

## Содержание

#### Введение

1. КРАТКИЙ ОБЗОР ГРАФИЧЕСКОГО РЕДАКТОРА	5
Функции Выбор список Атрибуты графического редактора Действие правой кнопки мыши Сабмодель Регистрация стандартного программного обеспечения OIS-DS Открытие экрана регистрации графического меню Вход в титульный экран Выбор имени модели Выбор имени модели Выбор способа действия Изменение масштаба изображения ВКЛ\ВЫКЛ Регистрация переменных Сохранение меню графического экрана Старт и завершение работы графического редактора Старт графического редактора Завершение работы графического редактора Краткий обзор графических экранов редактора	55666688899999010
2. ПАНЕЛЬ ИНСТРУМЕНТОВ ДЛЯ СОЗДАНИЯ МОДЕЛИ	17
Краткий обзор панели инструментов Инструментальные кнопки и их действие Кнопка движения точки Кнопка линии Кнопка ломаной линии Кнопка ломаной линии Кнопка создание сектора Кнопка создание сектора Кнопка азамкнутой кривой Кнопка замкнутой кривой Кнопка заполнения Кнопка заполнения (4 Типа) Кнопка заполнения прямоугольника текстом Кнопка заполнения прямоугольника текстом Кнопка копирования Кнопка копирования Кнопка реверса Кнопка реверса Кнопка масштаба Кнопка перемещения (шага)	17 19 20 21 22 23 23 24 25 25 26 29 30 31 32 33 44
3. МЕНЮ КОМАНД И ПАНЕЛЬ ИНСТРУМЕНТОВ	35
Меню файла (кнопка файл) Модели и подмодели Команда создания новой модели Команда открытия модели Команда объединения Команда сохранения Команда сохранить как Команда импорта точечного рисунка Команда экспорта GML Script Команда вызова списка экспорта переменных	35 36 37 39 39 41 41 42 43

Команда установки	44
Команда удаления модели	45
Команда печати	46
Команда выхода	46
Редактор меню	47
Команда отмены последнего деиствия	48
Команда удаления	49
Меню последовательности	49
Команда уплового поворота Команда диадаасна шкади (установка масштаба)	49 40
Команда диапазона шкалы (установка масштаоа) Вид меню	49 50
бид меню Команла вхола	50
Выберите меню	51
Вилерите меню	51
Команда «панорама»	53
Команда увеличения масштаба изображения	54
Команда уменьшения масштаба изображения	54
Команда изменения масштаба изображения	54
Команда «перенабор»	54
Команда перерисовки	54
Меню панели инструментов	55
Сохранение статуса панели инструментов	56
Восстановление статуса панели инструментов	56
Меню объекта	56
Команда группирования	58
Заполнение команды группы	58
Команда заполнения комплекснои группы	58
команда разгруппирования объекта	20 50
Графическая команда	00 59
графическая команда Команда выбора имени объекта	58
Команда высора имени освекта Команла текста	59
Команда текста Команда вызова информации об объекте	60
Команда изменения атрибутов отображения	61
Способ выбора объекта	62
Команда разрыва субмодели	62
Применение команды преобразования модели	62
Команда удаления преобразования модели	63
Команда вызова списка	63
Меню Динамики	63
Редактирование команды «динамика»	64
Команда переименования переменных	65
Команда отображения переменных	66
Команда назначения переменных	66
Меню вариантов	70
	70
Быстрый способ рисования (команда фиксации)	71
Быстрая привязка к сетке Быстрое соединение эдементов (фиксания к объекту)	71
Выстрое соединение элементов (фиксация к объекту)	71
Список выбора элементов	72
Список выбора субмолелей	72
Меню работы с точкой	73
Меню выбора способа работы	73
Команда точки	74
Центрирование	74
Поиск центра	75
Поиск точки	75
Повторное использование команды	75
Установка контрольной точки	75
Повторная установка	76
Меню Помощи	76

Команда вызова номера версии	76
4. ДИНАМИКА	72
Краткий обзор динамики	77
Формат динамики	77
Действия	78
Переменные	79
Подмодели	83
Динамические подмодели (процедура)	84
Атом тега для атома процесса	86
5. Создание Подмоделей	89
Изменение признаков отображения	89
Отображение индикатора цветом	90
Развёртка точечного экрана	91
Заполнение фигуры (барграф)	92
Стрелочный барграф «Амперметр»	95
Отображение динамики	97
6. Создание Плоских Моделей	101
7. Отображение времени для графического экрана	102
8. Резервная копия и восстановление	103
Создание резервной копии	103
Восстановление	132
9. Обслуживание файлов базовых моделей	104
Резервирование	106
Установка	107
Приложение 1. Рекомендации по динамике	108
Динамическое действие	108
Строка управления форматом	109
Стандарт с функцией библиотеки, признаков динамики	111
Приложение 2. SL-GMS Сохраненные Слова	112
Приложение 3 Стандартные субмодели	114
Список стандартных подмоделей	114
Детальный показ ошибок доступа	169
Приложения 4 Добавление стандартных функций	171
Содержание приложения	171
Образцовая папка	176
Описание переменных	176
Описание функций	178
Приложение 5. Спецификация информационных тегов	210
Список типа признака	283
Информация о тревогах	284
Регистрация тревог тегов РСS	286
SVR (Внутренний) Регистрация тегов тревоги	296
Регистрация тегов тревоги DPCS	298
Регистрация тегов тревоги PLC	303

Параметр точки («точечный дисплей»)	306
Представление тега PCS	306
SVR (Внутренний) представление тега	343
DPCS Представление тега	349
Выбор линии параметров	369
Тренд точечного дисплея	371

## 1. Краткий обзор Графического Редактора

Графический Редактор - инструмент рисунка, для создания графических моделей. Вы можете использовать этот инструмент для создания объектов экрана, и определить их динамическое поведение. Этот гибкий инструмент имеет множество функций помимо обычных графов и изображений.

Рисунок 1.1 - пример варианта, моделирования экрана графического редактора.



1.1 Экран графического редактора

## 1. Краткий обзор Графического Редактора

#### Функции

Этот раздел описывает главные функции Графического Редактора.

#### Выбор списка

В Графическом Редакторе, вы можете выбрать один или более объектов SL-GMS. Набор выбранных объектов {целей} называют Списком Выбора.

Список Выбора разрешает редактировать модели сгруппированными или отдельно. Элементы для любого объекта {цели} могут быть немедленно просмотрены.

Кроме того, они могут быть отобраны, добавлены, перемещены, или удалены из любой модели, чтобы облегчить высокую точность объекта.

#### Атрибуты графического редактора

Любой графический признак, Графического Редактора может быть изменён. Цвет объекта {цели}, положение {позиция}, масштаб, вращение, и так далее, являются действиями, атрибутов графического редактора, по отношению к объекту {цели}. Во временя выполнения, графические признаки объекта {цели} изменяются в ответ на изменения {замену} внешних данных, графического редактора.

#### Действие правой кнопки мыши

Щелкните правой кнопкой мыши, когда объекты {цели} отобраны, для отменены объектов {целей}. Щелкните правой кнопкой мыши, когда объект {цель} редактируется, действие отменено.

#### Подмодель

Графический Редактор позволяет многократно использоваться типовые подмодели, с сохраненными динамическими свойства первоначальной {оригинальной} Модели.

Различные модели дифференцируются от оригинала, переименованием переменных. Например, вы можете создать многоступенчатую систему из одинаковых подмоделей, отображая различные переменные того же процесса. Можно создать одну модель объединяя описания динамических свойств, которые управляются ТАGовой переменной.

Затем, создайте модель групповой системы, которая состоит из отдельных моделей. В каждом случае, переименуйте TAGNo переменную так, чтобы это имело уникальное название {имя} типа w1, w2, и т.д.

Также, можно изменять модели с многократными переменными. Например, в случае модель "Трансформатора", цвет первичной обмотки изменяется согласно одной цветной переменной, а цвет вторичной обмотки изменяются согласно другой переменной.

#### Регистрация стандартного программного обеспечения OIS-DS

Этот раздел описывает, как регистрировать экран (графическую модель) созданный с использованием **стандартного программного обеспечения OIS-DS** Инжиниринг в графическом меню стандартного программного обеспечения.

#### Открытие экрана регистрации графического меню

Когда вы выбираете [Функции Экрана] - [Меню] в экране Маіп, Появляется экран регистрации меню. Вы можете регистрировать до 128 Меню. Переменные изменяют информацию, зарегистрированную для данного экрана. Вы можете рассмотреть это изменение в том же самом изображении какое записано в указанном окне меню.

		1) 2) <b>Engin</b> ering Tool[C: dit Monitoring Screen Ia Senu 1 128/126	5) <b>ISD SDATA</b> Trind Ap	8) 6) AOI -DS sys em plication Litility W	7)  (indow Help
)	<u>No.</u>	Item Name	Item No.	Title String	Paramater' 📤
	1	System Alarm	0		
	2	Alarm Summary	0		
	3	Trend Menu	0		
	4	Point Display	0	PointPID_0001	TAGNO PID_0001
	5	Point Display	0	PointPID_0002	TAGNO PID_0002
	6	Program Menu	0	PointPID_0003	TAGNO PID_0003
	7 Parameter Setup		0		
	8	(*) Graphic Menu	34	Graphic 1	
	9	9 Point Display			TAGIND PID_0004
	10	(*) Group Overview	10	GroupDesiplay	
	11	(*) Trend Menu	4	TrendMenu 4	
	12	Tag Search	n		

	Рисунок 1.2	Окно	регистрации	меню
--	-------------	------	-------------	------

## 1. Краткий обзор Графического Редактора

#### Отображение Области/Входа

1) Стартовый номер страницы Номер показан на стартовой линии страницы.

2) Конечный номер страницы Номер показан на линии конца страницы.

Номер
 Регистрационный номер меню. Числа от 1 до 128.

4) Название раздела Позволяет выбирать название экрана, чтобы выбрать меню из блока списка.

5) Номер раздела Позволяет определять номер регистрации в подменю, когда номер раздел (подменю) В виде " (\*) " выбран из блока списка номеров. Могут быть выбраны числа от 1 до 2560.

6) Титульная строка

Регистрирует название экрана, назначенного в меню (до 30 знаков).

7) Переменная изменяющая информацию Для экрана точки и Графического Экрана, назначенного в меню, номер тега и параметр

зарегистрированы здесь (до 254 знаков). Номер тега регистрируется в формате "TAG№:\*\*\*\*" (где \*\*\*\*\* - реальный номер тега).

