста.

Синтаксис команды:

ValueTI [имя сигнала],[x1]/(y1)

Типы параметров: ValueTI *Ident,Point*

Где: (x1)/(y1) – координата левой верхней точки выводимого текста относительно базовой координаты подпрограммы, в которую помещена данная команда.

Команде ValueTI должна предшествовать команда Colors с семью параметрами. Команде ValueTI может предшествовать команда Fon с одним параметром. При ее отсутствии фоном для выводимого текста считается текущее значение команды Brush.

Предшествующая команда Format определяет форматирование выводимого значения.

Параметры шрифта определяют команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Синтаксис команды Colors:

Colors [цвет первого интервала],[цвет второго интервала], ..., [цвет седьмого интервала]

Типы параметров: Colors *Clr,Clr,...*

Пример:

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

Fon clWhite

Format 10,3

ValueTI U,5/10

16.37. TimeValueTI – выводит значение сигнала телеизмерения в виде строки времени.

Телеизмерение должно иметь целочисленное значение, которое рассматривается как значение времени в секундах. Формат выводимой строки имеет вид: чч:мм:сс.

Где: чч – часы, мм – минуты, сс – секунды.

Например, значением телеизмерения является счетчик времени выполнения команды в контроллере.

Синтаксис команды:

TimeValueTI [имя сигнала],[x1]/(y1)

Типы параметров: ValueTI *Ident,Point*

Где: (x1)/(y1) – координата левой верхней точки выводимого текста относительно базовой координаты подпрограммы, в которую помещена данная команда..

Команде TimeValueTI должна предшествовать команда Colors с семью параметрами.

Команде TimeValueTI может предшествовать команда Fon с одним параметром. При ее отсутствии фоном для выводимого текста считается текущее значение команды Brush.

Команда Angle определяет угол поворота текста. Параметры шрифта определяют команды Font и FontStyle.

Синтаксис команды **Colors**:

Colors [цвет первого интервала],[цвет второго интервала], ..., [цвет седьмого интервала]

Типы параметров: Colors *Clr,Clr,...*

Пример:

SignTI Timer

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

Fon clWhite

TimeValueTI Timer,5/10

16.38. TextValueTI – выводит текст, выбираемый по значению указанного телеизмерения. Телеизмерение должно содержать целочисленное значение в заданном диапазоне, начинающимся с нуля, и определяет – какой из текстов следует выбрать из команды Texts для последующего вывода на экран монитора.

Например, значением телеизмерения является номер шага выполняемой команды в контроллере, а команда Texts содержит имена выполняемых команд.

Синтаксис команды:

TextValueTI [имя сигнала],[x1)/(y1)

Типы параметров: ValueTI *Ident,Point*

Где: (x1)/(y1) – координата левой верхней точки выводимого текста относительно базовой координаты подпрограммы, в которую помещена данная команда.

Команде TextValueTI может предшествовать команда Fon с одним параметром. При ее отсутствии фоном для выводимого текста считается текущее значение команды Brush.

Команда Colors определяет цвет выводимого текста. Имеет семь цветов. Цвет выбирается по состоянию телеизмерения (авария, предупреждение, норма и т.д.).

Команда Angle определяет угол поворота текста. Параметры шрифта определяют команды Font и FontStyle.

Синтаксис команды Texts:

Texts [0-й текст], [1-й текст],[2-й текст], ...

Типы параметров: Texts *String,String,...*

Текст может состоять из множества строк, разделяемых символами “\n” (без кавычек).

Число строк в тексте – не более 10. Можно задать способ подгонки строк командой Trim.

Синтаксис команды Colors:

Colors [цвет 0-го текста],[цвет 1-го текста], ...

Типы параметров: Colors *Clr,Clr,...*

Пример:

SignTI Step

Color clRed

Fon clWhite

FontStyle fsBold

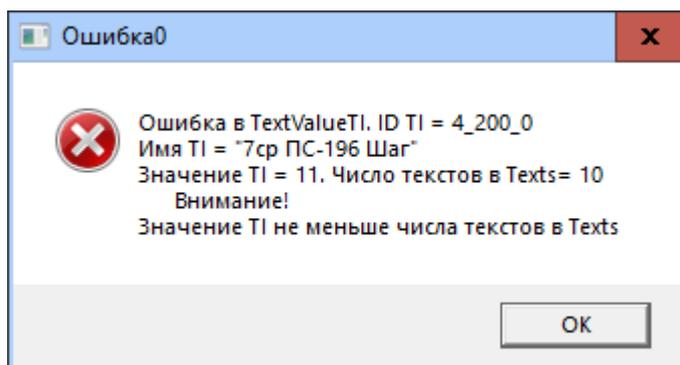
Font 10,MS Sans Serif

Texts "Шаг0","Шаг1","Шаг2","Шаг3","Шаг4","Шаг5","Шаг6","Шаг7","Шаг8","Шаг9"

TextValueTI Step,5/10

Внимание!

В процессе работы экранной формы, содержащей команды TextValueTI, по ошибке составителя прошивки контроллера возможно появление окна следующего содержания:



В котором сообщается, что значение телеизмерения превышает число текстов в команде Texts. Чтобы не загружать экран, появление таких сообщений ограничено числом 10. Последующие ошибки подобного рода игнорируются.

16.39. Особенности работы команд Multi...

Команды MultilineTC, MultilineTI, MultigonTC, MultigonTI, MultiTextTC, MultiTextTI рисуют изображения разными цветами в различных местах экрана в зависимости от состояния указанного сигнала. Перед выводом нового изображения старое стирается.

При исполнении команд MultilineTI, MultilineTC, MultigonTC, MultigonTI, MultiTextTC, MultiTextTI цвет комбинируется в каждом пикселе изображения с подложкой в соответствии с булевой операцией XOR (поразрядное сложение без переноса), что позволяет наблюдать изображение исполненной команды и картинку под ним. Следовательно, заданный цвет без искажений будет наблюдаться на белом фоне. Для коррекции цвета необходимо задать значение фона посредством команды Fon.

Не допускается совмещать изображения перечисленных команд с рисунками от других специальных команд, вид или цвет которых могут меняться в процессе функционирования системы.

Допускается совмещать выводимые на экран тексты от команд MultiTextTC, MultiTextTI с изображениями от команд MultilineTI, MultilineTC, MultigonTC, MultigonTI. Команды MultiTextTC, MultiTextTI должны следовать после команд MultilineTI, MultilineTC, MultigonTC, MultigonTI. В этих случаях цвета в команде Fon перед командами MultiTextTC, MultiTextTI должны повторять цвета команды Colors перед командами MultilineTI, MultilineTC, MultigonTC, MultigonTI. Также, допускается совмещать изображения от команд MultilineTI, MultilineTC с изображениями от команд MultigonTC, MultigonTI. Правило составления команды Fon соответствует предыдущему – Fon перед второй командой должен содержать цвета из Colors перед первой командой.

Пример совмещения изображений от команд MultigonTC и MultiTextTC. Выводим ромб с текстом на нем. Местоположение ромба и содержимое текста зависят от текущего состояния телесигнала.

```
Cell OverTC
{
  SignTC p
  Pen 1 //толщина контура ромба
  Color clLtGray //цвет контура ромба
  Points 0/10,20/20,40/10,20/0 //описание первого ромба для p со значением 0
  Points 0/30,20/40,40/30,20/20 //описание второго ромба для p со значением 1
  Fon clWhite,clWhite //Цвет фона на котором рисуем ромбы - белый
  //Рисуем один из ромбов в зависимости от значения телесигнала p,
  //предварительно стирая предыдущий ромб.
  MultigonTC clGreen,clRed,p
  //Фон для последующей команды MultiTextTC является цвет нарисованного ромба
  Fon clGreen,clRed
  Font 8, MS Sans Serif //с другими шрифтами м.б. искажения
  FontStyle
  //Местоположения текстов для обеих значений телесигнала p
  Points 190/4,190/34
  //Тексты для p=0 и p=1
  Texts "Nul","One"
```

```
//Рисуем один из текстов в зависимости от значения телесигнала p,  
//предварительно стирая предыдущий текст (на предыдущем ромбе).  
MultiTextTC clRed,clGreen,p  
}
```

16.40. MultilineTC - рисует кусочно-линейную кривую, положение и цвет которой зависит от состояния телесигнала. Цвет закрашки изменяется при изменении телесигнала. Указываются цвета для нулевого и единичного значений сигнала. Кусочно-линейные кривые определяются парой предшествующих команд Points. Первая команда Points описывает изображение для нулевого значения сигнала, а вторая – для единичного значения сигнала. Ширина линий предварительно задается командой Pen.

Синтаксис команды:

MultilineTC [цвет нулевого сигнала],[цвет единичного сигнала],[имя сигнала]

Типы параметров: MultilineTC *Clr,Clr,Ident*

Команде MultilineTC должны предшествовать команда Pen и две команды Points. Команде MultilineTC может предшествовать команда Fon с двумя параметрами. При ее отсутствии фоном считается текущее значение команды Brush.

Синтаксис команды **Points**:

Points (x1)/(y1),(x2)/(y2) ... - перечень координат точек кусочно-линейной кривой.

Типы параметров: Points *Point,Point,...*

Допускается команда Points без параметров. В этом случае изображение для соответствующего значения сигнала не выводится.

Синтаксис команды Fon:

Fon [Фон под местоположением кривых нулевого сигнала],[Фон под местоположением кривых единичного сигнала]

Пример:

Pen 3

Points 0/20,0/40

Points 10/20,10/40

Fon clWhite,clWhite

MultilineTC clRed,clGreen,sw

16.41. MultilineTI – рисует кусочно-линейную кривую, положение и цвет которой зависит от состояния флагов телеизмерения.

См. П.16.15. “Сигналы телеизмерений” и П.16.16. “Команды обработки телеизмерений”. Флаги телеизмерения определяют номер интервала. Каждому интервалу соответствует своя окраска и кусочно-линейная кривая. Набор цветов задается командой Colors. Набор кусочно-линейных кривых задается семью командами Points. Ширина линий предварительно задается командой Pen.

Команде MultilineTI может предшествовать команда Fon с семью параметрами. При ее отсутствии фоном считается текущее значение команды Brush.

Синтаксис команды:

MultilineTI [имя сигнала]

Типы параметров: MultilineTI *Ident*

Команде MultilineTI должны предшествовать команды Pen, Colors и семь команд Points.

Синтаксис команды Colors:

Colors [цвет первого интервала],[цвет второго интервала], ..., [цвет седьмого интервала]

Типы параметров: Colors *Clr,Clr,...*

Синтаксис команды Points:

Points (x1)/(y1),(x2)/(y2) ... - перечень координат точек.

Типы параметров: Points *Point,Point,...*

Допускается команда Points без параметров. В этом случае изображение для соответствующего значения сигнала не выводится.

Синтаксис команды Fon:

Fon [Фон под кривой первого интервала],[Фон под кривой второго интервала],..., [Фон под кривой седьмого интервала]

Типы параметров: Fon *Clr,Clr,...*

Пример:

Pen 3

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

Points 0/20,0/40

Points 10/20,10/40

Points 20/20,20/40

Points 30/20,30/40

Points 40/20,40/40

Points 50/20,50/40

Points 60/20,60/40

Fon clRed,clRed,clRed,clRed,clRed,clRed,clRed

MultilineTI sw

16.42. MultigonTC - рисует замкнутую закрашенную фигуру с кусочно-линейной границей, положение и цвет которой зависит от состояния телесигнала. Цвет закрашки изменяется при изменении телесигнала. Указываются цвета для нулевого и единичного значений сигнала. Кусочно-линейные контура определяются парой предшествующих команд Points. Первая команда Points описывает контур для нулевого значения сигнала, а вторая – для единичного значения сигнала. Цвет и толщина границ контуров задаются командами Color, Pen.

Синтаксис команды:

MultigonTC [цвет нулевого сигнала],[цвет единичного сигнала],[имя сигнала]

Типы параметров: MultigonTC *Clr,Clr,Ident*

Команде MultigonTC может предшествовать команда Fon с двумя параметрами. При ее отсутствии фоном считается текущее значение команды Brush.

Синтаксис команды Points:

Points (x1)/(y1),(x2)/(y2) ... - перечень координат точек кусочно-линейного контура.

Типы параметров: Points *Point,Point,...*

Допускается команда Points без параметров. В этом случае изображение для соответствующего значения сигнала не выводится.

Синтаксис команды Fon:

Fon [Фон под контуром нулевого сигнала],[Фон под контуром единичного сигнала]

Пример:

Pen 2

Color clLtGray
Points 0/0,20/0,20/10,0/10
Points 0/20,20/20,20/30,0/20
Fon clWhite,clWhite
MultigonTC clRed,clGreen,sw

16.43. MultigonTI - рисует замкнутую закрашенную фигуру с кусочно-линейной границей, положение и цвет которой зависит от состояния флагов телеизмерения. См. П.16.15. “Сигналы телеизмерений” и П.16.16. “Команды обработки телеизмерений”. Флаги телеизмерения определяют номер интервала, по которому выбирается цвет и кусочно-линейный контур рисуемой фигуры. Набор цветов задается командой Colors. Набор кусочно-линейных контуров задается семью командами Points. Цвет и толщина границ контуров задаются командами Color, Pen. Команде MultilineTI может предшествовать команда Fon с семью параметрами. При ее отсутствии фоном считается текущее значение команды Brush.

Синтаксис команды:
MultilineTI [имя сигнала]
Типы параметров: MultilineTI *Ident*

Команде MultilineTI должны предшествовать команды Colors и семь команд Points.
Синтаксис команды Colors:
Colors [цвет контура для первого интервала],[цвет контура для второго интервала],
..., [цвет контура для седьмого интервала]
Типы параметров: Colors *Clr,Clr,...*

Синтаксис команды Points:
Points (x1)/(y1),(x2)/(y2) ... - перечень координат точек.
Типы параметров: Points *Point,Point,...*

Допускается команда Points без параметров. В этом случае изображение для соответствующего значения сигнала не выводится.
Синтаксис команды Fon:
Fon [Фон под контуром первого интервала],[Фон под контуром второго интервала],..., [Фон под контуром седьмого интервала]
Типы параметров: Fon *Clr,Clr,...*

Пример:
Pen 2
Color clLtGray
Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed
Points 0/0,20/0,20/10,0/10
Points 0/20,20/20,20/30,0/30
Points 0/40,20/40,20/50,0/50
Points 0/60,20/60,20/70,0/70
Points 0/80,20/80,20/90,0/90
Points 0/100,20/100,20/110,0/110
Points 0/120,20/120,20/130,0/130
Fon clRed,clRed,clRed,clRed,clRed,clRed,clRed
//Вместо предыдущей команды можно написать: Brush clRed
MultigonTI Uout

16.44. MultiTextTC – рисует текст, положение, состав и цвет которого зависит от состояния телесигнала.

Синтаксис команды:

MultiTextTC [цвет текста нулевого сигнала],[цвет текста единичного сигнала], [имя сигнала]

Типы параметров: RectTC *Clr,Clr,Ident*

Команде MultiTextTC должны предшествовать команды Texts, Fon и Points в каждой по два значения.

Синтаксис команды Texts:

Texts [текст нулевого сигнала],[текст единичного сигнала]

Типы параметров: Texts *String,String*

Текст может состоять из множества строк, разделяемых символами “\n” (без кавычек).

Число строк в тексте – не более 10-и.

Синтаксис команды Points:

Points [Координата текста нулевого сигнала],[Координата текста единичного сигнала]

Типы параметров: Points *Point,Point*

Координата текста определяет положение верхнего левого угла первого символа текста.

Синтаксис команды Fon:

Fon [Фон под текстом нулевого сигнала],[Фон под текстом единичного сигнала]

Типы параметров: Fon *Clr,Clr*

Параметры шрифта задают команды Font и FontStyle.

Пример:

Texts “Фидер включен”,”Фидер выключен”

Points 10/20,10/40

Fon clWhite,clWhite

MultiTextTC clRed,clGreen,sw

16.45. MultiTextTI - рисует текст, положение, состав и цвет которого зависит от состояния флагов телеизмерения.

См. П.16.15. “Сигналы телеизмерений” и П.16.16. “Команды обработки телеизмерений”.

Флаги телеизмерения определяют номер интервала, по которому выбирается положение, состав и цвет рисуемого текста.

Синтаксис команды:

MultiTextTI [имя сигнала]

Типы параметров: MultiTextTI *Ident*

Команде MultiTextTI должны предшествовать команды Texts, Colors, Fon и Points со значениями для семи интервалов сигнала телеизмерения, определенными в таблице TI.

Синтаксис команды Texts:

Texts [текст первого интервала],[текст второго интервала], ...,[текст седьмого интервала]

Типы параметров: Texts *String,String*

Текст может состоять из множества строк, разделяемых символами “\n” (без кавычек).

Число строк в тексте – не более 10-и.

Синтаксис команды Points:

Points [Координата текста первого интервала],[Координата текста второго интервала],
...,[Координата текста седьмого интервала]

Типы параметров: Points *Point,Point*, ...

Координата текста определяет положение верхнего левого угла первого символа текста.

Синтаксис команды Colors:

Colors [цвет текста первого интервала],[цвет текста второго интервала], ...,[цвет текста седьмого интервала]

Типы параметров: Colors *Clr,Clr*, ...

Синтаксис команды Fon:

Fon [Фон под текстом первого интервала],[Фон под текстом второго интервала],..., [Фон под текстом седьмого интервала]

Типы параметров: Fon *Clr,Clr*

Параметры шрифта задают команды Font и FontStyle.

Пример:

Texts "Авария","Предаварийное состояние","Внимание","Норма",+ //перенос строки
" Внимание "," Предаварийное состояние "," Авария "

Points 10/10,10/20,10/30,10/40,10/50,10/60,10/70

Fon clRed,clRed,clRed,clRed,clRed,clRed,clRed

Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed

MultiTextTI U

Пример совместного применения команд MultigonTI и MultiTextTI для вывода переменного текста на фоне, зависящем от состояния телеизмерения.

Base 20,1

{Size 795/475,Origin}

Picture

{

...

SignTI 39_212_30

String "", "", "", "Токи в норме", " Токи в норме ", "Внимание к токам", "Введите АВР 6 кВ!!"

CallCell 157/50, AnyMTextTI, "Превышение суммы токов"

...

}

Library

{

...

Cell AnyMTextTI

{

SignTI ti

String s1,s2,s3,s4,s5,s6,s7

Pen 0

Points 0/0,140/0,140/17,0/17,0/0

Points 0/0,140/0,140/17,0/17,0/0

Points 0/0,140/0,140/17,0/17,0/0

Points 0/0,140/0,140/17,0/17,0/0

```

Points 0/0,140/0,140/17,0/17,0/0
Points 0/0,140/0,140/17,0/17,0/0
Points 0/0,140/0,140/17,0/17,0/0
Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed //цвет закраски полигона
//цвет фона экранной формы для его компенсации, если он белый
Fon clWhite,clWhite,clWhite,clWhite,clWhite,clWhite,clWhite
MultigonTI ti
Pen 1
Texts s1,s2,s3,s4,s5,s6,s7
Points 1/1,1/1,1/1,1/1,1/1,1/1,1/1,1/1
Colors clWhite,clBlack,clBlack,clWhite,clBlack,clBlack,clWhite //цвет текста
//цвет после MultigonTI для его компенсации
Fon clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed
Font 10,Arial
FontStyle fsBold
MultiTextTI ti
}
...
}

```

16.46. Особенности работы команд S...

При исполнении команд SPolylineTC, SPolygonTC, SRectTC, SRoundRectTC, SEllipseTC, STextTC, SButtonTC, SRoundButtonTC цвет рисунка определяет таблица состояний телесигналов. Задается парой предшествующих команд: Signals и States.

Синтаксис команды **Signals**:
 Signals [сигнал 1],[сигнал 2], ...
 Типы параметров: Signals *Ident,Ident,..*

Набор телесигналов выбирается из перечня формальных сигналов ячейки (команда SignTC).

Синтаксис команды **States**:
 States (значение первого сигнала),(значение второго сигнала),...
 Типы параметров: States *Int,Int, ...*

Значение сигнала – число: 0 или 1. Число параметров к команде States должно совпадать с числом параметров в команде Signals.

Команды типа S... могут работать в одном из четырех допустимых режимов. Режим работы определяется обязательным параметром “Режим” – первый слева направо в перечне параметров команды. Параметр “Режим” может принимать одно из следующих значений:

- rgOne – один сигнал;
- rgFirst – первый сигнал;
- rgLast – последний сигнал;
- rgAll – все сигналы.

Цвет определяется командой Colors.
 Синтаксис команды **Colors**:
 Colors [цвет для первого сигнала],[цвет для второго сигнала],...
 Типы параметров: Colors *Clr,Clr, ...*

Число параметров в команде Colors зависит от режима работы команды:

- Для режимов rgFirst или rgLast команда Colors может иметь число параметров равное числу сигналов в команде Signals или на один или два значения больше.
- Для режима rgOne команда Colors может иметь число параметров равное числу сигналов в команде Signals или на один, два или три значения больше.
- Для режима rgAll команда Colors может иметь два параметра или на один больше. Командам SButtonTC, SRoundButtonTC должны предшествовать две команды Colors. Первая - для описания цветов прямоугольника, а вторая – цветов текста. Остальным командам должна предшествовать одна команда Colors.

1) Рассмотрим работу команд в режимах rgOne, rgFirst, rgLast.

При появлении на входе системы сигнала из перечня Signals происходит исполнение команды. Исполнение начинается со сверки текущих значений всех сигналов из Signals с соответствующими значениями в States. По результату сравнения возможны четыре исхода:

1. Текущее состояние ни одного сигнала не соответствует таблице состояний.

Рисуем картинку цветом, заданным в команде Colors на N+1 позиции, где N – число параметров в команде Signals. Если число параметров в команде Colors не больше N, то используется цвет, заданный в команде ColorDisable.

2. Таблице состояний соответствует единственный сигнал.

Картинка рисуется цветом, заданным для данного сигнала в команде Colors.

3. Таблице состояний соответствует несколько сигналов.

Учитывается режим работы команды.

Если задан режим – rgOne, цвет картинки задает параметр из команды Colors, расположенный на N+2 позиции. При ее отсутствии используется цвет, заданный в команде ColorDisable.

Если задан режим – rgFirst, цвет определяет первый сигнал из числа соответствующих таблице состояний.

Если задан режим – rgLast, цвет определяет последний сигнал из числа соответствующих таблице состояний.

4. Хотя бы один сигнал из перечня Signals имеет неопределенное состояние.

4.1. Если задан режим rgOne - рисуем цветом из команды Colors, расположенной на N+3 позиции.

4.2. Если задан режим rgFirst, rgLast - рисуем цветом из команды Colors, расположенной на N+2 позиции.

При отсутствии указанной позиции используется цвет, заданный в команде ColorDisable.

2) Рассмотрим работу команд в режиме rgAll.

При появлении на входе системы сигнала из перечня Signals происходит исполнение команды. Исполнение начинается со сверки текущих значений всех сигналов из Signals с соответствующими значениями в States. По результату сравнения возможны три исхода:

1. Таблице состояний соответствуют все сигналы.

Рисуем картинку цветом, заданным в команде Colors на первой позиции.

2. Таблице состояний не соответствует хотя бы один сигнал.

Картинка рисуется цветом со второй позиции команды Colors.

3. Хотя бы один сигнал из перечня Signals имеет неопределенное состояние.

Рисуем цветом с третьей позиции команды Colors. При отсутствии третьей позиции цвет определяет команда ColorDisable.

Командам *STextTC*, *SButtonTC* и *SRoundButtonTC*, кроме команд *Signals*, *States* и *Colors* должна предшествовать команда *Texts*.

Синтаксис команды **Texts**:

Texts [текст для первого сигнала],[текст для второго сигнала],...

Типы параметров: *Texts String,String,...*

Текст может состоять из множества строк, разделяемых символами “\n” (без кавычек).

Число строк в тексте – не более 10-и.

Число параметров в командах *Texts* и *Colors* должно совпадать. Команды *STextTC*, *SButtonTC* и *SRoundButtonTC* выбирают по описанным выше правилам не только цвет, но и текст из команды *Texts*. Если окажется, что текст на заданной позиции команды *Texts* отсутствует, то выводится символ “?” (без кавычек).

16.47. SPolylineTC - рисует кусочно-линейную кривую, цвет которой определяет таблица состояний телесигналов. Команде *SPolylineTC* должны предшествовать команды *Vid*, *Lines*, *Signals*, *States*, *Colors*, *Points*. Команда *Points* определяет кусочно-линейную кривую.

Синтаксис команды:

SPolylineTC [режим]

Типы параметров: *SPolylineTC Ident*

Параметр “режим” определяет правила работы команды. Принимает значения: *rgOne*, *rgFirst*, *rgLast*, *rgAll*. Подробности смотри в предыдущем примечании.

Неопределенному состоянию соответствует цвет, заданный командой *ColorDisable*.

Если текущее значение поступившего сигнала соответствует заданному в таблице состояний и имеет статус аварии – происходит мигание полилинии до квитирования (чередование цвета полилинии с цветом команды *ColorAvar* один раз в секунду).

Могут предшествовать команды *ColorBlink*, *SignalBlinks* и *StateBlinks*.

Ширина линий предварительно задается командой *Pen*.

Вид *SPolylineTC* определяет команда *Vid*.

1) При задании *Vid 2D* или наличии в файле конфигурации предложения *volume=no* команда *SPolylineTC* рисуется круглой маской, диаметр которой задает команда *Pen*. Углы наклонов составляющих ее отрезков произвольны.

2) При задании *Vid 3D* команда *Polyline* рисуется как объемная фигура с обрезанными концами.

3) При задании *Vid Brd* команда *Polyline* имеет вид рельефного бордюра с обрезанными концами.

В случаях 2) и 3) имеем следующие особенности:

- В файле конфигурации необходимо наличие команды *volume=yes*.

- Команда *SPolylineTC* сглаживает углы между составляющими ее отрезками. Параметры сглаживания задает первый параметр команды *Lines*.

- Составляющие *SPolylineTC* отрезки (кодируются в предшествующей команде *Points*) должны быть параллельны осям координат. При нарушении этого правила рисование происходит в режиме *Vid 2D*.

- Если в *Points* первая и последняя координаты совпадают, то происходит их сглаживание в соответствии с командой *Lines* и имеем замкнутую сглаженную фигуру во всех ее углах.

- Конфигурацию концов полилинии определяют второй и третий параметры команды *Lines*, которые задают глубину проникновения (врезки) начала (конца) полилинии в

другую соприкасающуюся полилинию. При нулевых значениях параметров врезок полилинии имеют ровные окончания.

С целью поддержки задела допускается отключить прорисовку SPolylineTC в режимах Vid 3D и Vid Brd, всегда рисуя в режиме Vid 2D вне зависимости от заданного Vid.

Отключение производится в конфигуураторах графического редактора (Graph.exe) и АРМ (Arm.exe).

Пример.

```
SignTC p1,p2,p3
Signals p1,p2,p3
States 0,1,1
Colors clRed,clYellow,clGreen
Points 10/10,20/10
Pen 2
SPolylineTC rgLast
```

16.48. SPolygonTC – рисует замкнутую закрашенную фигуру с кусочно-линейной границей, цвет закраски фигуры определяет таблица состояний телесигналов. Команде SPolygonTC должны предшествовать команды Signals, States, Colors, Points. Команда Points определяет кусочно-линейную замкнутую кривую. Цвет и толщина обрамляющего контура задаются командами Color и Pen. Могут предшествовать команды ColorBlink, SignalBlinks и StateBlinks.

Синтаксис команды:

SPolygonTC [режим]

Типы параметров: SPolygonTC *Ident*

Параметр “режим” определяет правила работы команды. Принимает значения: rgOne, rgFirst, rgLast, rgAll. Подробности смотри в предыдущем примечании.

Неопределенному состоянию соответствует цвет, заданный командой ColorDisable.

Если текущее значение поступившего сигнала соответствует заданному в таблице состояний и имеет статус аварии – происходит мигание полигона до квитирования (чередование цвета полигона с цветом команды ColorAvar один раз в секунду).

Могут предшествовать команды ColorBlink, SignalBlinks и StateBlinks.

Пример.

```
SignTC p1,p2,p3
Signals p1,p2,p3
States 0,1,1
Colors clRed,clYellow,clGreen
Points 10/10,20/10,20/20,10/20
Pen 2
Color clGray
SPolygonTC rgFirst
```

16.49. SRectTC - рисует закрашенный прямоугольник, цвет которого определяет таблица состояний телесигналов. Команде SRectTC должны предшествовать команды Signals, States, Colors. Цвет и толщина обрамляющего контура задаются командами Color и Pen. Если сигнал имеет статус аварии – происходит мигание до квитирования (чередование цвета прямоугольника с цветом команды ColorAvar один раз в секунду). Могут предшествовать команды ColorBlink, SignalBlinks и StateBlinks.

Синтаксис команды:

SRectTC [режим] ,(x1)/(y1),(x2)/(y2) - координаты прямоугольника.

Типы параметров: SRectTC *Ident,Point,Point*

Параметр “режим” определяет правила работы команды. Принимает значения: rgOne, rgFirst, rgLast, rgAll.

Неопределенному состоянию соответствует цвет, заданный командой ColorDisable.

Пример.