#### Кнопка

8) < Actual Menu View> Кнопка (кнопка действительного меню) Показывает окно в том же самом изображении, где и окно действительного меню.

#### Вход в титульный экран

Следуйте за шагами приведенным ниже.

1) Выбрать ячейку "титульная строка" и номер линии и зарегистрироваться.

2) Щелкнуть два раза левой кнопкой мыши, или нажать ВВОД. В ячейке появиться курсор.

3) Ввести название с клавиатуры клавиатуры.

4) Нажать ВВОД или выбрать другую ячейку. "Титульная строка" установлена.

#### Вход в имя модели.

Следуйте за шагами, изложенными ниже.

1) Выбрать ячейку имени модели и номер линии, и зарегистрировать название.

2) Щелкнуть два раза левой кнопкой мыши, или нажать ВВОД. Курсор появится в ячейке имени.

3) Набрать имя с клавиатуры.

4) Нажать ВВОД или выбрать другую ячейку. "имя модели" установлено.

#### Выбор способа действия (права)

Следуйте за шагами ниже.

1) Выбрать ячейку "Operation Mode" ("Способа Действия") в зарегистрированном графическом экране.

2) Щелкнуть два раза левой кнопкой мыши, или нажать ВВОД Блок списка появляется в ячейке.

3) Щелкнуть на кнопке стрелки. Появится лист списка.

4) Выбрать способ действия для зарегистрированного графического экрана. Способ действия установлен.

Замок (Lock) - блокирует действие оператора и контроль действий.

Оператор - позволяет действовать, если права станции OIS - оператор, администратор или инженер.

Наблюдатель (Supervisor) - позволяет действовать, если права станции OIS администратор или инженер.

Инженер - позволяет действовать, если способ действия станции OIS - инженер способ.

## Выбор масштаба изображения (Zoom ON/OFF)

Следуйте за шагами ниже.

1) Выбрать ячейку масштаб (" Zoom ") в зарегистрированном графическом экране.

2) Щелкнуть два раза левой кнопкой мыши, или нажать ВВОД. Блок списка появляется в ячейке.

3) Щелкнуть на кнопке стрелки. Появится лист списка

4) Выбрать ON or OFF.

OFF - изменение масштаба изображения графического экрана OFF. ON - изменение масштаба изображения графического экрана ON.

#### Изменение имени переменных

После выбора отображаемой точки зарегистрированного графического экрана, выбрать номер тега "Параметр", чтобы определить спецификацию экрана.

1) Выбрать ячейку "изменение имени переменной" в "номере" (строке) зарегистрированного экрана отображения точки, или графического экрана.

2) Щелкнуть два раза левой кнопкой мыши, или нажать ВВОД. Курсор покажется в ячейке.

3) Ввести номер тега отображаемого параметра точки графического экрана с клавиатура.

4) Нажать ВВОД, или выбрать другую ячейку. Номер тега для параметра установлен.

#### Сохранение графического меню зарегистрированного экрана

Для сохранения установленного меню, выбираете [Файл] - [сохранить Файл] ([File]-[Save File]).

Когда сохранение заканчивается, появляется окно подтверждения завершения. Щелкните на <хорошо> <OK> кнопке.

## Старт и завершение работы графического редактора

#### Старт Графического Редактора

Для старта графического редактора, введите в команду имя программы программа с клавиатуры. Название программы графического редактора и окружающей среды, требуемой для соответствующих действий зависят от системы.

#### РЕКОМЕНДАЦИЯ:

Инсталляцию руководства OIS рекомендуется произвести в \OIS\SLGMS.NT4

#### Пример

#### C:\OIS\SLGMS.NT4> drawt

Если вы запускаете графический редактор с OIS-DS стандартного программного обеспечения, нужно только должны щелкнуть на кнопке графического редактора, если он зарегистрирован зарегистрирован.

#### РЕКОМЕНДАЦИЯ:

Чтобы использовать графический редактора, нужна лицензия для OIS (ключ) графического редактора. Без ключа включенного в порт принтера невозможно начать работу.

#### Завершение Графического Редактора

- Вы можете использовать один из следующих двух путей для выхода из графического редактора. • Выбирают [Файл] - [Прикладной Выход] ([File]-[Application Exit])
- команду.
- Щелкают два раза на меню системы.

## Краткий обзор Графических Экранов Редактора

Этот раздел включает краткий обзор графического редактора. Детальное рассмотрение панели инструментов и меню команд, вынесено в главу 2 и 3.

Графический редактор имеет линейку (панель) меню в верхней части. Линейка (панель) меню имеет элементы для различных команд меню. Вы можете выбрать функции из панели инструментов.



Рисунок 1.3 Главный экран графического редактора

#### 1. Создание панели инструментов объекта.



Рисунок 1.4 Создание панели инструментов объекта

В панели инструментов объекта, можно выбрать тип объекта. Кнопки панели указывают различные типы объектов. Например, чтобы создать текстовый объект, щелкните на кнопку <A>. В этом случае создаётся текстовыё объект, пока вы не выберете другой объект в панели объектов или щелкнете на кнопке <Select> (Arrow).

С помощью этой панели можно редактировать созданные объекты (двигать, масштабировать, или вращать).

Кнопка редактировать (Edit) используется после того, как вы выбираете объект. Выбор осуществляется щелчком на объекте. Выбранный объект отмечается пунктирной линией, которая указывает, что объект добавлен к списку выбранных объектов.

Вы можете выбрать объект последовательностью команд "Edit"-"Select". Щелчок на выбранном объекте снова, отменяет выбор. После того, как вы выберете объект, щелкаете на одной из пяти кнопок редактора (Edit) в правой части панели создания объекта. Вы можете измерить (отредактировать) объект{цель}.

#### ССЫЛКА:

Если иначе не определено, "щелкнуть" в этом руководство означает щелкнуть левой кнопкой мыши.

Щелчок правой кнопкой мыши соответствует быстрому, щелчку на кнопке <Select>.

2. Панель инструментов формирования файла



Рисунок 1.5 Панель инструментов формирования файла

Вы можете использовать панель инструментов формирования файла, чтобы легко использовать команды файла типа: «новая модель», «открыть», «сохранить» или «печатать». Нажим одной из этих кнопок открывает диалог обмена. Детальное описание отнесено к главе 2.

#### 3. Панель редакции.



Рисунок 1.6 Панель редакции

В панели редактора вы можете выбрать подменю " Edit" меню типа, удалить объект, Вызвать рисунок или выбрать из кнопок панели.

Например, в случае кнопки, которая выстраивает порядок прорисовки объекта, (выбирает объект) нажмите кнопку [Front] и щелкните на другом объекте. Выбранный объект помещается сверху. Заказанный рисунок доступен для изменений.

4. Вид панели инструментов.



Рисунок 1.7 Вид панели инструментов.

Вид панели инструментов изменяющей вид рабочей панели и управляющей видом каждого инструмента рабочей панели.

5. Панель объектов.



Рисунок 1.8 Панель объектов.

В панели объектов вы можете сгруппировать объекты или показать диалог установки различнх признаков объекта.

Нажмите кнопку <Group> «группировать» после отбора объектов группы так, чтобы они могли быть перемещаемы или вращаемы как один объект.

Вы можете установить динамические свойства для сгруппированных объектов. Кнопка <Ungroup> «разгруппировать» отменяет группировку объектов. Разгруппировывает в индивидуальные объекты.

6. Панель динамики.



Рисунок 1.9 Панель динамики.

В панели динамики вы можете установить динамические свойства для рисуемых фигур и показывать диалог, изменения переменной имени.

7. Панель опций.



Рисунок 1.10 Панель опций.

Панель опций показывает диалог установки сетки и моментальной привязки рисунка к сетке. Три кнопки в правой части панель опций предназначены для быстрой модификации. <Snap Grid> быстрая сетка.

Кнопка фиксирует точку к сетке, когда вы создаете объекты.

Щелкая на <Snap Gravity> производят быструю модификацию точки так, чтобы точка была по сути сфотографирована из объекта, предварительно установленного.

8. Палитра.



Рисунок 1.11 Палитра.

Панель палитры открывает диалог, выбора подмодели или ее палитры.

9. Панель точки.



Рисунок 1.12 Панель точки.

В панели точки вы можете использовать точки и контрольную точку, которые составляют фигуру.

10. Статусная строка (линейка).

, part ,	ready	select 1	(0.620, 66286)	NUM	1
---	-------	----------	----------------	-----	---

Рисунок 1.13 Статусная строка (линейка).

Статусная строка (линейка) используется, чтобы показать помощь для каждой команды, имен моделей, списка выбора счета {графа} и координат мыши графического редактора. Справа от "выбора" ("Select") числа объектов, в настоящее время отобранных, показан список выбора (Select List).

Щелчком на объекте добавить объект к списку выбора. Щелчок на кнопке <Select> или правой кнопке мыши отменяет все объекты в списке выбора.

11. Вид редактора.



#### Рисунок 1.13 Вид редактора

В виде редактора (Edit View) вы можете редактировать модели. Когда вы изменяете {заменяете } размер, отношение ширины к высоте остается 100:75. Вы не можете открыть, больше чем один (Edit View).

Edit View обычно показывает от (0,0) до (100,75) в координате окна WC-window системы.

#### ССЫЛКА:

Координата редактора (Edit View) не является точкой. Это изображение в виртуальных координатах координат окна WC.

Вы можете использовать команды типа "Pan", "Zoom-In", or "Zoom-Out" " панорамирование ", " изменение масштаба изображения ", чтобы редактировать другие части модели. [Edit]-[Input] [Редактируйте] - [Вход{Вклад}], также использует систему координаты окна WC. Во время выполнения OID-DS, отображается полная область модели.

12. Атрибуты графического диалога.

Вы можете открыть диалог графических признаков (Graphic Attributes), выбирая [Объекты {Цели}] - [Признаки] - [Графический] ([Objects]-[Attributes]-[Graphic]) в строки меню. В этом диалоге, вы можете изменить {заменить } атрибуты цвета, заполнение цветом, тип линии, шрифта, и фоновый цвет графических объектов.

Диалог представляет вкладки, которые управляет четырьмя признаками; заполнение, линия, текст, и фоновый цвет. Выбор каждой вкладки открывает диалог, соответствующий её признаку.