SignTC p1,p2,p3

Signals p1,p2,p3

States 0,1,1

Colors clRed,clYellow,clGreen

Pen 2

Color clGray

SRectTC rgOne,10/10,20/20

16.50. SRoundRectTC - рисует закрашенный прямоугольник со скругленными углами, цвет которого определяет таблица состояний телесигналов. Команде SRoundRectTC должны предшествовать команды Signals, States, Colors. Цвет и толщина обрамляющего контура задаются командами Color и Pen. Если сигнал имеет статус аварии – происходит мигание до квитирования (чередование цвета скругленного прямоугольника с цветом команды ColorAvar один раз в секунду). Могут предшествовать команды ColorBlink, SignalBlinks и StateBlinks.

Синтаксис команды:

SRoundRectTC [режим] ,(x1)/(y1),(x2)/(y2),(x3)/(y3)

(x1)/(y1) - координата верхнего левого угла

(x2)/(y2) - координата правого нижнего угла прямоугольника.

Кривые скругленных углов совпадают с секторами кривой эллипса шириной x3 и высотой y3.

Типы параметров: SRoundRectTC *Ident,Point,Point,Point*

Параметр “режим” определяет правила работы команды. Принимает значения: rgOne, rgFirst, rgLast, rgAll.

Неопределенному состоянию соответствует цвет, заданный командой ColorDisable.

Пример.

SignTC p1,p2,p3

Signals p1,p2,p3

States 0,1,1

Colors clRed,clYellow,clGreen

Pen 2

Color clGray

SRoundRectTC rgOne,10/10,20/20,3/3

16.51. SEllipseTC - рисует закрашенный эллипс, цвет которого определяет таблица состояний телесигналов. Команде SEllipseTC должны предшествовать команды Signals, States, Colors. Цвет и толщина обрамляющего контура задаются командами Color и Pen.

Если сигнал имеет статус аварии – происходит мигание до квитирования (чередование цвета эллипса с цветом команды ColorAvar один раз в секунду). Могут предшествовать команды ColorBlink, SignalBlinks и StateBlinks.

Синтаксис команды:

SEllipseTC [режим] ,(x1)/(y1),(x2)/(y2) - координаты верхней левой и правой нижней точек охватывающего прямоугольника.

Типы параметров: SEllipseTC *Ident,Point,Point*

Параметр “режим” определяет правила работы команды. Принимает значения: rgOne, rgFirst, rgLast, rgAll.

Неопределенному состоянию соответствует цвет, заданный командой ColorDisable.

Эллипс можно нарисовать объемным, если перед командой задать Pen=0 и Vid 3d, а в файл конфигурации внести предложение volume=yes.

Пример.

SignTC p1,p2,p3

Signals p1,p2,p3

States 0,1,1

Colors clRed,clYellow,clGreen

Pen 2

Color clGray

SEllipseTC rgLast,10/10,20/20

16.52. STextTC – рисует текст, состав и цвет которого зависит от таблицы состояний телесигналов. Команде STextTC должны предшествовать команды Signals, States, Colors, Texts, Fon.

Команде STextTC может предшествовать команда Fon с одним значением. При ее отсутствии фоном для выводимого текста считается текущее значение команды Brush.

Параметры шрифта задают команды Font и FontStyle.

Неопределенному состоянию соответствует цвет, заданный командой ColorDisable и текст в виде символа “?” (без кавычек).

Синтаксис команды STextTC:

STextTC [режим] ,(x1)/(y1)

Типы параметров: STextTC *Ident,Point*

Параметр “режим” определяет правила работы команды. Принимает значения: rgOne, rgFirst, rgLast, rgAll.

(x1)/(y1) – координата верхнего левого угла первого символа текста.

Рассмотрим особый случай:

1) Задан режим rgAll.

2) В команде Texts имеем два параметра.

3) Среди описанных в Signals сигналов присутствует хотя бы один неопределенный.

В этом случае рисуется символ “?” (без кавычек). Цвет символа зависит от числа параметров в команде Color. Если число параметров в ней равно трем – цвет определяет третий параметр, иначе, цвет задает команда ColorDisable.

Пример.

Signals p1,p2,p3

States 0,1,1

Colors clRed,clYellow,clGreen

Примечание.

При исполнении команды STextTC текст заносится с цветом фона, заданным командой Fon или Brush, с затиранием старого текста.

16.53. Особенности экранной формы щита и управление миганием.

Механизм мигания на экранной форме (ЭФ) реализован для отображения миганий светодиодов на диспетчерском щите. Не следует путать мигание по аварии с просто миганием. Их реализация различается. Щит состоит из набора светодиодов и панелей отображения значений телеизмерений. Светодиод может иметь зеленый, красный или желтый цвет. Зеленый и красный – основные цвета. Желтый цвет получают при одновременном включении зеленого и красного цветов. Бывают одноцветные светодиоды разной окраски. Имеют единственный основной цвет. Цвет светодиода может находиться во включенном или выключенном состоянии. Каждый цвет может мигать, вне зависимости от того включен он или выключен. Управление щитом выполняется через сервер, который взаимодействует с комплектом специализированных УСО щита. АРМ взаимодействует с сервером, посылая ему телеуправления. Правила посылки телеуправлений в автоматическом режиме описываются в таблицах SendTC и CheckTC. Каждому телеуправлению щита соответствует телесигнал для отражения состояния светодиода. Каждый основной цвет имеет два телеуправления и два телесигнала: включен/выключен и мигает/не мигает.

Для ручного управления щитом необходимо иметь экранную форму, вид которой повторяет реальный вид щита. При необходимости экранная форма может состоять из нескольких листов. Поведение экранной формы строится на основе сигналов щита (прописанных в CheckTC и SendTC). При составлении экранной формы изображения светодиодов с парой основных цветов следует описывать с помощью команд SEllipseTC, а одноцветных - команд EllipseTC языка АРМ-графика. Управление отдельными светодиодами выполняется через команду Mouse, используя действие “Управление через SendTC”. Имеются групповые команды управления светодиодами щита.

Мигание изображений светодиодов предназначено для привлечения внимания к произошедшим значимым событиям. Управление миганием реализуется парой команд: SignalBlinks и StateBlinks, которые воздействуют на следующие за ней команды из перечня: PolylineTC, PolygonTC, RectTC, RoundRectTC, EllipseTC, SPolylineTC, SPolygonTC, SRectTC, SRoundRectTC, SEllipseTC. Цвет мигания определяет команда ColorBlink. Мигание состоит в чередовании цвета мигания с цветом светодиода каждую секунду.

Команда SignalBlinks содержит список телесигналов мигания. Команда StateBlinks определяет при каких значениях каждый телесигнал мигания из команды SignalBlinks находится в мигающем состоянии. Число параметров команды StateBlinks должно совпадать с числом параметров в команде SignalBlinks.

Команды SignalBlinks и StateBlinks перед командами PolylineTC, PolygonTC, RectTC, EllipseTC должны иметь по одному параметру.

Работа команд SignalBlinks и StateBlinks с командами SPolylineTC, SPolygonTC, SRectTC, SRoundRectTC, SEllipseTC зависит от предшествующей им команды Signals. В перечисленных командах задать режим rgOne. Число параметров в командах SignalBlinks и StateBlinks должно равняться числу параметров команды Signals. Команда Signals должна содержать имена телесигналов включения основных цветов. Команда SignalBlinks должна содержать соответствующие им имена телесигналов мигания. В команде States записываются состояния телесигналов из Signals, при которых происходит включение

цвета. Значения включаемых цветов записываются в команде Colors, число параметров у которой равно N+2. В нашем случае цвет на N+2 позиции – clYellow (желтый). В команде StateBlinks записываются состояния телесигналов из SignalBlinks, при которых происходит мигание.

Важно заметить, что созданный механизм мигания можно использовать не только при создании ЭФ щита. Его применяют для привлечения внимания к произошедшим событиям. Например, при изменении состояния выключателя на ЭФ вокруг него можно запрограммировать мигающую рамку. Мигание можно сквитировать (погасить) при нажатии на пункт меню “Квитировать” на ЭФ, если соответствующим образом настроить конфигуратор АРМ.

Имеется несколько пунктов меню “Квитировать”:

- 1) В меню главного окна АРМ.
- 2) В окне “Внимание! Аварии и предупреждения”.
- 3) В каждой ЭФ.

По первым двум пунктам квитируется всё, кроме миганий.

Квитирование по третьему пункту (в ЭФ) настраивается в конфигураторе АРМ. Настройка позволяет квитировать либо всё, кроме миганий, либо только мигания. Если ЭФ имитирует работу щита, то настроить надо на квитирование всего, кроме миганий. В этом случае прекращение миганий управляется щитом для каждого светодиода индивидуально.

Пример мигания см. в П.16.12.2.

16.54. SignalBlinks – команда с перечнем телесигналов мигания.

Команда SignalBlinks используется в подпрограммах. Воздействует на последующие команды из перечня: PolylineTC, PolygonTC, RectTC, RoundRectTC, EllipseTC, SPolylineTC, SPolygonTC, SRectTC, SRoundRectTC, SEllipseTC до новой команды SignalBlinks или до конца текущей подпрограммы. При воздействии на команды SPolylineTC, SPolygonTC, SRectTC, SEllipseTC располагается за командой Signals. Число параметров в команде – не более семи.

Синтаксис команды:

SignalBlinks [имя телесигнала мигания],[имя телесигнала мигания], ...

Типы параметров: SignalBlinks *Ident, Ident, ...*

Имя телесигнала мигания - один из формальных параметров ячейки.

По команде SignalBlinks происходит мигание последующих команд из перечня, если хотя бы один телесигнал мигания в данный момент активен, т.е. текущее значение телесигнала мигания совпадает с заданным для него значением в команде StateBlinks. Мигание по команде SignalBlinks состоит в чередовании текущего цвета команды из перечня с цветом ColorBlink один раз в секунду.

Команда SignalBlinks без параметра отменяет действие предыдущей команды.

Пример:

ColorBlink clBlack

SignalBlinks blink //включает мигание при единичном значении телесигнала blink

StateBlinks 1

Points 10/20,10/40

PolylineTC clRed,clGreen,sw

SignalBlinks //отключает применение SignalBlinks для последующих команд

16.55. StateBlinks – команда со значениями телесигналов мигания, при которых происходит мигание. Располагается за командой SignalBlinks. Число параметров в команде – не более семи.

Синтаксис команды:

StateBlinks (значение телесигнала мигания), (значение телесигнала мигания), ...

Типы параметров: StateBlinks *Int, Int, ...*

Значение телесигнала мигания – 0, 1.

По умолчанию предполагается, что телесигналы мигания включают мигание при единичных значениях. Команда StateBlinks без параметра отменяет действие предыдущей команды, т.е. переходим к единичным значениям.

Пример:

Cell BulbRG

{SignTC r,g,br,bg

Pen 0

Signals r,g

States 1,1

Colors clRed,clGreen, clWhite,clYellow

SignalBlinks br,bg

StateBlinks 1,1 //с единицами можно и не задавать

SEllipseTC rgOne,0/20,5/25

}

16.56. Особенности работы команд Button...

Команды ButtonTC, RoundButtonTC, ButtonTI, RoundButtonTI, SButtonTC и SRoundButtonTC совмещают рисование прямоугольника и текста на нем. Расположением текста управляет команда TrimButton. По умолчанию текст размещается по центру прямоугольника. Цвет прямоугольника и текста зависят от состояния сигналов.

Набор цветов определяет пара команд Colors для прямоугольника и текста соответственно. Цвета не зависят от фона.

Состав текста, также, зависит от состояния сигналов и задается командой Texts.

Тексты в команде Texts могут состоять из множества строк, разделяемых символами “\n” (без кавычек). Число строк в тексте – не более 10-и. Угол поворота текста определяет команда Angle, которая может принимать значения: 0 или 90.

Цвет, вид и толщина обрамляющего контура задаются командами Color, Vid и Pen. Если задать Pen равным нулю, то обрамление будет отсутствовать. На экранных формах при нажатии левой кнопки мыши в областях команд Button, ButtonTC, RoundButtonTC, ButtonTI, RoundButtonTI, SButtonTC и SRoundButtonTC имитируется визуальный эффект утапливания кнопки. В этом случае должна предшествовать команда Vid 3D или Vid Brd.

16.57. ButtonTC- рисует закрашенный прямоугольник и текст на нем. Цвет прямоугольника, а также состав и цвет и текста определяет состояние указанного телесигнала. Команде ButtonTC должна предшествовать команда Texts и две команды Colors.

Первая команда Colors определяет цвета прямоугольника и имеет два параметра для нулевого и единичного значения сигнала. Если сигнал находится в состоянии Disable – прямоугольник рисуется цветом colorDisable.

Вторая команда Colors определяет цвет текста и должна иметь три параметра для нулевого, единичного и неопределенного (Disable) состояния сигнала. В команде Texts, также, должно быть три параметра.

Расположением текста управляет команда TrimButton. По умолчанию текст размещается по центру прямоугольника.

Синтаксис команды:

ButtonTC [имя телесигнала] ,(x1)/(y1),(x2)/(y2) - координаты прямоугольника.

Типы параметров: ButtonTC *Ident,Point,Point*

Пример.

```
colorDisable clLtGray
```

```
Colors clRed ,clGreen //прямоугольник
```

```
Colors clWhite, clWhite,clBlack //Текст
```

```
Texts "One","Two","Disable"
```

```
Pen 2
```

```
Vid 3D
```

```
Color clLtGray
```

```
ButtonTC s1,10/10,70/20 //s1 – формальное имя телесигнала в Cell
```

16.58. RoundButtonTC- рисует закрашенный скругленный прямоугольник и текст на нем. Цвет прямоугольника, а также состав и цвет и текста определяет состояние указанного телесигнала. Команде RoundButtonTC должна предшествовать команда Texts и две команды Colors.

Первая команда Colors определяет цвета прямоугольника и имеет два параметра для нулевого и единичного значения сигнала. Если сигнал находится в состоянии Disable – прямоугольник рисуется цветом colorDisable.

Вторая команда Colors определяет цвет текста и должна иметь три параметра для нулевого, единичного и неопределенного (Disable) состояния сигнала. В команде Texts, также, должно быть три параметра.

Расположением текста управляет команда TrimButton. По умолчанию текст размещается по центру прямоугольника.

Синтаксис команды:

RoundButtonTC [имя телесигнала] ,(x1)/(y1),(x2)/(y2),(x3)/(y3)

Где: (x1)/(y1) – координата верхнего левого угла, (x2)/(y2) – координата нижнего правого угла. Кривая скругленных углов совпадает с кривой эллипса с шириной x3 и высотой y3.

Типы параметров: RoundButtonTC *Ident,Point,Point,Point*

Пример.

```
colorDisable clLtGray
```

```
Colors clRed ,clGreen //прямоугольник
```

```
Colors clWhite, clWhite,clBlack //Текст
```

```
Texts "One","Two","Disable"
```

```
Pen 2
```

```
Vid 3D
```

```
Color clLtGray
```

```
RoundButtonTC s1,10/10,70/20,10/10 //s1 – формальное имя телесигнала в Cell
```

16.59. ButtonTI- рисует закрашенный прямоугольник и текст на нем. Цвет прямоугольника, а также состав и цвет и текста определяет состояние указанного телеизмерения. Команде ButtonTI должна предшествовать команда Texts и две команды Colors. В командах Colors каждому из семи интервалов задается цвет отображения состояния сигнала.

Первая команда Colors определяет цвета прямоугольника и имеет семь параметров. Если сигнал находится в состоянии Disable – прямоугольник рисуется цветом colorDisable.

Вторая команда Colors определяет цвета текста и должна иметь восемь параметров для семи интервалов телеизмерения и неопределенного (Disable) состояния сигнала. В команде Texts, также, должно быть восемь параметров.

Расположением текста управляет команда TrimButton. По умолчанию текст размещается по центру прямоугольника.

Синтаксис команды:

ButtonTI [имя телеизмерения] ,(x1)/(y1),(x2)/(y2) - координаты прямоугольника.

Типы параметров: ButtonTI *Ident,Point,Point*

Пример.

```
colorDisable clLtGray
```

```
Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed //прямоугольник
```

```
Colors clWhite,clBlack,clBlack,clWhite,clBlack,clBlack,clWhite,clDkGray //Текст
```

```
Texts "ButTI0","ButTI1","ButTI2","ButTI3","ButTI4","ButTI5","ButTI6","ButTID"
```

```
Pen 2
```

```
Vid 3D
```

```
Color clLtGray
```

```
ButtonTI t1,10/10,70/20 //t1 – формальное имя сигнала телеизмерения в Cell
```

16.60. RoundButtonTI- рисует закрашенный скругленный прямоугольник и текст на нем. Цвет прямоугольника, а также состав и цвет и текста определяет состояние указанного телеизмерения. Команде ButtonTI должна предшествовать команда Texts и две команды Colors. В командах Colors каждому из семи интервалов задается цвет отображения состояния сигнала.

Первая команда Colors определяет цвета прямоугольника и имеет семь параметров. Если сигнал находится в состоянии Disable – прямоугольник рисуется цветом colorDisable.

Вторая команда Colors определяет цвета текста и должна иметь восемь параметров для семи интервалов телеизмерения и неопределенного (Disable) состояния сигнала. В команде Texts, также, должно быть восемь параметров.

Расположением текста управляет команда TrimButton. По умолчанию текст размещается по центру прямоугольника.

Синтаксис команды:

RoundButtonTI [имя телеизмерения] ,(x1)/(y1),(x2)/(y2),(x3)/(y3)

Где: (x1)/(y1) – координата верхнего левого угла, (x2)/(y2) – координата нижнего правого угла. Кривая скругленных углов совпадает с кривой эллипса с шириной x3 и высотой y3.

Типы параметров: RoundButtonTI *Ident,Point,Point,Point*

Пример.

```
colorDisable clLtGray
```

```
Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed //прямоугольник
```

```
Colors clWhite,clBlack,clBlack,clWhite,clBlack,clBlack,clWhite,clDkGray //Текст
```

```
Texts "ButTI0","ButTI1","ButTI2","ButTI3","ButTI4","ButTI5","ButTI6","ButTID"
```

Pen 2
Vid 3D
Color clLtGray
RoundButtonTI t1,10/10,70/20,10/10 //t1 – формальное имя сигнала телеизмерения в Cell

16.61. SButtonTC- рисует закрашенный прямоугольник и текст на нем. Цвет прямоугольника, а также состав и цвет текста определяет таблица состояний телесигналов. Команде SButtonTC должны предшествовать команды Signals, States, Texts и две команды Colors.

Первая команда Colors определяет цвет прямоугольника. См. “Особенности работы команд S...”.

Вторая команда Colors определяет цвет текста. Число параметров в команде Texts, также, должно быть равно числу сигналов в команде Signals или больше. Неопределенному состоянию соответствует цвет прямоугольника, заданный командой Brush, а цвет текста определяет команда ColorDisable. Текстом является символ “?” (без кавычек).

Расположением текста управляет команда TrimButton. По умолчанию текст размещается по центру прямоугольника.

Синтаксис команды:

SButtonTC [режим] ,(x1)/(y1),(x2)/(y2) - координаты прямоугольника.

Типы параметров: SButtonTC *Ident,Point,Point*

Параметр “режим” определяет правила работы команды. Принимает значения: rgOne, rgFirst, rgLast, rgAll.

Рассмотрим особый случай:

- 1) Задан режим rgAll.
- 2) В команде Texts имеем два параметра.
- 3) Среди описанных в Signals сигналов присутствует хотя бы один неопределенный. В этом случае рисуется символ “?” (без кавычек). Цвет символа зависит от числа параметров во второй команде Color. Если число параметров в ней равно трем – цвет определяет третий параметр, иначе, цвет задает команда ColorDisable. В этом же случае прямоугольник закрашивается в зависимости от числа параметров в первой команде Color. Если число параметров в ней равно трем – цвет определяет третий параметр, иначе, цвет задает команда Brush.

Пример.

```
SignTC p1,p2,p3
Signals p1,p2,p3
States 0,1,1
Colors clRed,clYellow,clGreen
Colors clWhite,clBlack,clWhite,clBlack,clBlack
Texts “One”,”Two”,”Three”,”None”,”Disable”
Pen 2
Vid 3D
Color clGray
SButtonTC rgOne,10/10,70/20
```

16.62. SRoundButtonTC- рисует закрашенный скругленный прямоугольник и текст на нем. Цвет прямоугольника, а также состав и цвет и текста определяет таблица состояний

телесигналов. Команде `SRoundButtonTC` должны предшествовать команды `Signals`, `States`, `Texts` и две команды `Colors`.

Первая команда `Colors` определяет цвет прямоугольника. См. “Особенности работы команд `S...`”.

Вторая команда `Colors` определяет цвет текста. Число параметров в команде `Texts`, также, должно быть равно числу сигналов в команде `Signals` или больше.

Неопределенному состоянию соответствует цвет прямоугольника, заданный командой `Brush`, а цвет текста определяет команда `ColorDisable`. Текстом является символ “?” (без кавычек).

Расположением текста управляет команда `TrimButton`. По умолчанию текст размещается по центру прямоугольника.

Синтаксис команды:

```
SRoundButtonTC [режим] ,(x1)/(y1),(x2)/(y2),(x3)/(y3)
```

Где: $(x1)/(y1)$ – координата верхнего левого угла, $(x2)/(y2)$ – координата нижнего правого угла. Кривая скругленных углов совпадает с кривой эллипса с шириной $x3$ и высотой $y3$.

Типы параметров: `SRoundButtonTC Ident,Point,Point,Point`

Параметр “режим” определяет правила работы команды. Принимает значения: `rgOne`, `rgFirst`, `rgLast`, `rgAll`.

Рассмотрим особый случай:

1) Задан режим `rgAll`.

2) В команде `Texts` имеем два параметра.

3) Среди описанных в `Signals` сигналов присутствует хотя бы один неопределенный.

В этом случае рисуется символ “?” (без кавычек). Цвет символа зависит от числа параметров во второй команде `Color`. Если число параметров в ней равно трем – цвет определяет третий параметр, иначе, цвет задает команда `ColorDisable`.

В этом же случае прямоугольник закрашивается в зависимости от числа параметров в первой команде `Color`. Если число параметров в ней равно трем – цвет определяет третий параметр, иначе, цвет задает команда `Brush`.

Пример.

```
SignTC p1,p2,p3
```

```
Signals p1,p2,p3
```

```
States 0,1,1
```

```
Colors clRed,clYellow,clGreen
```

```
Colors clWhite,clBlack,clWhite,clBlack,clBlack
```

```
Texts “One”,”Two”,”Three”,”None”,”Disable”
```

```
Pen 2
```

```
Vid 3D
```

```
Color clGray
```

```
SRoundButtonTC rgOne,10/10,70/20,10/10
```

Пример описания раздела `Library`

```
Library
```

```
{Cell FD5 // тип библиотечной ячейки (имя для ссылок)
```

```
{SignTC sw,in,out
```

```
SignTI level,I,U
```

```
Menu 1,2,3 //Перечень пунктов контекстного меню для правого клика
```

```

Mouse 5/5,50/50,2 // координаты углов области клика и реакция на левый клик
Attr "Фидер. ФД5"
Attr "Предельная нагрузка – 5 А" // м.б. несколько атрибутов
Pen 2 // Ширина всех последующих примитивов черчения до новой команды Pen
Color clBlue // Цвет всех последующих примитивов черчения до новой команды Color
Points 20/40,100/40
Polyline
Ellipse 50/50,20,20 // координаты охватывающего прямоугольника
Signals U,I
Brush clLtGray
Color clGray
Texts "Ввод 1 ", "U", "I"
Pen 2
Format 8,2
TableTI 1,2,clBlack,10/20,100/40
Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed
LevelTI U,10/20,10/40
// цвет по нулю, цвет по единице, сигнал, местоположение
RectTC clRed,clGreen,sw,10/20,10/40
GraphTI 50,clRed,U,10/20,100/40 // Вывод графика
}
Cell ZD // еще один тип
...
}

```

16.63. Сигналы “Константа”

Поступающие от сервера на вход АРМ константы не содержат информацию об их состоянии. Константы описываются в таблице Const БД настроек АРМ.

Все команды вида xxxxCN отображают значения констант. Имя константы задается в команде ValueCN в качестве одного из параметров или перечисляются в команде Signals перед командой TableCN. Именем сигнала может быть одно из формальных имен, описанное в предложении SignCN ячейки. В процессе работы окна графики формальные имена сигналов ячеек заменяются фактическими.

16.64. TableCN – рисует таблицу текущих значений констант с именами столбцов и строк. Столбцы и строки образуют набор клеток таблицы. Каждой клетке соответствует свой сигнал, значение которого выводится в данной клетке. Слева от рядов клеток таблицы располагаются имена строк. Над столбцами клеток располагаются имена столбцов. Набор сигналов констант выбирается из перечня формальных сигналов ячейки (команда SignCN) и включается в перечень параметров команды **Signals**. Команда Signals должна предшествовать команде TableCN.

Синтаксис команды:

TableCN (число рядов),(число столбцов) ,[цвет текста],[x1)/(y1),(x2)/(y2)

Типы параметров: TableTI *Int,Int,Clr,Point,Point*

Где: (x1)/(y1),(x2)/(y2) - координаты прямоугольника для размещения таблицы. Команде TableCN должны предшествовать команды Signals и Texts. Предварительно заданная команда Vid определяет вид контура и клеток. Команда Format задает правила вывода значений сигналов в клетках таблицы.

Синтаксис команды **Signals**:

Signals [сигнал 1],[сигнал 2], ...

Типы параметров: Signals *Ident,Ident*,...

Число сигналов= число рядов*число столбцов.

Перечисляются сигналы первого ряда клеток таблицы, затем второго и т.д.

Напоминаем, что для записи большого числа параметров следует использовать признак переноса (символ +), после которого продолжайте ввод с новой строки.

Синтаксис команды **Texts**:

Texts <текст для первого ряда>,< текст для второго ряда >, ...

,<текст первой колонки>,<текст второй колонки>,...

Типы параметров: Texts *String,String*,...

Число текстов= число рядов+число столбцов

Если число текстов равно числу рядов, то колонки не имеют текстовых пояснений.

Если в команде Texts все имена строк пустые тексты вида “”, то столбец с именами строк не выводится.

Цвет фона таблицы задается предшествующей командой Brush. Ширина и цвет контура и разграничительных линий предварительно определяются командами Pen, Color. Команда Format определяет формат выводимых значений и подгонку имен строк и столбцов (влево, по центру, вправо). Вид таблицы определяет команда Vid. При задании команды Vid Brd рекомендуется задать команде Pen число не менее 3.

Если необходимо иметь в таблице незаполненные клетки, то при вызове CallCell задайте среди фактических параметров в команде SignCN слова “null” для соответствующих сигналов.

Команда Columns настраивает столбцы таблиц на вывод имен, времен измерений или значений сигналов. При ее отсутствии выводятся значения сигналов. Позволяет задать ширины столбцов.

Синтаксис команды **Columns**:

Columns [тип информации в столбце 1][ширина столбца 1],[тип информации столбце 2] [ширина столбца 2]...

Типы параметров: Columns *Ident[Int],Ident[Int]*,...

Тип информации в столбце – обязательный член параметра (тип - Ident). Он может быть одним из следующих идентификаторов:

- D или Date - выводит даты измерения сигналов
- T или Time - выводит времена измерения сигналов
- DT или DateTime - выводит даты и времена измерения сигналов
- V или Value – выводит значения сигналов
- N или Name - выводит имена сигналов, заданные в таблице сигналов БД настроек АРМ (ТС, TI или Const).

Ширина столбца - необязательный член параметра (тип - Int). Является целым числом.

При его отсутствии производится автоматический расчет ширины столбца на основании выводимой информации.

Команда Columns без параметров отключает действие предыдущей команды Columns.

Пример, в котором первая ячейка таблицы остается незаполненной:

```

Picture
{
...
SignTI null,1_2_3
CallCell 100/200,Tab,"Таблица"
...
}
Library
{
...
Cell Tab
{SignCN C1,C2
Signals C1,C2
Texts "Ввод 1 ","Уст. Т","Уст. Р"
Brush clLtGray
Color clGray
Pen 2
Format 8,2,trR //подгонка текстов по правым краям клеток
TableCN 1,2,clBlack,10/20,100/40
}
...
}

```

16.65. ValueCN – выводит значение сигнала константы в виде текста.

Синтаксис команды:

ValueCN [имя сигнала],(x1)/(y1)

Типы параметров: ValueCN *Ident,Point*

Команде ValueCN должна предшествовать команда Color, которая задает цвет выводимого текста. Фоном для выводимого текста считается текущее значение команды Brush. Предшествующая команда Format определяет форматирование выводимого значения.

Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Пример:

Color clRed

Brush clWhite

Format 10,3

ValueCN C,5/10 //C – формальный параметр

16.66. DateTC – выводит дату измерения телесигнала в виде текста.

Синтаксис команды:

DateTC [имя сигнала],(x1)/(y1)

Типы параметров: DateTC *Ident,Point*

Команде DateTC должна предшествовать команда Color, которая задает цвет выводимого текста. Фоном для выводимого текста считается текущее значение команды Brush.

Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Пример:
Color clRed
Brush clWhite
DateTC C,5/10 //C – формальный параметр

Пример результата работы команды DateTC: 20.09.2011

16.67. TimeTC – выводит время измерения телесигнала в виде текста.

Синтаксис команды:
TimeTC [имя сигнала],[x1)/(y1)
Типы параметров: TimeTC *Ident,Point*

Команде TimeTC должна предшествовать команда Color, которая задает цвет выводимого текста. Фоном для выводимого текста считается текущее значение команды Brush. Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Пример:
Color clRed
Brush clWhite
TimeTC C,5/10 //C – формальный параметр

Пример результата работы команды TimeTC: 9:50:32

16.68. DateTimeTC – выводит дату и время измерения телесигнала в виде текста.

Синтаксис команды:
DateTimeTC [имя сигнала],[x1)/(y1)
Типы параметров: DateTimeTC *Ident,Point*

Команде DateTimeTC должна предшествовать команда Color, которая задает цвет выводимого текста. Фоном для выводимого текста считается текущее значение команды Brush. Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Пример:
Color clRed
Brush clWhite
DateTimeTC C,5/10 //C – формальный параметр

Пример результата работы команды DateTimeTC: 20.09.2011 9:50:32

16.69. DateTI – выводит дату измерения телеизмерения в виде текста.

Синтаксис команды:
DateTI [имя сигнала],[x1)/(y1)
Типы параметров: DateTI *Ident,Point*

Команде DateTI должна предшествовать команда Color, которая задает цвет выводимого текста. Фоном для выводимого текста считается текущее значение команды Brush.

Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Пример:

Color clRed

Brush clWhite

DateTI U,5/10 //U – формальный параметр

16.70. TimeTI – выводит время измерения телеизмерения в виде текста.

Синтаксис команды:

TimeTI [имя сигнала],[x1]/(y1)

Типы параметров: TimeTI *Ident,Point*

Команде TimeTI должна предшествовать команда Color, которая задает цвет выводимого текста. Фоном для выводимого текста считается текущее значение команды Brush. Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Пример:

Color clRed

Brush clWhite

TimeTI C,5/10 //C – формальный параметр

16.71. DateTimeTI – выводит дату и время измерения телеизмерения в виде текста.

Синтаксис команды:

DateTimeTI [имя сигнала],[x1]/(y1)

Типы параметров: DateTimeTI *Ident,Point*

Команде DateTimeTI должна предшествовать команда Color, которая задает цвет выводимого текста. Фоном для выводимого текста считается текущее значение команды Brush. Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Пример:

Color clRed

Brush clWhite

DateTimeTI C,5/10 //C – формальный параметр

16.72. DateCN – выводит дату измерения константы в виде текста.

Синтаксис команды:

DateCN [имя сигнала],[x1]/(y1)

Типы параметров: DateCN *Ident,Point*

Команде DateCN должна предшествовать команда Color, которая задает цвет выводимого текста. Фоном для выводимого текста считается текущее значение команды Brush. Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Пример:

Color clRed
Brush clWhite
DateCN U,5/10 //U – формальный параметр

16.73. TimeCN – выводит время измерения константы в виде текста.

Синтаксис команды:
TimeCN [имя сигнала],[x1]/(y1)
Типы параметров: TimeCN *Ident,Point*

Команде TimeCN должна предшествовать команда Color, которая задает цвет выводимого текста. Фоном для выводимого текста считается текущее значение команды Brush. Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Пример:
Color clRed
Brush clWhite
TimeCN C,5/10 //C – формальный параметр

16.74. DateTimeCN – выводит дату и время измерения константы в виде текста.

Синтаксис команды:
DateTimeCN [имя сигнала],[x1]/(y1)
Типы параметров: DateTimeCN *Ident,Point*

Команде DateTimeCN должна предшествовать команда Color, которая задает цвет выводимого текста. Фоном для выводимого текста считается текущее значение команды Brush. Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Пример:
Color clRed
Brush clWhite
DateTimeCN C,5/10 //C – формальный параметр

16.75. Columns – настраивает колонки таблиц на вывод имени, времени измерения, единиц измерения или значения сигнала. При ее отсутствии выводятся значения сигналов. Позволяет задать ширины столбцов или отказаться от раскраски строк столбцов по состоянию сигналов.

Команда Columns может быть применена перед командами TableTC, TableTI, TableCN. Число параметров в команде должно равняться числу столбцов таблицы, перед командой вывода которой расположена команда Columns. Параметры команды Columns являются трехчленными. Первый член необязательный. Задает способ вычисления цвета фона клетки и цвета текста в ней. Второй член обязательный и определяет тип информации, выводимой в строках столбца. Третий член задает ширину столбца и не является обязательным. Напомним, что перечень сигналов таблицы определяет команда Signals.

Синтаксис команды:
Columns [знак][тип информации в столбце 1][ширина столбца 1], [знак][тип информации столбце 2] [ширина столбца 2]...
Типы параметров: Columns [*Ident*]*Ident*[*Int*], [*Ident*]*Ident*[*Int*],...

Знак – необязательный параметр (тип - Ident). Может являться с одним из следующих символов:

1) Знак отсутствует или команда Columns отсутствует. Текст в клетке имеет цвет, заданный третьим параметром при написании команды вывода таблицы. Клетка столбца окрашивается в цвет ColorDisabled, если сигнал имеет признак Disabled. Иначе, выбор цвета зависит от команды вывода таблицы.

Команда TableTC. Цвет выбирается из предшествующей команды Colors в зависимости от значения телесигнала, приписанного данной клетке. Если значение равно 0 – выбирается первый параметр в команде Colors, и второй параметр при единичном значении телесигнала.

Команда TableTI. Цвет выбирается из предшествующей команды Fon в зависимости от номера интервала. Номер интервала вычисляется по состоянию флагов приписанного клетке телеизмерения в соответствии с Таб. 2 “Правила вычисления номера интервала телеизмерений”.

2) ‘-‘ (минус). Означает отказ от раскраски текста и фона клетки по состоянию сигналов. Их цвета остаются исходными.

3) ‘+’ (плюс). Цвет текста определяет предшествующая команда Colors по номеру интервала. Интервал вычисляется по состоянию сигнала (телеизмерение) или его значению (телесигнал). Цвет клетки не меняет свою исходную окраску, т.е. имеет цвет, заданный командой Brush.

4) ‘*’ (звезда). Применяется только перед командой TableTI. Цвет текста и фона клетки определяют предшествующие команды Colors и Fon по номеру интервала. Интервал вычисляется по состоянию флагов телеизмерения, приписанного данной клетке. Перед командой TableCN задание знака не имеет смысла, т.к. сигналы констант не имеют состояний.

Тип информации в столбце – обязательный параметр (тип Ident). Является одним из следующих идентификаторов:

- D или Date - выводит даты измерения сигналов
- T или Time - выводит времена измерения сигналов
- DT или DateTime - выводит даты и времена измерения сигналов
- V или Value – выводит значения сигналов
- N или Name - выводит имена сигналов, заданные в таблице сигналов БД настроек АРМ (ТС, TI или Const).
- U или Unit - выводит единицы измерения значений сигналов, заданные в таблице телеизмерений БД настроек АРМ. Используется только перед командой TableTI.

Ширина столбца - необязательный член параметра (тип - Int). Является целым числом. При его отсутствии производится автоматический расчет ширины столбца на основании выводимой информации.

Команда Columns без параметров отключает действие предыдущей команды Columns. Если заданная ширина таблицы не равна расчетной, то ширины столбцов пропорционально изменяются (уменьшаются или увеличиваются). Изменению подлежат только столбцы, которым не заданы ширины столбцов. При задании ширин столбцов следует учесть, что при наличии команды Vid Brd размер таблицы уменьшается на две ширины бордюра. Рекомендуется последнему столбцу не задавать в Columns ширину столбца. Тогда он рассчитается как остаток от ширины таблицы и заданных ширин предыдущих столбцов.

Пример:

```

...
Picture
{
...
SignTI !_1_20,!_1_21
CallCell 10/100,TabTI,"Таблица TI"
...
}
Library
{
...
Cell TabTI
{
SignTI p11,p12
Brush clLtGray
Color clLtGray
Fon clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed
// Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed //для +
Colors clWhite,clBlack,clBlack,clWhite,clBlack,clBlack,clWhite //для *
Format 8,2,trR
Pen 6
Font 8,Tahoma
FontStyle
Signals p11,p11,p11,p11,p12,p12,p12,p12
Texts "", "", "Имя TI", "Время измерения", "Значение", "Ед. изм."
Columns -n270,-dt120,*v60,-u
Vid Brd
TableTI 2,4,clBlack,0/0,510/72
}
...
}

```

Вид на экранной форме:

Имя TI	Время измерения	Значение	Ед. изм
ПС Пакл. 110кВ Пан. 4РЛ1 Лукьяв-1 Актив.мощн.	23.10.2015 13:58:29	1062,00	кВт
ПС Пакл. 110кВ Пан. 4РЛ1 Лукьяв-1 Реактивн.мощ	23.10.2015 13:58:29	108,00	кВар

16.76. ButtonTCTI

В команде ButtonTCTI предполагается, что телеизмерение содержит целочисленное значение в заданном диапазоне, начинающимся с нуля, и указывает – какой по номеру телесигнал из списка Signals отвечает за прорисовку кнопки.

Команда ButtonTCTI рисует закрашенный прямоугольник и текст на нем. Цвет прямоугольника, а также состав и цвет текста определяет состояние выбранного по телеизмерению телесигнала в команде Signals. Команде ButtonTCTI должны предшествовать две команды Colors и две команды Texts.

Первая команда Colors определяет цвет прямоугольника и имеет два параметра для нулевого и единичного значения выбранного телесигнала. Если телесигнал находится в состоянии Disable – прямоугольник рисуется цветом colorDisable.

Вторая команда Colors определяет цвет текста. Должна иметь три параметра для нулевого, единичного и неопределенного (Disable) состояний выбранного телесигнала.

В командах Texts число параметров должно равняться числу телесигналов в команде Signals. Первая команда Texts определяет тексты для нулевых значений телесигналов, а вторая – для единичных значений телесигналов. Каждая текстовая строка в команде Текст может состоять из множества подстрок, разделяемых символами “\n” (без кавычек). Число подстрок – не более 10-и. Каждая подстрока выводится на кнопке с новой строки.

Команды Pen, Color и Vid определяют контур кнопки.

Параметры шрифта задают команды Font и FontStyle. Угол поворота текста предварительно задается командой Angle.

Расположением текста управляет команда TrimButton. По умолчанию текст размещается по центру кнопки.

Синтаксис команды:

ButtonTCTI [имя телеизмерения] ,(x1)/(y1),(x2)/(y2) - координаты прямоугольника.

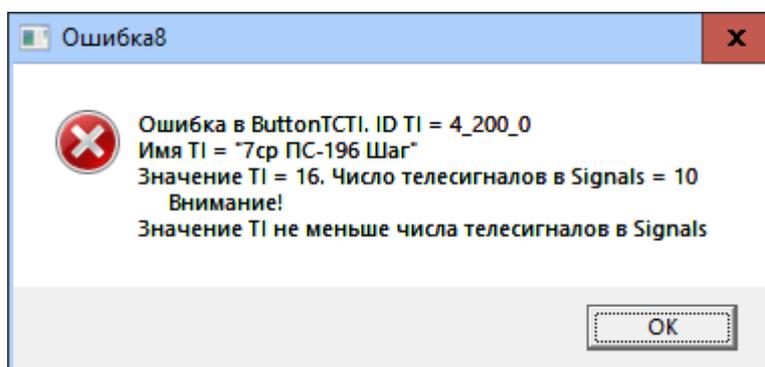
Типы параметров: ButtonTCTI *Ident,Point,Point*

Пример.

```
Cell Btn
{
  SignTI i
  SignTC s0,s1,s2,s3,s4,s5,s6
  Signals s0,s1,s2,s3,s4,s5,s6
  ColorDisable clLtGray
  Colors clRed ,clGreen //цвет прямоугольника
  Colors clWhite, clWhite,clBlack //цвет текста
  Texts "Good0","Good1","Good2","Good3","Good4","Good5","Good6" //0
  Texts "Err0","Err1","Err2","Err3","Err4","Err5","Err6" //1
  Pen 3
  Vid 3D
  Color clLtGray
  ButtonTCTI i,0/0,60/20 //i – формальное имя телеизмерения в Cell
}
```

Внимание!

В процессе работы экранной формы, содержащей команды ButtonTCTI и RoundButtonTCTI, по ошибке составителя прошивки контроллера возможно появление окна следующего содержания:



В котором сообщается, что значение телеизмерения превышает число текстов в команде Texts. Чтобы не загружать экран, появление таких сообщений ограничено числом 10. Последующие ошибки подобного рода игнорируются.

16.77. RoundButtonTCTI

В команде RoundButtonTCTI предполагается, что телеизмерение содержит целочисленное значение в заданном диапазоне, начинающимся с нуля, и указывает – какой по номеру телесигнал из списка Signals отвечает за прорисовку скругленной кнопки.

Команда RoundButtonTCTI рисует закрашенный скругленный прямоугольник и текст на нем. Цвет прямоугольника, а также состав и цвет и текста определяет состояние выбранного по телеизмерению телесигнала в команде Signals. Команде RoundButtonTCTI должны предшествовать две команды Colors и две команды Texts.

Первая команда Colors определяет цвет скругленного прямоугольника и имеет два параметра для нулевого и единичного значения выбранного телесигнала. Если телесигнал находится в состоянии Disable – прямоугольник рисуется цветом colorDisable.

Вторая команда Colors определяет цвет текста. Должна иметь три параметра для нулевого, единичного и неопределенного (Disable) состояний выбранного телесигнала.

В командах Texts число параметров должно равняться числу телесигналов в команде Signals. Первая команда Texts определяет тексты для нулевых значений телесигналов, а вторая – для единичных значений телесигналов. Каждая текстовая строка в команде Текст может состоять из множества подстрок, разделяемых символами “\n” (без кавычек). Число подстрок – не более 10-и. Каждая подстрока выводится на кнопке с новой строки.

Команды Pen, Color и Vid определяют контур кнопки.

Расположением текста управляет команда TrimButton. По умолчанию текст размещается по центру кнопки.

Синтаксис команды:

RoundButtonTCTI [имя телеизмерения] ,(x1)/(y1),(x2)/(y2),(x3)/(y3)

Где: (x1)/(y1),(x2)/(y2) - координаты прямоугольника. Кривая скругленных углов совпадает с кривой эллипса с шириной x3 и высотой y3.

Типы параметров: RoundButtonTCTI *Ident,Point,Point,Point*

Пример.

Cell Btn

```
{
SignTI i
SignTC s0,s1,s2,s3,s4,s5,s6
ColorDisable clLtGray
Colors clRed ,clGreen //цвет прямоугольника
Colors clWhite, clWhite,clBlack //цвет текста
Texts "Good0", "Good1", "Good2", "Good3", "Good4", "Good5", "Good6" //0
Texts "Err0", "Err1", "Err2", "Err3", "Err4", "Err5", "Err6" //1
Pen 3
Vid 3D
Color clLtGray
RoundButtonTCTI i,0/0,60/20,10/10 //i – формальное имя телеизмерения в Cell
}
```

17. Раздел Insert.