Рисунок 1.15 Графические атрибуты (заполнение участка закрашивания)

При помощи графических атрибутов (заполнение участка закрашивания) вы можете изменить признаки, связанные с закрашиванием участков отобранного объекта. Выберите объекты, которые закрывают области типа прямоугольника, многоугольника, или заполнения текста в прямоугольнике рабочего вида и щелкните на кнопке <Fill> (участок закрашивания). Вы можете заполнить область в соответствии с набором признаков на диалоге.

Вы можете установить образцы и их тип, используя этот диалог. Нажим кнопки [Unfill] отменяет заполняющийся признак выбранных объектов и изменяет их на состояние до заполнения.

Щелчком на кнопке <Другие Цвета> в каждом диалоге, вкладка открывает "256 Цветных Палитр", При помощи которой вы можете выбрать один из 256 цветов.

#### ССЫЛКА:

Изменение {Замена} графических признаков действительно только для объектов, созданных в рабочей области.

Для присоединенных подмоделей, измените признаки подмоделей непосредственно.

256 color palette															
0	1	2	3	4	5	6		\$	3	10	11	12	13	14	15
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
2	33	34	35	36	37	38	39	40	41	42	43	-14	45	46	47
18	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
54	65	66	6.7	68	69	70	71	72	73	74	75	76	77	78	79
0	81	82		84	\$5	86	87	88	89	90	91	92	93	94	95
96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
12	113	1 <b>1</b> 4	115	116	117	118	119	120	121	122	123	124	125	126	127
28	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
44	145	146	147	148	149	150	151	152	153	154	155	156	157	158	15 9
60	16-1	162	163	<u>164</u>	165	166	167	168	16-9	170	171	172	173	<b>17</b> 1	175
76	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
92	193	194	195	196	1.97	198	199	200	201	202	2.03	204	205	206	207
Ø\$	209	210	211	212	213	214	215	21.6	21.7	218	219	220	221	222	22.3
22.4	225	226	227	228	229	230	231	232	233	234	235	236	237	238	23.9
24.0	241	242	243	2.44	245	246	247	248	249	250	251	252	253	254	255
							(	Cl	056	3					

Рисунок 1.16 Цветовая палитра

Graphic Attributes	×
Fill Line Text Background Color	
Style Edge Width	0       1       2       3       1       6       6         0       9       16       11       12       13       14       15         16       17       18       19       26       26       27       28       28       22         20       26       26       27       28       29       36       21
	Other Colors

Рисунок 1.17 Графические атрибуты. Таблица линий.

Во вставки "Линии" диалога графических атрибутов, вы можете изменить признаки, связанные с информацией о линиях выбранных объектов. Выберите объекты в рабочем виде и выберите "Стиль", ("Style") чтобы изменить стиль линий в объектах. Выберите "Ширину поля" ("Edge Width"), чтобы изменить ширину от края. Щелкните на цветной палитре и установите цвет линий.

Graphic Attributes		x
Fill Une Text	Background Color         Type         © True Type         © Hershy         Direction	
	Font/Size	

Рисунок 1.18 Графические атрибуты. Вкладка «Текст»

Во вкладке "Текст" Графические атрибуты, Вы можете изменить признаки выбранного объекта касающиеся текста.

Выберите текст или заполняемый текстом прямоугольник в рабочем виде и щелкните на <Шрифт/Размер> (<Font/Size>). Нажатая кнопка открывает диалог выбора шрифта. Выберите шрифт и его размер, чтобы изменить текст объекты в рабочем представление. Вы можете выбрать кнопку <Выравнивание Текста>, чтобы выровнять текст по правому или левому краю, или изменить цвет текста, выбирая желательный цвет от цветной палитры.

Graphic Attributes	×
Fill   Line   Text   Background Color	
Set background Color	0 1 2 3 4 5 6 1 0 9 18 11 12 13 14 15 16 17 18 19 22 21 22 23
Update to the latest	25 25 27 28 29 38 31
No background color object	
	Other Colors

Рисунок 1.19 Графические атрибуты. Подменю «фоновый цвет»

В Подменю «фоновый цвет» вы можете установить фон (цвет), который делает объекты невидимыми, когда вы перемещаете, преобразовываете, или удаляете объекты .

Перемещение или удаление объекта выполняется из предыдущего положения {позиции}, вместе с фоновым цветом. Фоновый цвет применяется для всех объектов в моделях и подмоделях (когда фоновый цвет используется для подмодели).

Щелчок на "установка фонового цвета" ("Set Background Color") проверяет блоки признаков добавления фонового цвета к рабочей модели. Фоновый цвет выбирается из цветовой палитры. Для SL-GMS, заполняющий цвет, для заполнения объекта, расположенный в задней части всех объектов в модели - второстепенный цвет. Поэтому, для изменения фонового цвет объекта, расположенного в задней части всех объектов в модели, изменяют фоновый цвет в этой вкладке. Когда заполняющийся цвет объекта изменен в соответствии с вкладкой заполнения, измененный цвет становится фоновым.

Чтобы проверить текущий фоновый цвет, нажмите кнопку <Последнее изменение> (<Update to the Latest>) во вкладке «фоновый цвет».

Когда модель не заполнена объектами, никакой фоновый цвет не может быть установлен.

# 2. Панель инструментов для создания объекта и её использование.

Глава 2 описывает панель инструментов для создания объекта, которая используется, чтобы рисовать графику, используя графический редактор, и способы её использования.

Панель инструментов для создания объекта показывает различные типы объектов, которые могут быть созданы, используя графический редактор. Различные, созданные объекты, добавляются к текущей модели.

Когда вы щелкаете на кнопке, перечень действий становится выдвинутым на первый план и остается в этом состоянии, пока вы не щелкаете на любой другой кнопке типа кнопки <Select>.

Для различных объектов, показанных в панели создания объекта, вы должны выбрать набор определенных точек в рабочем окне. Например, набор точек может быть центрами для создания окружностей.

Чтобы окончить любой тип процесса создания объекта, щелкните на любой другой кнопке в панели создания объекта или на кнопке «выбор» (<Select>). Это действие одинаково для всех типов объектов.

#### Краткий обзор создания объекта.

Панель инструментов для создания объекта имеет кнопки для различных типов объектов, созданных с помощью графического редактора и кнопки, для редактирования созданных объектов типа масштабирования и вращения.



Рисунок 2.1 Панель инструментов для создания объекта.

Панель инструментов для создания объекта имеет следующие кнопки (слева на право)

1. Кнопка выбора.

Заканчивает действия для выбора объекта.



2. Кнопка движения точки.

Точки (шаги), которые составляют объект (соединяя точки получают контур объекта).



3. Кнопка линии.

Запуск команд рисующих линии объекта.



Кнопка ломаной линии.

Запуск команд рисующих ломаные линии объекта.



5. Кнопка кривой линии.

Запуск команд рисующих кривые линии объекта..





Запуск команды рисующей текстовые объекты.



15. Кнопка заполненного круга.

Запуск команды рисующей заполненные прямоугольники с текстом.

17. Кнопка копирования.

Запуск команды копирующей объекты.

18. Кнопка вращения.

Запуск команды вращающей объекты



19. Кнопка поворота.

Запуск команды поворачивающей объекты



œ.

20. Кнопка изменения размера.

Запуск команды изменяющей размеры объекта

21. Кнопка перемещения.

Запуск команды перемещающей объекты

#### Инструментальные кнопки и их действие

#### Кнопка перемещения точки.

Эта кнопка позволяет перемещать точки выбранного объекта. Одновременно может быть выбран только один объект. Если выбрано несколько объектов, перемещаются только точки выделенного объекта.

Когда кнопкой перемещения точки <Move Point> щелкают, точки выбранного объекта отмечаются маркерами X-стиля.

Когда необходимая точка выбрана щелчком мыши, другие маркеры исчезают.

Выбирая точки для перемещения, щелкая в рабочем представлении, объект можно повторно изменять.

Новые положения {позиции} могут быть выбраны, пока не щелкают правой кнопкой мыши. Три точки сектора или текст не может быть перемещен.

Пример:



#### Перемещение точек объекта

1) Выбрать объект, который вы хотите редактировать. Щелкните на объекте, который вы хотите редактировать.

2) Переместить точки.

Щелкните на кнопке < Движение точки> (<Move Point>). В выделенных точках появятся маркеры.

3) Щелкнуть на точке, которую вы хотите переместить.

Другие маркеры исчезают.

4) Щелкнуть на точки предназначенной для изменения.
Объект повторно изменён.
Повторите это действие сколько необходимо. Когда вы определили соответствующее положение местонахождение точки, щелкните правой кнопкой мыши.
Маркеры повторно покажутся в определенных точках.
Чтобы переместить другие точки, повторите шаги 2 - 4.

5) Закончить работу с перемещением точек. Щелкните правой кнопкой мыши.

#### Кнопка линии

Этой кнопкой создаются прямые линии. Прямые линии имеют только две точки, в отличие от ломаных которые могут иметь больше чем две. Если выбрана третья точка, это - следующая точка линии.



#### Создание линии объекта.

1) Выбрать способ (кнопку) создания линии. Щелкните на кнопке <Line> в панели создания объекта.

2) Выберите первую точку для начала прямой линии. Переместите курсор мыши в позицию (а) и щелкните на ней.

3) Выбрать вторую точку для конечной точки прямой линии.

Переместите курсор мыши в позицию (b) и щелкните на ней, чтобы показать прямую линию. Чтобы создать другую прямую линию, повторите шаги 2 и 3,.

4) Окончание создания линий.

Щелкните на кнопке <Select> или любой другой кнопке в панели создания объекта.

#### Ломаная линия

Эта кнопка добавляет к модели ломаные линии, составленные из прямых линий. Ломаные отличаются от прямых линий тем, что прямые линии имеют только две точки, тогда как ломаные имеют больше чем две точки, связывающие прямые линии.

# Ломаная линия создаётся, выбором начальной точки, а затем создаётся так много дополнительных точек как необходимо. Прямая линия создаётся между каждой парой выбранных точек.

Когда щелкают кнопкой <Select>, заканчивают создание объекта.

Пример:



#### Создание ломаной линии объекта

1) Выбрать способ (кнопку) создания ломаной линии. Щелкните на кнопке <Polyline> в панели создания объекта.

2) Выберите первую точку. Переместите курсор мыши в позицию (а) и щелкните на ней.

Выбрать вторую точку.
 Переместите курсор мыши в позицию (b) и щелкните на ней.