Раздел начинается с ключевого слова `Insert` без параметров. Имеет открывающую и закрывающую фигурные скобки. Между скобками могут быть заданы команды `SignTC`, `SignTI` и `Chart`. Команды `SignTC` или `SignTI` (фактические сигналы) должны предшествовать команде `Chart`, задавая перечень выводимых на график сигналов.

17.1. `Chart` – прорисовка графиков сигналов.

Синтаксис команды:

`Chart (x1)/(y1),(x2)/(y2), (тип), <Имя окна>`

Где:

- `x1`, `y1`, `x2`, `y2` - координаты верхнего левого и нижнего правого углов прямоугольника расположения окна на экранной форме;

- Тип – число, определяющее тип сигнала. Допускаются: 0 – телесигналы, 1 – телеизмерения.

- Имя окна - текст, заключенный в кавычки. Выводится в верхней части окна с графиками.

Типы параметров: `Chart Point,Point,Int,String`

Каждое окно с графиками состоит из заголовка окна (сверху), оси значений (слева) и оси времени (снизу). Центральная часть окна содержит графики сигналов на разграфленной подложке. Описание настройки окна содержится в одной из секций файла конфигурации `Insert.cfg`. Файл хранится в корне проекта (где расположен файл `Arm.exe`). Число секций в файле конфигурации равно числу окон с графиками во всех разделах `Insert` текущего проекта АРМ. Секция файла конфигурации содержит следующие параметры:

- `GraphFile` – имя файла экранной формы с разделом `Insert`.

- `Chart` - порядковый номер команды `Chart` в разделе `Insert`.

Параметры `GraphFile` и `Chart` предназначены для привязки команды `Chart` из раздела `Insert` к конкретной секции файла. При последовательном описании команд `Chart` из одного раздела `Insert` допускается не повторять параметр `GraphFile`.

- `Axis` – правило вывода осей в окне с графиками: `yes` – выводить; `no` – не выводить.

- `Title` – правило вывода заголовка: `yes` – выводить; `no` – не выводить.

- `Slice` – правило показа сигналов из срезов: `yes` – показывать; `no` – не показывать.

- `Align` – правило спрямления телеизмерений: `yes` – спрямлять; `no` – не спрямлять.

При задании `No` точки телеизмерений соединяются на рисунке прямыми наклонными линиями. При задании `Yes` графики телеизмерений изображаются в виде ступенчатых фигур. Каждая горизонтальная линия фигуры продолжает предыдущее значение телеизмерения до момента изменения его значения.

- `Disable` - правило показа недействительных сигналов: `yes` – показать; `no` – не показывать.

При задании `Yes` участки с недействительными значениями сигналов окрашиваются на графиках в светло серый цвет. Сигнал считается недействительным, если поступил от КП с флагом `Disable`.

- `Color` – цвет периферии окна, на которой рисуются заголовок и оси.

- `ColorTitle` – цвет заголовка.

- `Interval` - интервал наблюдения в сек.

Если с момента запуска АРМ прошло времени меньше заданного интервала наблюдения `Interval`, то прорисовка графиков работает в режиме уплотнения. В этом режиме значение времени слева на оси времени постоянное. При появлении нового сигнала значение времени справа устанавливается равным времени измерения поступившего сигнала. График перестраивается с учетом нового интервала времени.

Если интервал наблюдения превысил время работы АРМ, то прорисовка графиков переходит в режим скользящего окна, в котором интервал времени в окне постоянный и равен значению Interval. При появлении нового сигнала графики перестраиваются с пересчетом оси времени. Последнее значение на оси времени устанавливается равным времени измерения поступившего сигнала.

Пример секции файла Insert.cfg:

```
% Описание BIN-файла
GraphFile=kot; % Имя BIN-файла
Chart=1; % Порядковый номер Chart в разделе Insert
Axis=yes; % Показ осей
Title=yes; % Показ заголовка
Slice=yes; % Включать из срезов
Align=no; % Спрямять TI
Disable=yes; % Показ disable
Color=clLtGray; % Цвет формы
ColorTitle=clBlack; % Цвет заголовка
Interval=1200; % Интервал наблюдения в сек.
```

Настроить окно с графиками можно путем редактирования файла Insert.cfg в текстовом редакторе (кодировка Windows). Более удобный способ настройки состоит в использовании окна “Таблица сигналов и настроек <Имя окна>”, которое можно вызвать в выпадающем меню при щелчке правой кнопкой мыши по окну с графиками на экранной форме.

Графики сигналов имеют различные цвета. На поле с графиками можно выделить окно просмотра, для чего подведите курсор мыши к верхнему левому углу выделяемого окна, нажмите левую кнопку мыши и, удерживая ее, переместите курсор мыши в нижний правый угол выделяемого окна. Отпустите кнопку мыши, после чего произойдет перестройка поля с графиками. Вернуться к исходному изображению можно выполнив противоположные действия. Нажмите кнопку мыши и, удерживая ее, переместите курсор мыши в произвольную точку, расположенную левее и выше исходной. Отпустите кнопку мыши.

18. Анимация.

Под анимацией понимается имитация движения с помощью изменения (и перерисовки) формы объектов или показа последовательных изображений с фазами движения. В АРМ-графика анимация реализуется следующим способом:

1. В таблице ТС вводятся телесигналы-таймеры с уникальными идентификаторами. Существуют два вида таймеров: базовые, подчиненные. Номер объекта идентификатора таймера должен принадлежать одному из объектов таблицы Objects. В поле Flag телесигнала базового таймера вводится строка: TimerXXXX, где XXXX – число. Например: Timer1000, Timer500. Наличие данного флага означает, что телесигнал расчетный и существует только в АРМ. Число задает миллисекунды. Значение телесигнала изменяется программой через заданное число миллисекунд на противоположное (из нуля в единицу и наоборот).

Уточнение поведения подчиненных таймеров производится в таблице SubTimers базы данных настроек АРМ. Каждый базовый таймер может иметь один или более подчиненных таймеров. В процессе работы базовый таймер переключает значения

приписанных ему подчиненных таймеров в соответствии с временными интервалами, заданными в полях Shift, SpanOn и SpanOff.

■ Поле Shift (сдвиг) определяет число тактов от начала включения базового таймера до установки единичного значения подчиненного таймера (первый раз).

■ Поле SpanOn задает длительность нахождения подчиненного таймера в единичном состоянии. Длительность задается как число тактов базового таймера.

■ Поле SpanOff задает длительность нахождения подчиненного таймера в нулевом состоянии. Длительность задается как число тактов базового таймера.

2. Телесигналы-таймеры, как базовые, так и подчиненные используются в специальных командах рисования для изменения вида картинок. Далее приводятся примеры применения таймеров.

Пример 1.

В данном примере в таблицу телесигналов введены три базовых таймера с идентификаторами:

5_300_200 – таймер с периодом в 200 миллисекунд;

5_300_1000 – таймер с периодом в 1 секунду;

5_300_5000 – таймер с периодом в 5 секунд;

Base 9,1,"PC-27"

{Size 984/681,Origin

}

Picture

{

//Рисуем полоску с бегущими квадратами. Сдвиг квадратов через 1 сек.

SignTC 5_300_1000

CallCell 600/430,Multi,""

//Вентилятор с вращающимися лопастями. Сдвиг лопастей через 0.2 сек.

//5_300_5000 телесигнал-таймер включения-выключения вентилятора через 5 сек.

SignTC 5_300_5000,5_300_200

CallCell 610/475,Mill1,""

//Вентилятор с постоянно вращающимися лопастями. Сдвиг лопастей через 0.2 сек.

SignTC 5_300_200

CallCell 645/475,Mill2,""

}

Library

{

Cell Multi

{

SignTC c1

Pen 3

Color clBlack

Points 0/0,60/0,60/10,0/10,0/0

Polyline

Pen 0

RectTC clLime,clGreen,c1,0/0,10/10

RectTC clGreen,clLime,c1,10/0,20/10

RectTC clLime,clGreen,c1,20/0,30/10

RectTC clGreen,clLime,c1,30/0,40/10

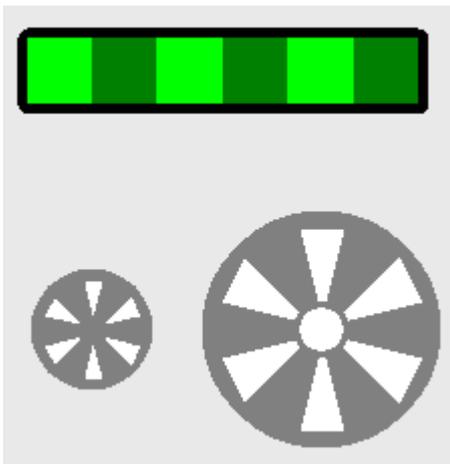
RectTC clLime,clGreen,c1,40/0,50/10

```

RectTC clGreen,clLime,c1,50/0,60/10
}
//Высота вентилятора – 16 пректных единиц.
//sw - включает-выключает вентилятор
//c1 – таймер поворота лопастей
Cell Mill1
{
SignTC sw,c1 //sw=0 – выключен, 1 – включен (лопасти вращаются)
Pen 3
Color clDkGray
Arc -8/-8,8/8,0/8,0/8
Pen 1
Signals sw,c1,c1
States 0,0,1
Colors clDkGray,clWhite,clDkGray
Points 0/0,-2/8,2/8,0/0,6/6,8/2,0/0,8/-2,6/-6,0/0,2/-8,-2/-8,0/0,-6/-6,-8/-2,0/0,-8/2,-6/6
SPolygonTC rgFirst
Colors clWhite,clDkGray,clWhite
Points 0/0,2/8,6/6,0/0,8/2,8/-2,0/0,6/-6,2/-8,0/0,-2/-8,-6/-6,0/0,-8/-2,-8/2,0/0,-6/6,-2/8
SPolygonTC rgFirst
EllipseTC clDkGray,clDkGray,c1,-2/-2,2/2
}
//Высота вентилятора – 32 пректных единицы.
Cell Mill2
{
SignTC c1
Pen 5
Color clDkGray
Arc -16/-16,16/16,0/16,0/16
Pen 1
Points 0/0,-4/16,4/16,0/0,12/12,16/4,0/0,16/-4,12/-12,0/0,4/-16,-4/-16,0/0,-12/-12,-16/-4,0/0,-16/4,-12/12
PolygonTC clWhite,clDkGray,c1
Points 0/0,4/16,12/12,0/0,16/4,16/-4,0/0,12/-12,4/-16,0/0,-4/-16,-12/-12,0/0,-16/-4,-16/4,0/0,-12/12,-4/16
PolygonTC clDkGray,clWhite,c1
EllipseTC clWhite,clWhite,c1,-4/-4,4/4
}
}

```

Вид примера на экранной форме. К сожалению, картинка статична.



Пример 2.

Таблица телесигналов имеет:

- базовый таймер с идентификатором 5_300_200 и периодом в 200 миллисекунд;
- три подчиненных таймера с идентификаторами 5_300_201, 5_300_202, 5_300_203.

В таблице SubTimers базы настроек АРМ подчиненные таймеры описаны следующим образом:

NumAbonB ▾	NomGrB ▾	NomParB ▾	NumAbon ▾	NomGr ▾	NomPar ▾	Shift ▾	SpanOn ▾	SpanOff ▾
1	300	200	1	300	201	0	1	2
1	300	200	1	300	202	1	1	2
1	300	200	1	300	203	2	1	2

В таблице:

- поля NumAbonB, NomGrB, NomParB определяют идентификаторы телесигналов базовых таймеров;
- поля NumAbon, NomGr, NomPar определяют идентификаторы телесигналов подчиненных таймеров.
- описание полей Shift, SpanOn и SpanOff см. выше.

Текст программы:

```

Base 1,3,"test_3"
{Size 630/430,Origin
Tools Arise
}
Picture
{
Vid 2D
SignTC !_300_201,!_300_202,!_300_203
CallCell 60/20,MultiRun,"Run"
SignTC !_300_203,!_300_202,!_300_201 //наоборот
CallCell 60/40,MultiRun,"Run Back"
}
Library
{
Cell MultiRun
{

```

```

SignTC c1,c2,c3
Pen 3
Color clBlack
Points 0/0,150/0,150/10,0/10,0/0
Polyline
Pen 0
RectTC clLime,clGreen,c1,0/0,10/10
RectTC clLime,clGreen,c2,10/0,20/10
RectTC clLime,clGreen,c3,20/0,30/10
RectTC clLime,clGreen,c1,30/0,40/10
RectTC clLime,clGreen,c2,40/0,50/10
RectTC clLime,clGreen,c3,50/0,60/10
RectTC clLime,clGreen,c1,60/0,70/10
RectTC clLime,clGreen,c2,70/0,80/10
RectTC clLime,clGreen,c3,80/0,90/10
RectTC clLime,clGreen,c1,90/0,100/10
RectTC clLime,clGreen,c2,100/0,110/10
RectTC clLime,clGreen,c3,110/0,120/10
RectTC clLime,clGreen,c1,120/0,130/10
RectTC clLime,clGreen,c2,130/0,140/10
RectTC clLime,clGreen,c3,140/0,150/10
}
}

```

Вид примера на экранной форме.



Темные полосы перемещаются слева на право (верхний рисунок) и справа налево (нижний рисунок) с периодом 200 мсек.

19. Вывод текущего времени.

В таблице ТС можно добавить телесигнал с уникальным идентификатором. Номер объекта идентификатора должен принадлежать одному из объектов таблицы Objects. В поле Flag добавленного телесигнала вводится строка: DateTime. Телесигнал является расчетным и существует только в АРМ. Время измерения телесигнала меняется каждую секунду, отражая текущее время на часах компьютера. Используется для вывода текущего времени на экранных формах посредством команд: DateTC, TimeTC, DateTimeTC, TableTC. При использовании команды TableTC необходимо воспользоваться командой Columns с одним из следующих идентификаторов:

- D или Date - выводит даты измерения сигналов
 - T или Time - выводит времена измерения сигналов
 - DT или DateTime - выводит даты и времена измерения сигналов
- Телесигнал, помеченный как DateTime, не записывается в архив.

Пример:

```

Base 3,1,"7cp PY-6кВ КНС-1"
{Size 629/429,Origin

```

```

Ground E9E9E9
}
Picture
{
SignTC 1_100_1000 // Телесигнал, помеченный в поле Flag как DateTime
CallCell 150/160,DT,"DateTime"
}
Library
{
Cell DT
{SignTC c1
Color clRed
Brush clWhite
DateTimeTC c1,0/0

Brush clLtGray
Color clLtGray
ColorDisable clLtGray
Pen 1
Font 8,Tahoma
FontStyle
Signals c1
Colors clRed,clGreen
Texts "", "Текущее время"
Format 8,2,trC
Columns dt
Vid 2d
TableTC 1,1,clBlack,0/20,110/60,tbText
}
}

```

09.04.2019 9:02:41

Текущее время

09.04.2019 9:02:41

Приложение А. Группы команд. Краткое описание команд.