Выбрать третью точку.
 Переместите курсор мыши в позицию (с) и щелкните на ней.

5) Выбрать следующую точку.

Переместите курсор мыши в позицию (d) и щелкните на ней.

6) Выбрать последнюю точку.

7) Закончить создания ломаной линии. Щелкните на кнопке <Select> или любой другой кнопке в панели создания объекта, чтобы закончить создание ломаной линии.

Чтобы создавать другую ломаную линию, щелкните правой кнопкой мыши вместо кнопки <Select>.

#### Кнопка создания кривой линии.

Графический редактор позволяет плавно соединять выбранные точки, для создания кривой линии, которая проходит через эти точки. Когда точки выбраны, каждая пара точек связана ломаной. Щелчок на кнопке <Select> заменяет ломаную гладкой кривой линией.

Пример:



Создание кривой линии объекта.

1) Выбрать способ (кнопку) создания кривой. Щелкните на кнопке <Spline> в панели создания объекта. 2) Выбрать первую точку. Переместите курсор мыши в позицию (а) и щелкните на ней.

3) Выбрать вторую точку.

Переместите курсор мыши в позицию (b) и щелкните на ней.

4) Выбрать третью точку.

Переместите курсор мыши в позицию (с) и щелкните на ней.

5) Выбрать дополнительные точки где необходимо.

6) Выбрать последнюю точку.

7) Закончить создание ломаной.

Щелкните на кнопке <Select> или любой другой кнопке в панели создания объекта, чтобы заменить ломаную, кривой.

Чтобы создавать другую кривую, щелкните правой кнопкой мыши вместо кнопки <Select>.

#### Кнопка создания сектора.

Щелчком этой кнопки создаётся сектор. Строятся три точки по дуге. Графический редактор создает сектор, который лучше всего соответствует трём выбранным точкам.

Создание трёх точек:

Точки сектора всегда рисуются против часовой стрелки. Вторая и третья точка должны быть выбраны против часовой стрелки от первой точки.

Пример:



#### Создание сектора объекта по трем точкам.

1) Выбрать кнопку создания сектора по трем точкам.

Щелкните на кнопке <Three Points Sector> в панели создания объекта.

2) Выберите первую точку для края сектора.

Переместите курсор мыши в позицию (а) и щелкните на ней

3) Выбрать вторую точку против часовой стрелки от первой точки, которую вы создали. Переместите курсор мыши в позицию (b) и щелкните на ней.

Выбрать третью точку против часовой стрелки от второй точки, которую вы щелкнули.
 Переместите курсор мыши в позицию (с) и щелкните на ней.
 Три точки соединятся и маркеры исчезнут. Для создания других секторов, повторите шаги 2 - 4.

5) Закончить создание секторов.

Щелкните на кнопке <Select> или любой другой кнопке в панели создания объекта. Чтобы создавать другие сектора щелкните правой кнопкой мыши.

#### Кнопка создания многоугольника.

Кнопка создания многоугольников добавляет многоугольные объекты к модели.

Многоугольники (замкнутые ломаные линии) могут появляться как треугольники, трапеции, или любые формы, составленные из прямых, замкнутых между собой, линий.

Графический редактор автоматически соединяет первую точку с последующими выбранными точками, когда на выбранных местах щелкают мышью. Многоугольники могут быть заполнены.

Пример:



#### Создание многоугольника

1) Выбрать способ (кнопку <Polygon>) создания многоугольника. Щелкните на кнопке <Polygon> в панели создания объекта.

2) Выбрать первую точку.

Переместите курсор мыши в позицию (а) и щелкните на ней.

3) Выбрать вторую точку. Переместите курсор мыши в позицию (b) и щелкните на ней.

4) Выбрать третью пункт {точку}. Переместите курсор мыши в позицию (с) и щелкните на ней.

5) Выбрать дополнительные точки как вам необходимо.

6) Закончить операцию создания многоугольника.

Щелкните на кнопке <Select> или любой другой кнопке в панели создании объекта, чтобы закончить операцию создания многоугольника. Для создания другого многоугольника, щелкните правой кнопкой мыши.

Первая точка и последняя точка соединяются автоматически.

#### Кнопка создания замкнутой кривой.

Замкнутые кривые создаются в рабочем виде графического редактора, щелкая кнопкой «Closed splines». Графический редактор соединяет точки в многоугольник, при щелчке кнопки <Select>.

Щелчок на кнопке <Bыбор> (<Select>) при закрытии операции создания замкнутой кривой, соединяет точки концов многоугольника и преобразует его в замкнутую кривую.

Созданные замкнутые кривые могут быть заполнены.

Пример:



Создание замкнутой кривой:

1) Выбрать кнопку создания замкнутой кривой. Щелкните на кнопке <Closed Spline> в панели создания объекта.

2) Выбрать первую точку. Переместите курсор мыши в позицию (а) и щелкните на ней.

Выбрать вторую точку.
 Переместите курсор мыши в позицию (b) и щелкните на ней.

4) Выбрать третью точку.

Переместите курсор мыши в позицию (с) и щелкните на ней.

5) Выбрать дополнительные точки как вам необходимо.

6) Закончить операцию создания замкнутой кривой.

Щелкните на кнопке <Select> или любой другой кнопке в панели создания объекта, чтобы заменить многоугольник, соединяющие выбранные точки, замкнутой кривой. Чтобы создать другую замкнутую кривую, щелкните правой кнопкой мыши на кнопке <Select>. Первая и последняя точка автоматически связаны.

#### Кнопка создания прямоугольника.

С помощью этой кнопки создаются прямоугольники по двум выбранным противоположным углам для каждого прямоугольника.

Прямоугольники могут быть заполнены.

Пример:



Создание прямоугольника:

1) Выбрать кнопку создания прямоугольника.

Щелкните на кнопке <Rectangle> в панели создании объекта.

2) Выбрать первый угол.

Переместите курсор мыши в позицию (а) и щелкните на ней.

3) Выбрать второй угол.

Переместите курсор мыши в позицию (b) и щелкните на ней. Появляется прямоугольник. Чтобы создавать другой прямоугольник, повторите шаги 2 и 3.

4) Закончить создание прямоугольников.

Щелкните на кнопке <Select> или любой другой кнопке в панели создании объекта.

#### Кнопка создания круга

Круги создаются при помощи этой кнопки. Местоположение и размер круга определяются выбором центра и точки на окружности. Круги могут быть заполнены.

Пример:



Создание окружности (круга).

1) Выбрать кнопку создания круга.

Щелкните на кнопке <Circle> в панели создании объекта.

2) Выбрать центр круга.

Переместите курсор мыши в позицию (а) и щелкните на ней.

3) Выбрать точку на окружности (предполагаемой окружности). Переместите курсор мыши в позицию (b) и щелкните на ней. Чтобы создавать другой круг, повторите шаги 2 и 3.

4) Закончить создания круга. Щелкните на кнопке <br/> Select> или любой другой кнопке в панели создании объекта.

#### Кнопки заполнения (4 типа)

Панель создания объекта, графического редактора, имеет четыре типа заполняющих кнопок.



Заполнение многоугольника

Заполнение замкнутой кривой

Заполнение прямоугольника

Заполнение круга

Когда многоугольник, замкнутая кривая, прямоугольник, или круг созданы, графический редактор автоматически может заполнить их цветом.

Заполненный объект остаётся в том же положении где был создан незаполненный.

#### Кнопка создания текстового объекта.

Текст объект добавляется с помощью этой кнопки. Текст - любой пригодный для печатания символ}, введенный с клавиатуры.

Текстовый объект создается выбором точки, для размещения текста и затем, и набором символов с клавиатуры.

Text Entry	×
Enter Text:	
Cancel	OK

Рисунок 2.2 Окно диалога создания текстового объекта.

Чтобы создавать текст в несколько строк, в окне диалога создания текстового объекта, между предполагаемыми строками, набрать с клавиатуры два символа "\n". Предположим, что вы хотите создать следующий текст:

Это – пять строк текста, с пробелом в третьей строке

Для этого ввести с клавиатуры:

- \n линия текста, \n\n {пробел} \n линия текста <ВВОД> и нажать кнопку <ОК>.

Вы можете изменить элементы текста, что вы создали, пока вы не выберете следующий текстовый объект. Вы также можете изменить весь текст.

Пример:



Создание текстового объекта:

1) Выбрать кнопку создания текста. Щелкните на кнопке <Text> в панели создании объекта.

2) Выбрать точку для размещения текста. Переместите курсор мыши в позицию (а) и щелкните на ней. Появляется диалог создания текста.

Введите в текст с клавиатуры.
 Введите следующий.
 Сосредоточьте, выровняйте текст <ВВОД>
 Текст, который вы ввели, появляется на экране.

4) Выбрать <текстовая вкладка>( <Text Tab>) в диалоге графических признаков (см. Главу 1). Выберите соответствующий признак текста, из вкладки текста. Чтобы создавать другой текстовый объект, повторите шаги 2 - 4.

5) Закончить создания текстового объекта: Щелкните на кнопке <br/>
Select> или любой другой кнопке в панели создании объекта.

#### Изменение атрибутов Текста

Панель создания объекта имеет две кнопки для создания текстового объекта; кнопка <Text> и кнопка <Заполненный прямоугольник текста> (<Filled Text Rectangle>). Те же самые атрибуты используются когда уже создан.

Цвет текста изменяется щелчком на цветной палитре в признаке "Текста" диалога графических признаков, который выбирается ;"Объект" - "Признаки" - "Графика" "Object"-"Attributes"-"Graphics".



Рисунок 2.3 (Графические атрибуты) Вкладка «Текст».

В текстовой вкладке диалога графических признаков, вы можете изменить признаки текста типа; настройка, шрифт, размер, расположение.

Многие из этих признаков функционируют также, как цвет. Например, для всех текстовых объектов, созданных после изменения признаков; изменение (если необходимо) нужно производить повторно.

Для изменения признаков текстового объекта; выберите объект измените его атрибуты в текстовой вкладке..

Настройка текста устанавливает смещение для текста относительно точки, выбранной как положение текстового объекта.

Для настройки текста во, вкладке текста, имеется 15 кнопок. Кнопка, указывающая текущее выравнивание, выделена

Кнопки горизонтального выравнивания указывают смещение от центра вправо и лево. Кнопки вертикального выравнивания указывают (определяют) основание (нижний край), основу (базу), половину (середину), шапку (заголовок) и вершину текста.

Щелкните на соответствующей кнопке, чтобы изменить выравнивание. Высота заглавной буквы - высота письма "А".

Команда [Объект] - [Признаки] - [Текст] ([Object]-[Attributes]-[Text]) изменяет содержание выбранного текстового объекта.

Выберите текст, который вы хотите изменить, и выберите [объект] - [Признаки] - [Текст]. Появляется окно диалога редактора текста.

После того, как вы отредактировали текст, щелкаете на кнопке <Apply> (применить).

Edit Text		×
Edit Text:		
abcdefg		
Apply	Reset	Cancel

Рисунок 2.4 Окно диалога текстового редактора.

Текст заменяется для каждого объекта отдельно. Когда части текста выбраны, редактируется Порядок отображения, Вы можете выбрать направление текста «направление текста» (direction of the Text).

Вы можете выбрать из двух направлений; "Слева направо" или "сверху вниз". Направление по умолчанию, "Слева направо". Чтобы изменять направление текста, щелкните на стрелке направления, которое вы желаете.

#### ВЫБОР ШРИФТА

Типы "Истинный Тип" ("True Type") и "Hershy" - типы шрифта текста. "Истинный Тип" является истинным шрифтом, установленным MS-Windows.

Если Вы используете шрифт, которые установлены в вашем компьютере; [MS Gothic] и [MS Mincho] установлены в стандартном MS-Windows.

Щелчком кнопки <Font/Size> (шрифт/размер) вызывается диалог выбора шрифта. Форма диалога, который появляется, зависит от опции, которую вы выбрали в "Типе" ("Туре"), то есть "Истинный Тип" или "Hershy".

В диалоге <Font/Size>, вы можете изменить размер и стиль шрифта.

Font		? ×
Eont: Arial	Font style: Regular	Size:
Trial         Trial Black         Trial Narrow         Trial Conic Sans MS         Trial Courier New         Trial Caramond	Regular Italic Bold Bold Italic	14     ▲       16     Cancel       18     20       22     24       26     ▼
	Sample AaBbYy	Zz
	Seript:  Western	<b>_</b>

Рисунок 2. Диалог конфигурации шрифта типа "Истинный Тип" ("True Type")



Рисунок 2.6 Диалог конфигурации шрифта типа "Hershy"

#### ССЫЛКА:

Содержание, показанное в "Типе" ("Туре") - не всегда является шрифтом выбранного объекта. Если вы изменяете номер или размер "Истинный тип" объекта в диалоге "Hershy" шрифт, или если вы изменяете стиль шрифта или размер шрифта "Hershy" объекта в диалоге "Истинный тип" шрифта, шрифт не появится как вы ожидаете.

#### Кнопка заполнения прямоугольника с текстом

Щелкая кнопкой "Заполнение прямоугольника с текстом" ("Filled Text Rectangles") создаются прямоугольники с текстом заполненные цветом.

Текст, в прямоугольнике, написан поверх заполнения. В «заполненном прямоугольнике с текстом» быстрее стереть и изменять текст, чем стереть и изменить обычный текстовый объект.

«Заполненные прямоугольники текста» используются, чтобы создать текст, изменяться появлением.

Изменение текста требует, чтобы старый текст был стерт, изменяя текст в стиранием цвета - относительно медленное действие для большинства графических автоматизированных рабочих мест.

Текст, в «заполненных прямоугольниках текста», удален, намного быстрее.

«Заполненные прямоугольники текста» создаются выбором двух противоположных углов для прямоугольника и входом в показанный текстовый диалог.

Чтобы создавать текст с несколькими строками, вводите для выделения строк, два символа "\n" (вы не должны печатать двойной апостроф при вводе этих символов).

Для информации относительно того, как изменять признаки (атрибуты) текста в [Признаки Объекта] - [Текст] ([Object Attributes]-[Text]) вкладке, обратитесь к "Изменяющимся Признакам Текста" ("Changing Text Attributes").

Для информации относительно того, как изменять признаки (атрибуты) прямоугольников (форма, ширина, цвет, временное изменение цвета сторон, и т.д.). Обращайтесь к разделу; вкладка [Графические Признаки] - [Линия] ([Graphic Attributes]-[Line]).

#### ССЫЛКА:

Графический редактор не препятствует появлению текста вне «заполненного прямоугольника текста».

Текст, появляющийся вне «заполненного прямоугольника» не стирается, когда изменяется текст в прямоугольнике. Важно что в прямоугольнике может быть достаточно большой текст.

Пример:



#### Создание «заполненного прямоугольника текста»

1) Выбрать кнопку создания заполненного «прямоугольника текста». Щелкните на кнопке <Заполненный Прямоугольник Текста> впанели создании объекта.

2) Выбор атрибутов.

Из текстовой вкладки диалога графических признаков, выберите соответствующие атрибуты текста. Во вкладке заполнения (цветная палитра) подберите заполняющий, цвет и цвет текста для «заполненного прямоугольника текста».

3) Выбрать угол для прямоугольника. Переместите курсор мыши в позицию (а) и щелкните на ней.

4) Выбрать противоположный угол.

Переместите курсор мыши в позицию (b) и щелкните на ней.

Появляется «заполненный прямоугольник». В то же самое время открывается диалог ввода текста.

5) Ввести текст от клавиатуры.

В диалоге ввода текста, введите следующее:

График напряжения

Текст, в который вы ввели, появляется в прямоугольнике. Для создания другого объекта «заполненный прямоугольник», повторите шаги 3 - 5.

6) Закончить создание «Заполненных прямоугольников текста».

Щелкните на кнопке <Select> или любой другой кнопке в панели создании объекта.

#### Кнопка копирования

С помощью этой кнопки создаются копии объектов.

Как только объекты выбраны, графический редактор выделяет их прямоугольником эха и помещает маркер типа X, показывая контрольную точку для копии. Контрольная точка указывает, где скопированные объекты появляются по отношению, к выбранной точке.

Пример:



Копирование Объекта

1) Выбрать объект.

Щелкните на объекте, который вы хотите копировать. Число объектов, которые вы выбрали, показано в правой строке статуса.

Создать копии объекта.
 Щелкните на кнопке <Replicate> (копировать) в панели создании объекта.
 Объект который вы выбрали, выделен с пунктиром, и показаны маркеры (ссылки).

3) Щелкнуть на точке, где вы хотите создать копию. Копия появляется и добавляется к списку выбора.

4) Редактировать копию.

Используя редактирующие кнопки, в панели создании объекта, можно редактировать копию как вам необходимо.

Чтобы создавать другую копию, повторите шаги 3 и 4.

5) Закончить копирование.

Щелкните на кнопке <Select> или любой другой кнопке в панели создании объекта.

#### Кнопка вращения

Эта кнопка поворачивает выбранные объекты так, что они появляются как зеркальное изображение оригинала объекта.

Для создания зеркального объекта, объекты сначала выбираются к «зеркалу» (линии). Нажимается кнопка <Reverse>. Создаётся ссылка (линия), щелкая дважды, чтобы выбрать две точки линии «зеркала».

#### **INSTRUCTON:**

Вы должны фиксировать преобразования.



Создание зеркально отраженного объекта

1) Выбрать объект, который вы хотите к зеркально отразить. Щелкните на объекте, который в хотите зеркально отразить.

2) Зеркальное отражение объекта.

Щелкните на кнопке <Reverse> (вращение).

Щелкните в одном конце линии ссылки (оси зеркального отражения). Щелкните в другом конце линии ссылки (оси зеркального отражения).

выбранный объект исчезает, и появляется зеркальный объект.

3) Закончить создание зеркально отраженных объектов. Повторите шаг 2 или щелкните на кнопке <Select>.

#### Кнопка вращения

Кнопка вращения, вращает выбранные объекты. По умолчанию, объекты вращаются вокруг их центров.

Точку, вокруг которой объект вращается, называют его контрольной точкой. Контрольную точку объекта изменяют используя меню выбора точки изменения.

Когда щелкают кнопкой вращения, появляется пунктирная линия ссылки, с ее левым концом в контрольной точке.

Угол вращения объекта выбирается щелчком кнопкой мыши в точке, относительно контрольной точки и пунктирной линии ссылки. Ввод определенного угла осуществляется с использованием опции выбора угла вращения в меню редакции.



Рисунок 2.7 Диалог вращения.

Принято считать положительный угол вращения объекта - против часовой стрелки, а отрицательный угол вращает объекта - по часовой стрелке.

Чтобы закончить "Вращение", щелкните на кнопке <Select> или правой кнопкой мыши.

#### ИНСТРУКЦИЯ:

Вы должны обратиться к назначениям преобразований.

Пример:



#### Вращение Объекта

1) Выбрать объект, который вы хотите вращать. Щелкните на объекте, который вы хотите вращать.

2) Выбрать кнопку вращения в панели создания объекта. Щелкните на кнопке <Rotate>, чтобы показать линию ссылки.

3) Выбрать угол вращения.

Щелкните на точке, которая указывает угол относительно контрольной точки и линия ссылки. Объект повернётся вокруг контрольной точки.

4) Закончить «вращение».

Щелкните на кнопке <Select>, чтобы закончить выбор угла вращения.

#### Кнопка масштабирования

Эта кнопка изменяет размеры объекта. Масштабирование делает объекты больше или меньше. По умолчанию объекты равномерно изменяются в X и Y направлениях.

Опция "Равный" ("Equal"), может быть выбрана в меню [Редактирование] - [Масштаб] ([Edit]-[Scale]) так, чтобы объекты не были искажены. Например, если круг неравномерно масштабирован, получится эллипс.

Для масштабирования объектов щелчком добавить их к списку выбора, затем щелкнуть кнопкой <Scale> (масштаб).Объекты окружаются размытой границей. Твердая граница, представляет масштабирующий фактор.

Фактор масштабирования выбирается перемещением твердой границы в области размытой границы и щелчком объекты делаются меньше, или щелчком вне размытой границы, чтобы делать объекты больше. (Сжимать или растягивать)

Чтобы закончить масштабирование, щелкнув на кнопке <Select> или правой кнопке мыши.

Чтобы изменять размер неравноценно в горизонтальном или в вертикальном направлении, выберите в меню "Горизонтально" или "Вертикально" редактировать ([Edit]-[Mode]) Чтобы масштабировать изменяя/не изменяя отношение длина/ширина, выберите "Неравный/Равный коэффициент" в меню[Редактирование] - [Способ] ([Edit]-[Mode]).