Команды в разделах Picture и Library делятся на группы. Специальные команды и описания используются только в разделе Library. Команда CallCell используется только в разделе Picture. Остальные команды используются в обоих разделах. Команда Chart используется только в разделе Insert.

Группа команд управления:

CallCell - вызов библиотечной ячейки (подпрограммы),
Color - цвет последующих примитивов черчения,
ColorAvar - цвет мигания в аварийном состоянии,
ColorDisable - цвет неопределенного состояния,
ColorBlink – цвет мигания по команде SignalBlinks,
Pen - ширина последующих примитивов черчения,
Brush - цвет закраски замкнутых областей,
Font - шрифт и высота текста,

FontStyle - специальные характеристики текста,
Mouse - реакция при нажатии левой кнопки мыши,
MouseTC – запуск мышью одной из экранных форм по состоянию телесигнала,
Format - формат вывода чисел с плавающей запятой,
Vid - вид изображения,
Scale – масштаб выводимого изображения,
PassTC – условие срабатывания команды Mouse,
Angle – угол наклона текстовых строк,
Style – режим рисования линий в команде Polyline или контуров в командах Polygon и Rect.
Opt – настройка команд LevelTI, MeterTI и ArrowTI,
Trim – определяет подгонку (выравнивание) строк многострочных текстов.
TrimButton – управляет расположением текста на кнопке.

Группа команд описания:

Int - перечень целых чисел (фактические значения),
Float - перечень чисел с плавающей точкой (фактические значения),
Clr - перечень цветов (фактические значения),
Point - перечень координат точек (фактические значения),
String - перечень текстовых строк (фактические значения),
Ident - перечень идентификаторов (фактические значения),
SignTC - перечень телесигналов (фактические сигналы),
SignTI- перечень телеизмерений (фактические сигналы),
SignTY - перечень телеуправлений (фактические сигналы),
SignCN- перечень констант (фактические сигналы),
Menu - контекстное меню для правой кнопки мыши,
Points - перечень координат точек,
Header – заголовок окна вызываемой экранной формы
OutInt – перечень целых чисел для передачи в вызываемую экранную форму,
OutFloat – перечень действительных чисел для передачи в вызываемую экранную форму,
OutClr - перечень цветов для передачи в вызываемую экранную форму,
OutPoint – перечень координат точек для передачи в вызываемую экранную форму,
OutString – перечень текстовых строк для передачи в вызываемую экранную форму,
OutIdent – перечень идентификаторов для передачи в вызываемую экранную форму,
OutSignTC – перечень телесигналов из таблицы TC для передачи в вызываемую форму,
OutSignTI – перечень телеизмерений из таблицы TI для передачи в вызываемую форму,
OutSignTY – перечень телеуправлений из таблицы TY для передачи в вызываемую форму,
OutSignCN – перечень констант из таблицы Const для передачи в вызываемую форму.

Группа команд рисования:

Arc - дуга окружности или эллипса,
Chord - закрашенная фигура, ограниченная дугой окружности или эллипса и хордой,
Pie - закрашенный сектор окружности или эллипса,
Polygon - закрашенная фигура с кусочно-линейной границей,
Polyline _ кусочно-линейная кривая,
Rect - закрашенный прямоугольник,
RoundRect - закрашенный прямоугольник со скругленными углами,
Ellipse - закрашенная окружность или эллипс,
Text - текст,
Image – изображение из файла,
Button - кнопка для команды Mouse.

Группа специальных команд описания:

Int - формальные имена для целых чисел,
Float - формальные имена для чисел с плавающей точкой,
Clr - формальные имена для цветов,
Point - формальные имена для координат точек,
String - формальные имена для текстовых строк,
Ident - формальные имена для идентификаторов,
SignTC - формальные имена для телесигналов,
SignTI - формальные имена для телеизменений,
SignTY - формальные имена для телеуправлений,
SignCN - формальные имена для констант,
Attr – текстовые параметры библиотечной ячейки,
Signals перечень формальных сигналов,
Texts – перечень текстов,
Colors – перечень цветов,
Fon – перечень цветов фона,
FreshTC и FreshTI - имитация сигнала для восстановления затертой части картинки,
SignalBlinks – перечень имен телесигналов мигания,
StateBlinks – перечень значений телесигналов мигания.
Columns – настройка колонок таблиц на вывод параметров сигнала.

Группа специальных команд рисования:

TableTI - таблица текущих значений телеизмерений,
TableTC- таблица текущих значений телесигналов,
LevelTI - полоса уровня, зависящая от значения телеизмерения,
ArrowTI - стрелка в полукруге для показа значений телеизмерения,
MeterTI – имитация стрелочного индикатора для показа значений телеизмерения,
PolylineTC - кусочно-линейная кривая, цвет которой определяет состояние телесигнала,
PolygonTC – полигон, цвет которого определяет состояние телесигнала,
RectTC - прямоугольник, цвет закрашки определяет состояние телесигнала,
RoundRectTC - прямоугольник со скругленными углами, цвет закрашки определяет состояние телесигнала,
EllipseTC - закрашенный эллипс, цвет которого определяет состояние телесигнала,
PolylineTI - кусочно-линейная кривая, цвет которой определяет значение телеизмерения,
PolygonTI – полигон, цвет которого определяет значение телеизмерения,
RectTI - закрашенный прямоугольник, цвет которого определяет значение телеизмерения,
RoundRectTI - прямоугольник со скругленными углами, цвет закрашки определяет значение телеизмерения,
EllipseTI - закрашенный эллипс, цвет которого определяет значение телеизмерения,
GraphTI - график сигналов телеизмерений,
TextTC – текст, состав и цвет которого зависит от состояния телесигнала,
TextTI – текст, состав и цвет которого зависит от значения телеизмерения,
MultilineTI – полилиния, положение и цвет которой зависит от значения телеизмерения,
MultilineTC – полилиния, положение и цвет которой зависит от состояния телесигнала,
MultigonTC – полигон, положение и цвет которого зависит от состояния телесигнала,
MultigonTI – полигон, положение и цвет которого зависит от значения телеизмерения,
ValueTI - значение сигнала телеизмерения в виде текста,
TimeValueTI - значение сигнала телеизмерения в формате времени (чч:мм:сс),
TextValueTI – текст, выбираемый по значению сигнала телеизмерения,
MultiTextTC – текст, положение, состав и цвет которого зависит от состояния телесигнала,
MultiTextTI – текст, положение, состав и цвет которого зависит от значения телеизмерения,

SPolylineTC - кусочно-линейная кривая, цвет которой определяет таблица состояний телесигналов,
 SPolygonTC – полигон, цвет которого определяет таблица состояний телесигналов,
 SRectTC - закрашенный прямоугольник, цвет которого определяет таблица состояний телесигналов,
 SRoundRectTC - прямоугольник со скругленными углами, цвет закрашки определяет таблица состояний телесигналов,
 SEllipseTC - закрашенный эллипс, цвет которого определяет таблица состояний телесигналов,
 STextTC – текст, состав и цвет которого зависит от таблицы состояний телесигналов,
 ButtonTC - прямоугольник и текст, состав и цвет которых определяет состояние телесигнала,
 RoundButtonTC - скругленный прямоугольник и текст, состав и цвет которых определяет состояние телесигнала,
 ButtonTI - текст, состав и цвет которых определяет состояние сигнала телеизмерения,
 RoundButtonTI - скругленный прямоугольник и текст, состав и цвет которых определяет состояние сигнала телеизмерения,
 SButtonTC - прямоугольник и текст, состав и цвет которых определяет таблица состояний телесигналов,
 SRoundButtonTC - скругленный прямоугольник и текст, состав и цвет которых определяет таблица состояний телесигналов,
 TableCN - таблица текущих значений констант,
 ValueCN - значение сигнала константы в виде текста
 DateTC – дата измерения телесигнала в виде текста
 TimeTC – время измерения телесигнала в виде текста
 DateTimeTC – дата и время измерения телесигнала в виде текста
 DateTI – дата измерения телеизмерения в виде текста
 TimeTI – время измерения телеизмерения в виде текста
 DateTimeTI – дата и время измерения телеизмерения в виде текста
 DateCN – дата измерения константы в виде текста
 TimeCN – время измерения константы в виде текста
 DateTimeCN – дата и время измерения константы в виде текста
 ButtonTCTI - прямоугольник и текст, состав и цвет которых определяет состояние телесигнала, выбираемого по значению телеизмерения,
 RoundButtonTCTI - скругленный прямоугольник и текст, состав и цвет которых определяет состояние телесигнала, выбираемого по значению телеизмерения,

Команды внедрения окон:

Chart - описание окна с графиками в разделе Insert

Примечание.

Допускается использовать вместо команды CallCell команду CallSell.

Приложение Б.

Перечень основных значений цветов в командах

<u>Параметр</u>	<u>Значение</u>
clAqua	Голубой
clBlack	Черный
clBlue	Синий
clCream	Кремовый
clDkGray	Темно серый

clFuchsia	Розовый
clGray	Серый
clGreen	Темно зеленый
clLime	Зеленый
clLtGray	Светло серый
clMaroon	Малиновый
clMedGray	Средне серый
clMoneyGreen	Серо зеленый
clNavy	Темно синий
clOlive	Оливковый
clPurple	Пурпурный (фиолетовый)
clRed	Красный
clSilver	Серебристый
clSkyBlue	Небесно голубой
clTeal	Бирюзовый
clWhite	Белый
clYellow	Желтый

Приложение В.

Перечень стилей текстов

<u>Параметр</u>	<u>Значение</u>
fsBold	жирный шрифт
fsItalic	курсив
fsUnderline	подчеркнутый.
fsStrikeOut	зачеркнутый (текст изображается с горизонтальной линией на нем).

Приложение Г.

Перечень ключевых слов и символов.

Имена разделов:

- Base – базовое описание листа;
- Var –перечень переменных;
- Input – входные параметры программы
- Picture – описание картинки;
- Library – библиотека подпрограмм;
- Insert – описание окон с графиками;
- Cell – подпрограмма в разделе Library.

Примечание.

Допускается использовать вместо ключевого слова Cell ключевое слово Sell.

Ключевые слова раздела Base:

- Size - размер картинки;
- Fixed - признак немасштабируемого изображения;
- Simple – упрощенное оформление выводимой на экран формы;
- Origin – вывод картинки без нарушения пропорций.
- MainMenu - определяет видимость меню окна.
- Tools - определяет видимость и поведение панели инструментов (см. Arise).
- StatusBar - определяет видимость строки состояния.

Arise – делает панель инструментов окна экранной формы всплывающей.
Ground – цвет подложки, на которой рисуется экранная форма.
Library - имя файла внешней библиотеки.
Off, On – невидим, видим.
Position - определяет местоположение экранной формы (ЭФ) на мониторе.

Параметры команды Position:

Default, Designed (содержит координату), LeftTop, RightTop, RightDown, LeftDown, ScreenCenter.

Типы переменных:

Int – целое число
Float – число с плавающей точкой
Clr - цвет
Point – координата точки
String – текстовая строка
Ident – идентификатор:
SignTC – телесигнал из таблицы TC
SignTI – телеизмерение из таблицы TI
SignTY – телеуправление из таблицы TY
SignCN – константа из таблицы Const

Перечень команд (Приложение А).

Перечень основных значений цветов (Приложение Б).

Перечень стилей текстов (Приложение В).

Перечень значений параметра команды Vid:

2D – плоские изображения;
3D – выпуклые изображения;
Brd или Border – рельефная граница изображения.

Перечень значений параметров команды Columns:

D или Date - вывод даты измерения сигналов колонки
T или Time - вывод времени измерения сигналов колонки
DT или DateTime - вывод даты и времени измерения сигналов колонки
V или Value – вывод значения сигналов колонки
N или Name - вывод имен сигналов колонки
U или Unit – вывод единиц телеизмерений в колонке команды TableTI

Перечень значений параметра “режим” команд SPolylineTC, SPolygonTC, SRectTC, SEllipseTC, STextTC, SButtonTC:

rgOne – один сигнал;
rgFirst – первый сигнал;
rgLast – последний сигнал;
rgAll – все сигналы.

Перечень значений параметра “Правило подгонки текста” команд Trim, Format:

trR – числовые значения выводимых в таблицу телеизмерений подгоняются по правым краям клеток таблицы;

trC – подгоняются по центрам клеток;

trL - подгоняются по левым краям клеток.

Перечень значений параметра “Тип фигуры” команды TableTC:

tbCirc – круг;

tbRect – прямоугольник;

tbText – имя состояния сигнала (из столбцов: TextNul, TextOne таблицы TC).

Перечень значений параметра “Стиль” команды Style:

psSolid - сплошные линии;

psDash - штриховые линии;

psDot - частые штриховые линии.

Перечень значений параметров команды Opt:

Первый параметр “Состав текста”:

oNo – текст отсутствует.

oPercent – текстом является процентное отношение значения текущего сигнала к максимально допустимому его значению в заданном диапазоне.

oValue - текстом является значение текущего сигнала.

oValueUnit - текстом является значение текущего сигнала с его единицей измерения.

oMiddlePercent – текстом является процентное отношение значения текущего сигнала к максимально допустимому его значению в заданном диапазоне. Текст располагается по высоте в середине сегмента.

oMiddleValue - текстом является значение текущего сигнала. Текст располагается по высоте в середине сегмента.

oMiddleValueUnit - текстом является значение текущего сигнала с его единицей измерения. Текст располагается по высоте в середине сегмента. Текст располагается в середине сегмента.

Второй параметр “Видимость шкалы”:

oScaleOff – шкала выключена, ход стрелки – 90 градусов.

oScaleOn – шкала включена, ход стрелки – 90 градусов.

oRoundScaleOff – шкала выключена, ход стрелки – 270 градусов.

oRoundScaleOn – шкала включена, ход стрелки – 270 градусов.

Третий параметр “Ориентация”:

oAny – уровень ориентируется вдоль длинной стороны прямоугольника, заданного командой LevelTI.

oUp – уровень ориентируется снизу вверх

oRight – уровень ориентируется слева направо

Команда Opt, также, позволяет управлять видом и толщиной стрелки посредством следующих идентификаторов:

oArrowX, где X является числом и задает толщину стрелки. Например: oArrow3.

oOutArrowX, где X является числом и задает толщину стрелки. При отсутствии данной команды стрелка рисуется от центра вращения. При наличии данной команды рисуется

только верхняя часть стрелки, длина которой составляет 20 процентов от полной длины стрелки. Например: oOutArrow1.

Зарезервированные слова при описании фактических параметров в командах SignTC, SignTI.

null - неиспользуемые сигналы

LinkXX – внутренний расчетный сигнал состояния связи с объектом под номером XX.

Перечень специальных символов:

// - начало комментария;

{ - открывающая скобка;

}- закрывающая скобка;

= - присвоение значения переменной;

+ - признак продолжения строки параметров на следующей строке.

Приложение Д.

Примеры программ.