Для ввода разрядов масштаба с клавиатуры, выберите кнопку <Scale> в панели создания объекта и выбрать команду [Редактировать] - [Разряд масштаба] ([Edit]-[Scale Rate]).

Scale	×
Enter Scale rate:	ОК
	Cancel

Рисунок2.8 Диалог изменения масштаба.

Объект масштабируется от его контрольной точки, которая является центром расширения объекта по умолчанию.

Ссылка на используемую контрольную точку осуществляется с использованием опций выпадающего меню контрольной точки.

#### ИНСТРУКЦИЯ:

Вы должны внимательно назначать преобразование.

Пример:



#### Изменение размера объекта

1) Выбрать объект, размер которого вы хотите изменить. Щелкните на объекте, размер которого вы хотите изменить.

#### 2) Изменить размер объекта.

Щелкните на кнопке <Scale> (масштаб). Появляется размытая граница и маркеры ссылки. Чтобы уменьшить объект, щелкните внутри размытой границе. Чтобы увеличить объект, щелкните вне размытой границы. Объект увеличится или уменьшится..

3) Закончить изменение масштаба.

Повторите шаг 2, или щелкните на кнопке <Select>, чтобы закончить изменять размер.

#### Кнопка движения{шага}

Кнопка шага выбирает объекты. Когда кнопкой <Move> (движение) щелкают «прямоугольник выделения» вокруг объектов, объекты перемещаются.

Прямоугольник расширения показывает четыре угла, которые отмечают степень перемещения объекта (ов).

Маркер показывает контрольные точки используемые для перемещения выбранного объекта. Эта контрольная точка может быть изменена используя опции выпадающего меню в контрольной точке.

Выбранные объекты перемещаются щелчком в выбранной контрольной точке.

Для окончания "Движения {Шага}" (перемещения) щелкните на кнопке <Select> или правой кнопке мыши.

#### ИНСТРУКЦИЯ:

Вы должны обратиться к следующим пунктам преобразования:

Пример:



#### Перемещение объекта

1) Выбрать объект, который вы хотите переместить. Щелкните на объекте который вы хотите переместить.

2) Переместить объект.

Щелкните на кнопке <Move> (перемещение) и новой точке для перемещённого объекта. Выбранный объект появляется в новом положении.

3) Закончить перемещение объектов.

Повторите шаг 2, или щелкните на кнопке <Select> (выбор) для окончания перемещения.

# 3. Меню команды и панель инструментов

Глава 3 описывает меню команд и панель инструментов графического редактора. Все команды меню графического редактора могут быть выполнены с помощью панели инструментов.

Eile	<u>E</u> dit	<u>V</u> iew	Object	<u>D</u> ynamics	<u>O</u> ptions	Palettes	<u>P</u> oint	<u>H</u> elp	
	_	-							

Рисунок 3.1 Строка главного меню графического редактора.

## Меню Файла

В меню файла вы можете выполнить действия по созданию новой модели, открыть, сохранить, сохранить как, устанавливать, печатать, и выходить.

<u>N</u> ew File	Ctrl+N	
Ōpen	Ctrl+O	
<u>M</u> erge		
<u>S</u> ave	Ctrl+S	
Save <u>A</u> s		
<u>I</u> mport	•	<u>B</u> itmaps
<u>E</u> xport	•	<u>G</u> ML scripts
Insta <u>l</u> I		Submodel <u>V</u> ariable List
Delete Model Files		
<u>P</u> rint	Ctrl+P	
Exit Application		
File		×
	d source former	
	U 🌇 🖼	🛅 🗷 🖉 🗶 📕

Рисунок 3.2 Меню файла и панель инструментов файла.

Панель инструментов файла имеет следующие кнопки от слева направо:

1) Кнопка создания новой модели



2) Кнопка открытия модели

Открывает новый файл.

3) Кнопка объединения





Объединяет указанную модель с рабочей моделью.

4) Кнопка «сохранить»



сохраняет модель под тем же самым названием {именем}.
5) Кнопка «сохранить как»

Сохраняет модель под указанным названием {именем}.

6) Кнопка импорта рисунка



7) Кнопка экспорта GML скриптов.

Экспорт GML скриптов.

8) Кнопка списка переменных подмоделей.



6.3

Экспорт списка переменных подмоделей.

9) Кнопка установки



Устанавливает глобальные модели

10) Кнопка удаления файла модели



Удаляет файлы модели.

11) Кнопка печати



12) Кнопка выхода



## Модели и субмодели

Модели, с которыми работает редактор, состоят из моделей и подмоделей. В основном, подмодели и модели используются как экраны. Есть местные и глобальные модели и подмодели. Глобальные модели состоят из пользовательских и главных. Используя открытый файл модели, вы можете выбрать локальную или глобальную модель или подмодель. Если вы выбрали глобальную модель, то можно выбрать еще пользовательскую или главную.

Локальный справочник указывает модели или подмодели, которые можно использовать только на компьютере, где работает редактор. Глобальная Модель - модель, которая может использоваться на всех OIS's в то время, когда на OIS запущен редактор.

Когда вы создаете и сохраняете модель или подмодель, это сохраняется в том же месте. То есть когда вы сохраняете модель, вы можете рассмотреть файл модели, который вы создавали только на OIS, где запущен редактор. На других OIS, Вы не можете увидеть то, что вы сохранили.

Функция меню файла устанавливает вызов. Сохраненная модель передается в папку "Local" всем OIS's в сети и перемещает модель из местной в глобальную. Когда вы сохраняете модель, созданную редактором, она сохраняется локально, так что вы можете проверить это в OIS, где запущен редактор. Если нет проблем, устанавливающиеся копии файла модели переме-щаются ко всем OIS из локальных в глобальные выбором «Install» вменю «файл». Из области «глобальных» модели перемещаются в пользовательскую область. Главная область, становится областью для стандартных экранов и основных подмоделей. Файлы в главной области могут быть открыты и используемы для подмо-делей. Однако, если вы изменяете и сохраняете их, они сохраняются в области «локальных». При перемещении только в глобальную пользовательскую область, модели защищены от неправильных действия.

Последовательность поиска моделей в OIS выполняется в следующем порядке.

- 1) Местная (локальная) модель
- 2) Местная подмодель
- 3) Глобальная пользовательская подмодель
- 4) Глобальная пользовательская подмодель
- 5) Глобальная главная подмодель
- 6) Глобальная главная подмодель

Если вы редактируете и сохраняете модели в папке пользовательских моделей, они сохранены в местных (user) моделях.

Так как они вверху последовательности поиска (вверху показаны измененные модели), в пользовательской модели нет оригинальных моделей.

#### Команда создания новой модели

Команда выбора новой модели удаляет текущую модель и содержание редактирующего вида графического редактора. Здесь значение атрибутов устанавливается повторно. Если текущее состояние текущей модели не сохранено, диалог новой модели запрашивает подтверждение на изменение данных (рисунок 3.3) Появляется запрос выбор [отмена] отменяющий (выбирающий) установку новой модели и возобновляет (отменяет) текущую модель. Отбор [ОК] инициализирует текущую модель.



Рисунок 3.3 Диалог подтверждения открытия новой модели.

#### Команда открытия новой модели

Когда вы убираете текущую модель и выбираете новую модель, открывается диалог читения следующей модели..



Рисунок 3.4 Диалог открытия новой модели.

Выберите [Глобальную Модель] ([Global Model])или [Местная Модель] ([Local Model]), используя кнопки-метки. Выберите [модель] или [Подмодель] для объекта, используя кнопки-метки.

Когда кнопки-метки выбрана, в блоке списка появляется соответствующие названия {имена} моделей.

Щелкните на названии {имени} модели в списке, и щелкните на кнопке <OK>, чтобы загрузить модель в «рабочий вид». Когда текущее состояние текущей модели не сохранено, открывается диалог подтверждения (Рисунок 3.5). Выбор [Отмена], отменяет открытие и возобновляет текущую модель. Выбор [OK] инициализирует текущую модель.



Рисунок 3.5 Диалог подтверждения

Вы можете также войти в фильтр названия {имени} модели, которое Вы хотите рассмотреть в списке нижеприведенного диалога.

Например, если вы вводите " \*irect \* " и нажимаете ключ вкладки, названия {имена} моделей показаны в блоке списка и строке " \*irect \* ", которая показывает доступные модели.

Если резервный блок проверен, вы можете открыть модели, которые поддержаны.

Open		×
File D1 GetsDemo2 D1 GraphMenu D1 GraphMenuButton2 D1 GraphMenuButton3 D1 Menu D1 MenuButton D1 PenTagi D1 SubGetsDemo1 D1 SubGetsDemo2 D2 GetsDemo2 D2 GraphMenuButton3 D2 GraphMenuButton3 D2 Menu D2 SubGetsDemo1 D2 SubGetsDemo1 D3 SubGetsDemo1 D4 SubGetsDemo1 D5 SubGetsDem01 D5 SubGetsDem01 D5 SubGetsDem01 D5 SubGet	Directory  Global(G)  Local(L)  Model  Submodel  Backup	OK Cancel
•		

Рисунок 3.6 Диалог открытия резервных моделей

В списке показаны резервные модели, которые автоматически создаются, при сохранении моделей. Число {Номер} перед каждой моделью (01,02...) указывает версию резервных. Меньший номер относится к более поздней модели.

Выберите любую модель и щелкните на кнопке <OK>, чтобы загрузить модель в рабочий вид.

#### Команда объединения

Эта команда добавляет части модели к текущей модели. Название, появляющееся в списке имён моделей, остается тем же самым. Объединенная модель все еще существует как файл модели, и копия его частей становится частью текущей модели.

Действия, для выбора файла те же самые как и для команды открытая модели.

#### Команда сохранения

Эта команда сохраняет рабочую текущую модель. Название {имя} - то же самое название {имя}, которое используется при загрузке модели командой открытия модели.

Если при запуске графического редактора никакая модель не загружена, появляется диалог, который спрашивает вас образцовое имя.

Когда вы открываете глобальную модель и сохраняете её, тот же самый диалог сохраняет её как локальную модель.

Save as		×
File	Directory	OK
	C Global( <u>G</u> )	
agitatormigi_kami almrev_bas	● Local(L)	
ButtonTest		Cancel
circle1	O Model	
CurV_AImCNF1_chm	<ul> <li>Submodel</li> </ul>	
ddrmigi_kami		
demo_circle demo_sensor1		
demo_sensor2		J
demo_vivi demo_viv2		1
demo_vlv3	🗖 Dackup	
•	1	
· ·		

Рисунок 3.7 Диалог «сохранить как».

Выберите или введите название {имя} файла для того, чтобы сохранить в локальном справочнике. SL-GMS принимает файл соответствующий названию модели. Выберите [модель] или [Подмодель], используя кнопки-метки, и щелкните кнопку <OK>, чтобы сохранить.

Символы, которые могут использоваться в названии модели - буквы, цифры, и подчеркивание.

Вы не можете использовать непечатные символы такие как черта, пробел, ESC ключ для названия {имени}модели. Вы не можете использовать то же самое название {имя} для «сохранения как» SL-GMS. Сохраненные слова SL-GMS внесены в список в Приложении 2, " Сохраненные SL-GMS слова" ("SL-GMS Reserved Words").

Если вы пробуете сохранить модель под названием, которое не может использоваться, появится диалог предупреждения "Сохраненные слова GML не могут быть определены как название {имя} модели".



Рисунок 3.8 диалог «внимание»

Если Вы пробуете сохранить модель под тем же самым названием {именем} как и модель, уже зарегистрированная,

Появляется диалог подтверждения перезаписи. Щелкните на кнопке [ОК], чтобы перезаписать модель. Щелкните на кнопке [отмена] для отмены сохранения.



Рисунок 3.9 Диалог подтверждения перезаписи.

**Команда «сохранить как»** Эта команда сохраняет модель под другим названием {именем}. Появляется диалог сохранения (Рисунок 3.7).

Определите название {имя} модели, и сохраните модель. Действия диалога - то же самое как и команда «сохранить».

#### Команда импорта точечного рисунка.

Команда [Файл] - [Импорт] - [Рисунок] ([File]-[Import]-[Bitmap]) читает файлы точечных рисунков и добавляет их в работу как область объекта.

Когда вы выбираете эту команду, появляется диалог, для выбора файла рисунка.



Рисунок 3.10 Диалог точечного рисунка

Коробка списка Диалога показывает названия {имена} bitmap файлов под BITMAP справочником в справочник определил, когда Графический Редактор был установлен.

Выберите bitmap файл, называют и щелкают на <хорошо> кнопка, чтобы закрыть диалог. Тогда, щелкните на

основание уехало из bitmap на рабочей области, чтобы определить местонахождение объекта {цели} bitmap на рабочей модели.

Если [Scalable] коробка чека {проверки} отобрана, признак, который автоматически измеряет объект {цель}

согласно изменяющемуся размеры из окна добавляется к расположенному объекту {цели} bitmap.

#### ССЫЛКА:

Формат точечного рисунка (bitmap), с которым может обращаться (работать) графический редактор - стандартный BMP формат MS-windows.

Чтобы загрузить точечный рисунок (bitmap) в графический Редактор, расширение файла должно быть изменено {заменено} на ".i".

Масштабирование вступает в силу, когда для отображения установлено 256 цветов.

OIS использует 65536 цветов, так что масштабирование не вступает в силу (не работает).

### Команда экспорта GML Script

Команда [Файл] - [Экспорт] - [GML Script] ([File]-[Export]-[GML Script]) генерирует GML Script рабочей модели как текстовый файл (.g файл).

Файлы модели в формате .g отличаются от файлов модели по обычному формату .m1, так как это текстовый формат. Вы можете проверить и изменить содержание «Script», если вы понимаете спецификации SLGML script.

Файлы модели в формате .g могут использоваться в различных компьютерах, которые имеют различную архитектуру. Конвертируйте {преобразуйте} .g файл в двойной файл, используя SL-GML или gm1 программу.

Save As		? >
Savejn:	🔄 Temp 📃 🧾	
image ibres_alogo mkai2 Ras iest Tritemp	🔁 Vbe ວກ 🖻 x.g	
File <u>n</u> ame:	xq	<u>S</u> ave
Save as <u>type</u> :	ASCII model file(*.g)	▼ Cancel

Когда вы выбираете эту команду, появляется диалог, для определения название .g файла.

Рисунок 3.11 Диалог сохранения GML Script.

Определите директорию (раздел) и название {имя} файла, для сохранения «SLGML script» и щелкните на кнопке <Save> (сохранить), чтобы создать файл «script» (.g файл). Если файл с тем же самым названием {именем} существует, появляется диалог подтверждения перезаписи (Рисунок 3.12).



Рисунок 3.12 Переписывает Диалог Подтверждения

Сохраняющие форматы моделей и подмоделей состоят из трех форматов; .m1.m2, и .g. Формат .m1 - стандартный формат для моделей и подмоделей. Формат .m2 - формат с высокой скоростью чтения, так как модели и подмодели оптимизированы.

Чтобы конвертировать {преобразовывать} модели или подмодели в этот формат, используйте команду, которая конвертирует {преобразовывает} файлы в формат .m1 и в формат .m2. Файл формата .g - ASCI, в который GML создаёт подмодель. Поэтому вы можете загрузить это в редактора текста.

Этот формат совместим в гетерогенной окружающей среде. Так файлы моделей могут быть импортированы/экспортированы между MS-WindowsNT и Unix. Однако, некоторые признаки не сохраняются. Примеры использования могут включать "проверку различия между подобными файлами моделей".

#### Команду экспорта списка переменных подмоделей.

Команда [Файл] - [Экспорт] - [Список переменных подмоделей] ([File]-[Export]-[Submodel Variable List]), показывает лист переменных подмоделей, расположених в рабочей модели.

Sul	omodel Mariable List			×
١	dodel name	variables	Values	
	demo_vlv1 demo_vlv2 demo_vlv3	LCOLOR LCOLOR TAGNO	1 2 "PID_0001"	
	Select All	Unselect All	Output to CSV file	ок

Рисунок 3.13 Диалог внесения в список переменных подмоделей.

Щелкните на кнопке <OK>, чтобы закрыть диалог.

Чтобы выбрать или отказаться от выбора пунктов, щелкните на пунктах в списке. Кнопка <Выбирают всё> (<Select All>), выбирает все переменные. Кнопка <отказ от выбора всего> (<Unselect All>) отменяет все выбранные пункты.

Чтобы сохранить информацию о выбранных пунктах в CSV файле, щелкните на кнопке <выход к CSV Файлу> (<Output to CSV File>).

Когда вы выбираете <выход к CSV Файлу> (<Output to CSV File>), появляется диалог экспорта списка переменный .csv файла (Рисунок 3.14).

Войдите в файл .csv , и выберите [сохранить], чтобы сохранить.csv файл.

Export a subr	nodel's renamed list to	a CSV file			?	×
Save in:	🔄 Temp	-	£	<b>e</b> *	<b>:::     </b>	
imoge ibres_alogo mkai2 Ras test Thlemp	ि Vbe on श्वि Arcconv.csv श्वि Hstmsg.csv					
File <u>n</u> ame: Save as <u>t</u> ype:	CSV Files (*.csv)				<u>S</u> ave Cancel	]

Рисунок 3.14 Диалог экспорта файла.csv списка переменных подмоделей

#### Команда установки.

Эта команда регистрирует локальную модель как глобальную модель. Когда вы выбираете эту команду, появляется установочный диалог

Install		×
File Name agitatormigi_kami almrev_bas Avalve_chm ButtonTest circle1 coolingfan CurV_AImCNF1_chm CurV_AImCNF3 ddmnigi_kami demo_sensor1 demo_sensor2 demo_skv1 demo_skv2 demo_vk3 demijbenhidari_kami denjibenhidari_kami	Directory  C Clocal(L)  Model  Submodel  C	OK Apply( <u>A</u> ) Cancel

Рисунок 3.15 Установочный диалог.

В этом диалоге, вы можете выбрать [Модель] или [Подмодель], чтобы установить как глобальную модель.

Выбрать [Применить] ([Apply]), чтобы установить выбранное имя модели или подмодели. Если вы выбираете [OK], диалог закрывается после того, как действия по установке закончатся. Если модель или подмодель с тем же самым названием {именем} была установлена, появляется диалог подтверждения перезаписи.



Рисунок 3.16 Диалог подтверждения перезаписи.

Если установка терпит неудачу, появляется диалог [копирование файла не удалось] ([Failed to copy the File]).

Возможная причина такого отказа {неудачи} состоит в том, что в базовом компьютере не был создан необходимый раздел (директория).



Рисунок 3.17 Диалог неудавшегося копирования файла

#### ИНСТРУКЦИЯ:

Команда установки делает глобальный эквивалент моделей среди множества OIS. Для того, чтобы сделать эквивалент модели среди множества OIS, вы должны выполнить команду установки.

#### Команда удаления файла модели.

Эта команда удаляет образцовые сохранённые файлы моделей. Когда вы выбираете эту команду, появляется диалог удаления выбранных файлов.

Delete model files		×
Delete model files File Name agitatormigi_kami almrev_bas Avalve_chm ButtonTest circle1 coolingfan CurV_AImCNF1_chm CurV_AImCNF3 ddrmigi_kami demo_circle demo_sensor1 demo_sensor2 demo_vlv3 demjibenhidari_kami denjibenue_kami	Directory  Global(G)  Local(L)  Model  Submodel  Backup	OK Apply( <u>A</u> ) Cancel
•		

Рисунок 3.18 Диалог удаления файла модели.

Когда вы выбираете раздел (директорию) и тип модели кнопками-метками, переданные модели отображаются в блоке списка. Выберите модель, которую вы хотите удалить из блока списка и щелкните на кнопке «применить» <Apply>, чтобы удалить файл модели. Щелкните на кнопке <OK>, чтобы закрыть диалог удаления файлов моделей.

#### Команда печати

Эта команда распечатывает модели находящиеся в рабочей области.

Когда вы выбираете [Печать], появляется стандартный диалог печати MS-windows. Выберите [ОК] для печати,

Print		? ×
- Printer		
<u>N</u> ame:	PRO	<u> P</u> ioperties
Status: Type: Where:	Ready EPSON MJ-930C LPT1:	
Comment:		🗖 Print to fije
Print range	,	Copies
• <u>A</u> I		Number of <u>c</u> opies: 1
C Pages	s from: 1 ta:	1 2 3 Collate
O <u>S</u> elec	tion	
		OK Cancel

Рисунок 3.19. Диалог печати.

#### Команда выхода

Эта команда закрывает графического редактора.