Программы составлены под сигналы БД настроек АРМ Комсомольскнефти. Скопируйте нижеприведенные программы в отдельные файлы с конвертацией в стандартный Windows формат, например, воспользовавшись блокнотом. Используйте программы в Graph.exe как полигон для испытаний. Начните с замены сигналов на прописанные в Вашей БД.

1. Пример использования команд раздела Picture, способы применения переменных, команды отображения уровней, базовые команды визуализации состояний сигналов.

```
Base 1,1,"Первый лист" // номер абонента, номер и имя листа
{Size 630/430,Origin}
Var
{int hFont
 Point pt
 Clr Red,Green
 signTC tc1,tc2
 signTI ti1,ti2
 ident nam
 String t1,t2
}
Picture
{Pen 3
 Red=clRed
 Green=clGreen
 Image 0/0,630/430,"Cn_ob.bmp"
 FontStyle fsBold,fsItalic,fsUnderline
 hFont=10
 Font hFont,Arial
 Color clLime
 Brush 80ff
 Arc 10/10,90/90,90/50,10/50
 Text 40/25,"Arc"
 Chord 110/10,190/90,190/50,110/50
 Text 130/25,"Chord"
 Ellipse 210/10,290/90
 Text 230/25,"Ellipse"
```

```

Pie 310/10,390/90,390/30,310/30
Text 340/25,"Pie"
Brush clBlue
Points 10/110,90/110,10/190,90/190
Polyline
Angle 45
Text 10/150,"Polyline"
Color ffff80
Rect 110/110,190/190
Text 140/125,"Rect"
Points 210/110,290/110,290/190,210/190
Polygon
Text 230/125,"Polygon"
Brush Red
RoundRect 310/110,390/190,30/30
Text 315/125,"RoundRect"
FontStyle //Сброс
t2="Норма"
nam=dns1-1
tc1=1_1_62
tc2=1_1_63
ti1=1_1_20
SignTC tc1
CallCell 410/20,Control,"Управление"
SignTI ti1,1_1_21 // перечень сигналов таблицы TI (фактические параметры)
CallCell 10/210,Level,"Уровень"
SignTC tc1,tc2,Link1
t1="Проба ТС"
pt=0/70
CallCell 10/310,StuffТС,"Проба ТС"
t1="Проба TI раз"
CallCell 170/210,StuffTI,"Проба TI раз"
SignTI 1_1_22,1_1_23
t1="Проба TI два"
pt=10/70
CallCell 400/210,StuffTI,"Проба TI два"
Menu 1,2,3,4,8,9,10
Mouse 0/0,630/430,0
}
Library
{Cell Control
{SignTC cl
Pen 3
Color clLtGray
Menu 1,2,4
Brush clLtGray
Rect 0/0,110/160
Color clGray
//При нажатии соответствующей кнопки:
//1. Перейти к листу по имени, заданному переменной nam
Mouse 10/10,100/30,1,nam
Button 10/10,100/30,clBlack,"На dns1-1"
//2. Выполнить телеуправление, если
// значение сигнала заданного переменной tc1 единица
PassTC tc1,1
Mouse 10/40,100/60,2,3_1_150,1 //ТУ
Button 10/40,100/60,clBlack,"ТУ"
PassTC //Сброс
//3. Изменить значение телесигнала в контроллере
Mouse 10/70,100/90,3,tc1 //ТС
Button 10/70,100/90,clBlack,"ТС"
//4. Изменить значение телеизмерения в контроллере
Mouse 10/100,100/120,4,ti1 //TI
Button 10/100,100/120,clBlack,"TI"

```

```

//5. Выполнить макрокоманду из таб. Macro под номером 1
Mouse 10/130,100/150,5,1 //Macro
Button 10/130,100/150,clBlack,"Macro"

}
Cell Level
{SignTI U,I
Attr "Фидер. ФД5"
Attr "Предельная нагрузка - 5 А" // м.б. несколько атрибутов
Menu 5,6,7,8
Mouse 0/0,100/100,0
Font 8,Arial
FontStyle
Pen 2
Color clLtGray
Colors Red,clYellow,clLime,Green,clLime,clYellow,Red
Brush fffdcc
ArrowTI Red,U,0/0,80/40
LevelTI U,0/60,70/75
LevelTI U,90/0,105/70
LevelTI I,110/0,125/70
}
Cell StuffTC
{SignTC sw0,sw1,Link
Color clBlack
Text pt,t1
Pen 2
Color clLtGray
Points 0/0,30/0,30/40,0/40
RectTC Red,Green,sw0,50/0,80/30
EllipseTC Red,Green,Link,90/0,120/30
ColorAvar clBlue
PolylineTC Red,Green,sw1
Points 5/5,25/5,25/35,5/35
PolygonTC Red,Green,sw0
Font 20,Arial
FontStyle fsBold
Texts "Авария","Норма"
Colors Red,Green
Fon clWhite
TextTC sw0,30/40
ColorAvar clWhite
Points 35/0,45/0
Points 35/0,45/10
Fon clWhite,clWhite
MultilineTC Red,Green,sw0
}
Cell StuffTI
{SignTI U,I
Signals U,I
Color clBlack
Text pt,t1
Colors Red,clYellow,clLime,Green,clLime,clYellow,Red
Pen 2
Color clLtGray
Points 0/0,30/0,30/40,0/40
PolylineTI U
RectTI U,50/0,80/30
EllipseTI U,90/0,120/30
Brush fffdcc
GraphTI 43,clBlue,U,130/0,190/70
Points 5/5,25/5,25/35,5/35
PolygonTI U
Font hFont,Arial

```

```

FontStyle fsBold
Fon clWhite
Texts "Авария", "Предаварийное", "Внимание", t2, "Внимание", +
"Предаварийное", "Авария"
TextTI U, 0/40
Points 35/0, 45/0
Points 35/5, 45/5
Points 35/10, 45/10
Points 35/15, 45/15
Points 35/20, 45/20
Points 35/25, 45/25
Points 35/30, 45/30
Colors Red, clYellow, clLime, Green, clLime, clYellow, Red
Fon clWhite, clWhite, clWhite, clWhite, clWhite, clWhite, clWhite,
MultilineTI U
}
}

```

2. Пример второго листа с командами вывода таблиц, команды Multi..., работа с командами SMulti... и FreshTC .

```

Base 1, 2, "Второй лист" // номер абонента, номер и имя листа
{Size 630/430, Origin} // размер листа
Var
{SignTC tc1, tc2, tc3, tc4, tc5, tc6, tc7, tc8}
Picture
{Pen 3
Vid 3d
Color clLtGray // Цвет всех последующих примитивов
Image 0/0, 630/430, "Cn_ob.bmp"
Menu 1, 2, 4
Mouse 0/0, 100/100, 0
SignTI 1_1_20, 1_1_21, 1_1_22, 1_1_23, 1_1_30, 1_1_36, 1_1_37, 1_1_38, 1_1_39, +
1_1_40, 1_1_41, 1_2_20, 1_2_21, 1_2_22, 1_2_23, 1_2_30, 1_2_36, 1_2_37, 1_2_38, +
1_2_39, 1_2_40, 1_2_41, 1_3_20, 1_3_21, 1_3_22, 1_3_23, 1_3_30, 1_3_36, 1_3_37, +
null, 1_3_39, 1_3_40 //с пустой ячейкой в таб.
CallCell 100/30, TabTI, "Таблица TI"
SignTC 1_-1_62, 1_-1_63, 1_1_70, 1_1_71, 1_1_72, 1_1_73, 1_2_70, 1_2_71, 1_2_72, +
1_2_73, 1_3_70, 1_3_71, 1_3_72, 1_3_73, 1_4_70, 1_4_71, 1_4_72, 1_4_73, 1_5_70, +
1_5_71, 1_5_73, 1_6_70, 1_6_71, 1_7_70, 1_7_71, 1_7_73, 1_7_74, 1_8_70, 1_8_71, +
1_8_73, 1_8_74, 1_9_70
CallCell 160/220, TabTC, "Таблица TC"
SignTC 1_-1_62
CallCell 400/20, StarTC, "Звезда TC"
SignTI 1_1_20
CallCell 400/200, StarTI, "Звезда TI"
SignTC Link1
CallCell 30/30, Lnk, "Связь"
SignTC 1_1_70, 1_1_71
SignTI 1_1_20
CallCell 30/100, L, "LL"
//Проверка FreshTC
tc1=1_1_70
tc2=1_1_71
tc3=1_1_72
tc4=1_1_73
tc5=1_2_70
tc6=1_2_71
tc7=1_2_72
tc8=1_2_73
//Moto затирает SPolygonTC из OverMoto
//для восстановления вставлен FreshTC
SignTC 1_3_70, 1_3_71

```

```

    CallCell 15/150,Moto,"Группа Двигатель"
    SignTC tc1,tc2,tc3,tc4,tc5, tc6,tc7,tc8
    CallCell 15/150,OverMoto,"Группа "
    SignTC tc1,tc2,tc3,tc4
    CallCell 15/220,Uni,"Тест SUniTC"
}
Library
{Cell StarTC
{SignTC p
Points 0/60,56/45,90/0,124/45,180/60,140/100,140/160,90/140,40/160,40/100
PolygonTC clGreen,clRed,p
Points 180/10,200/20,220/10,200/0
Points 180/30,180/50,220/50,220/30
//Fon clWhite,clWhite
MultigonTC clGreen,clRed,p
Points 190/4,190/34
Texts "Nul","One"
Fon clGreen,clRed
MultiTextTC clRed,clGreen,p
}
Cell StarTI
{SignTI p
Points 0/60,56/45,90/0,124/45,180/60,140/100,140/160,90/140,40/160,40/100
Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed
PolygonTI p
Pen 2
Points 180/0,180/15,220/15,220/0
Points 180/10,180/25,220/25,220/10
Points 180/20,180/35,220/35,220/20
Points 180/30,180/45,220/45,220/30
Points 180/40,180/55,220/55,220/40
Points 180/50,180/65,220/65,220/50
Points 180/60,180/75,220/75,220/60
//Fon clWhite,clWhite,clWhite,clWhite,clWhite,clWhite,clWhite,clWhite,
MultigonTI p
Points 190/2,190/12,190/22,190/32,190/42,190/52,190/62
Texts "Null","One","Two","Three","Four","Five","Six"
Colors clBlack,clBlack,clBlack,clBlack,clBlack,clBlack,clBlack
Fon clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed
MultiTextTI p
Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed
Fon clWhite
ValueTI p,0/20
}
Cell TabTC
{SignTC p11,p12,p13,p14,p15,p16,p17,p18,p21,p22,p23,p24,p25,p26,p27,p28,+
p31,p32,p33,p34,p35,p36,p37,p38,p41,p42,p43,p44,p45,p46,p47,p48
Signals p11,p12,p13,p14,p15,p16,p17,p18,p21,p22,p23,p24,p25,p26,p27,p28,+
p31,p32,p33,p34,p35,p36,p37,p38,p41,p42,p43,p44,p45,p46,p47,p48
Pen 1
Color clLtGray
Brush clLtGray
Texts "Линия 1","Линия 2","Линия 3","Линия 4","Линия 5","Линия 6",+
"Линия 7","Линия 8","Связь","Готов","Обрыв","Вызов"
Colors clRed,clBlue
// чис. рядов, колонок, цвет текста, корд.углов
Format 8,2,trC //подгонка текстов по центру
TableTC 8,4,clBlack,0/0,200/163,tbRect //Прямоугольники
}
Cell Lnk
{SignTC Link
EllipseTC clRed,clGreen,Link,-10/-10,10/10
}
Cell TabTI

```

```

{SignTI p11,p12,p13,p14,p15,p16,p17,p18,p21,p22,p23,p24,p25,p26,p27,p28,+
p31,p32,p33,p34,p35,p36,p37,p38,p41,p42,p43,p44,p45,p46,p47,p48
  Signals p11,p12,p13,p14,p15,p16,p17,p18,p21,p22,p23,p24,p25,p26,p27,p28,+
p31,p32,p33,p34,p35,p36,p37,p38,p41,p42,p43,p44,p45,p46,p47,p48
  Menu 5,6,7,8
  Mouse 0/0,100/100,0
  Pen 2
  Color clLtGray
  Brush clLtGray
  Texts "Линия 1","Линия 2","Линия 3","Линия 4","Линия 5","Линия 6",+
"Линия 7","Линия 8","кВар","кВт","U","I"
// чис. рядов, колонок, цвет текста, корд.углов
  Format 8,2,trR //подгонка текстов вправо
  TableTI 8,4,clBlack,0/0,260/163
}
Cell L
{SignTC c1,c2
  SignTI i
  ColorAvar clLtGray
  pen 5
  Points -15/25,15/25,15/-25,-15/-25,-15/25
  PolylineTC clRed,clGreen,c1
  Pen 0
  RectTC clGreen,clRed,c1,-10/-20,10/0
  RectTC clRed,clGreen,c2,-10/0,10/20
  Colors clRed,clYellow,clLime,clGreen,clLime,clYellow,clRed
  ValueTI i,10/30
}
Cell Moto
{SignTC s1,s2
  Signals s1,s2
  States 1,1
  Colors clRed,clWhite,clGray
  Pen 3
  SRectTC rgLast,0/0, 30/30
  Colors clWhite,clWhite,clWhite,clWhite
  Pen 1
  FreshTC tc1,tc2,tc3,tc4,tc5,tc6,tc7,tc8
  SEllipseTC rgLast,2/2,28/28
}
Cell OverMoto
{SignTC s1,s2,s3,s4,s5,s6,s7,s8
  Signals s1,s2,s3,s4,s5,s6,s7,s8
  States 1,1,1,1,1,1,1,1
  Colors
clYellow,clGreen,clGreen,clYellow,clRed,clLtGray,clWhite,clWhite,clGray
  Points 3/15,15/3,27/15
  Pen 1
  Color clBlack
  SPolygonTC rgLast
}
Cell Uni
{SignTC p1,p2,p3,p4
  Signals p1,p2,p3,p4
  States 0,0,1,1
  Colors clRed,clGreen,clBlue,clLime,clYellow
  Points 0/0,50/0
  SPolylineTC rgLast
  Points 0/10,50/10,50/50,0/50
  SPolygonTC rgLast
  SRectTC rgLast,0/60,50/100
  SEllipseTC rgLast,0/110,20/130
  Fon clWhite
  Texts "p1=0 - Red","p2=0 - Green","p3=1 - Blue","p4=1 - Lime","No - Yellow"
}

```

```
STextTC rgLast,0/140
Signals p1,p1
States 0,1
Colors clRed,clGreen
SEllipseTC rgLast,30/110,50/130
Pen 1
Color clLtGray
Brush clLtGray
Texts "p1","p2","p3","p4"
Colors clRed,clBlue
Signals p1,p2,p3,p4
// чис. рядов, колонок, цвет текста, корд.углов
TableTC 4,1,clBlack,70/0,110/80,tbCirc //круги
}
}
```