Когда вы выбираете эту команду, появляется диалог выхода. Выберите [OK] для выхода из графического редактор. Выберите [Отмена], если вы не хотите выходить из графического редактора.



Рисунок 3.20 Диалог выхода.

# Меню редактора.

Меню редактора, позволяет вам исполнить вращение, масштабирование, и выбрать объект списке выбора.



Рисунок 3.21 Меню редактора и панель инструментов редактора.

Панель инструментов редактора имеет следующие кнопки слева направо:

1) Кнопка отмены.

Отменяет последнее действие.

2) Кнопка удаления

Удаляет объекты.





Перемещает выбранный объект перед указанным объектом.



Перемещает выбранный объект в заднюю часть указанного объекта.

5) Кнопка «впереди всех»



Перемещает выбранный объект перед всеми объектами.

6) Кнопка «позади всех»

Перемещает выбранный объект в заднюю часть всех объектов.

7) Кнопка вращения.

Поворачивает объект под углом, который вы вводите.

8) Кнопка масштаба.

Изменяет объект в соответствии с масштабом, который вы вводите.

9) Кнопка горизонтального масштаба.



Изменяет объект в горизонтальном направлении. Ограничивает движение мыши в горизонтальном направлению, при рисовании/редактировании объекта.

10) Кнопка вертикального масштаба.



Изменяет объект в вертикальном направлении. Ограничивает движение мыши в вертикальном направлению, при рисовании/редактировании объекта.

11) Кнопка Unequal/Equal (равное/неравное соотношение).



Изменяет объект в равном или неравном отношении.



Позволяет вам ввести значение координат вместо щелчка мыши.

13) Кнопка «Выбрать всё».



ا ص

Выбираются все объекты.

14) Кнопка выбора по расширению.





Выборов объектов по расширению.

15) Кнопка выбора по имени.



Выборов объектов по названию объекта.





Располагает объекты послойно.

17) Кнопка выбора объектов.



Выбирает объект.

#### Команда отмены последнего действия.

Эта команда отменяет последнее действие. Если вы выбираете эту команду немедленно после какого-либо действия, состояние объекта возвращается к первоначальному (предыдущему) состоянию.

Примечание: Вы можете отменить только одно действие. Отмену действия необходимо производить непосредственно после действия.

#### Команда удаления.

Эта команда удаляет текущий объект, в списке выбора моделей. Если немедленно, после того как вы удалили некоторые объекты, использовать команду отмены предыдущего действия, можно отменить удаление.

## Меню последовательности.

Вы можете выбрать четыре команды из меню последовательности; [последовательность] - [передний план] ([Order]-[Front]), [последовательность] - [задний план] ([Order]-[Back]), [последовательность] - [впереди всех] ([Order]-[Front of All]), и [последовательность] - [позади всех] ([Order]-[Back of All]).

Эти команды управляют последовательностью объектов в списке части текущей модели. Команда выбора последовательности объектов изменяет последовательность их отображения в текущем окне.

Объекты добавляются к списку части модели в созданной последовательности. Первый объект в списке части модели показан первым в окне модели, а последний объект показан последним в окне модели. При просмотре объектов в окне модели, они перестраиваются в списке в соответствии с очередностью просмотра.

Команда [последовательность] - [передний план] ([Order]-[Front]).
 Эта команда перемещает объекты в списке выбора перед выбранным объектом.
 Объекты располагаются по первоначальному выбору, когда щелкают «[передний план]», тогда объект размещается в соответствии с выбором.

2) Команда [последовательность] - [задний план] ([Order]-[Back]). Эта команда перемещает объекты в списке выбора на задний план выбранного объекта. Объекты располагаются по первоначальному выбору, когда щелкают «[задний план]», тогда объект размещается в соответствии с выбором.

3) Команда [последовательность] - [впереди всех] ([Order]-[Front of All]), Эта команда перемещает объект, в списке выбора, перед всеми другими объектами в модели. то есть объект перемещается до конца списка части модели.

4) Команда [последовательность] - [позади всех] ([Order]-[Back of All]) Эта команда перемещает объект, в списке выбора, после всех других объектов в модели, то есть объекты перемещаются к началу списка части модели.

#### Команду вращения.

Эта команда вращает объект, в списке выбора, в соответствии с указанным углом вращения. За дальнейшей информацией обратитесь к разделу: «кнопка «Rotate» панели создания объекта.

#### Команда установки масштаба.

Эта команда изменяет объекты, в списке выбора, в соответствии с указанной величиной масштаба. За дальнейшей информацией обратитесь к разделу: «кнопка «Scale» панели создания объекта.

## Меню модификации.

Меню модификации определяет варианты изменения объекта в списке выбора. За дальнейшей информацией, обратитесь к разделу: «кнопка Scale, в панели создания объекта. Команды [Горизонтально] и [Вертикально] ограничивают перемещение мыши от предыдущего положения.

Например; когда вы рисуете горизонтальную линию, щелкаете на Кнопке <Line>, определяете точку начала линии объекта, щелкает мышью, и выбирает мышью команду [Редактор] - [модификация] - [горизонтально] ([Edit]-[Mode]-[Horizontal]), мышь перемещается только в горизонтальном направление. Далее вы определяете точку конца горизонтальной линии, до которой она должна быть проведена.

Точно так же, при перемещении объекта; выберите объект, щелкните на кнопке <Move> (переместить), и выберите команду [Редактор] - [модификация] - [Горизонтально] ([Edit]-[Mode]-[Horizontal]), чтобы ограничить перемещение объекта горизонтально. Если вы выбираете команду: [Редактор] - [модификация] - [неравное/равное отношение] ([Edit]-[Mode]-[Unequal/Equal Ratio]) ограничение отменяется.

#### Команда входа

Когда вы выбираете команду входа, вы можете ввести в координаты точки с клавиатуры, вместо того, чтобы выбирать координаты мышью. Когда вы выбираете эту команду, появляется диалог ввода координат точки.

Text Entry	×
Enter X and Y coords of point:	
Cancel	OK

Рисунок 3.22 Команда входа.

Дискретность редактирования координат в графическом редакторе - от (0,0) до (100,75) в единицах координат WCwindow.

Чтобы задать точку с клавиатуры, задайте два значения, разделенные пробелом, и нажмите <OK> кнопку.

Диапазон первого значения - от 0 до 100, а диапазон второго - от 0 до 75 в единицах координат WCwindow. Если используется кнопка <Zoom Out> или <Pan>, любые значения вне этих диапазонов не показываются, начиная со значений точки вне окна WC-window.

Вы можете вводить десятичные числа.

# Меню выбора

Меню выбора включает варианты выбора объектов.

1) Команда [Выбор] - [Всё] ([Select]-[All])

Эта команда выбирает все объекты в текущей модели.

2) Команда [Выбор] - [степень] [Select]-[By Extent].

Эта команда выбирает объекты, которые показаны полностью в пределах прямоугольника названного степенью (область выбора). Пользователь должен выбрать две точки в противоположных углах прямоугольника, чтобы полностью окружить(выбрать) желательные(необходимые) объекты. Так выбираются даже неотображаемые объекты. Любые выбранные объекты добавляются к существующему списку выбора.

3) Команда [Выбор] - [По имени] ([Select]-[By Name]).

Эта команда выбирает объекты по имени. Когда вы выполняете эту команду, появляется диалог ввода названия объекта.

Text Entry	×
Enter Object Name:	
Cancel	OK

Рисунок 3.23 Диалог ввода.

Введите название {имя} объекта, которое вы хотите выбрать и щелкните на кнопке <OK>, чтобы выбрать объект, имя которого вы ввели.

4) Команда [Выбор] - [Убывает] ([Select]-[Descend])

Эта команда разрешает выбор объектов в пределах групп.

При каждом выборе число {номер} убывает. В круглых скобках при выборе: статус указывает, как глубоко в пределах групп объект может быть выбран (уровень не ниже - 0).

Уровень 1 разрешает выбирать объекты в пределах групп, уровень 2 позволяет выбирать объекты в подгруппе группы, и так далее.

Щелчок кнопкой «выбор» уменьшает глубину выбора.

Эта опция часто используется для изменения привязки в пределах группы. Опция не изменяет признаки местоположения объекта в модели (то есть, не разрешается действия, которые изменяют матрицу преобразования объекта).

5) Команда [Выбор] - [модель].

Эта команда непосредственно выбирает объекты рабочей модели.

# Меню «вид» (View)

В меню «вид», вы можете изменить вид для рабочей области, и показать/скрыть управление для панели инструментов.

Tool bars * Status Bar Save Toolbar Status Get Toolbar Status Eeset View Regram F5	<ul> <li>Object Qreation Tool</li> <li>Eile Tool</li> <li>Edit Tool</li> <li>Miew Tool</li> <li>Object Tool</li> <li>Object Tool</li> <li>Option Tool</li> <li>Point Tool</li> <li>Palette Tool</li> <li>Help Tool</li> </ul>	
Yiew 🛞 🖸 🔛 🍬 🗞 RD		

Рисунок 3.24 Меню «вид» и вид панели инструментов.

Панель инструментов имеет следующие кнопки слева направо:



Изменяет вид.



Увеличивает масштаб изображения на представлении {виде}.

3) Кнопка «уменьшить масштаб изображения».



Уменьшает масштаб изображения на представлении {виде}.

4) Кнопка «увеличение».



Увеличивает представление {вид} указанной области.

5) Кнопка «перенабора» (изменения) вида.



Повторно устанавливает исходный вид.





Изменяет рабочее представление {вид}.



7) Кнопка вызова панели инструментов для создания объекта.

Показывает/скрывает строку инструментов для создания объекта.



8) Кнопка вызова панели инструментов для создания файла.



Показывает/скрывает строку инструментов для создания файла.

9) Кнопку инструмента редакции.

Показывает/скрывает панель инструментов для редактирования.

10) Кнопка панели инструмента Представления {Вида}.



Показывает/скрывает панель инструментов для представления {вида}.

11) Кнопка вызова панели инструмента объекта.



Показывает/скрывает панель инструментов для объекта.



12) Кнопка панели инструмента создания динамики.

Показывает/скрывает панель инструментов для создания динамики.

#### 13) Кнопка инструмента опций.



Показывает/скрывает панель инструментов опций.



14) Кнопка панели инструмента палитры.

Показывает/скрывает панель инструментов для создания (изменения) палитры.



Показывает/скрывает панель инструментов для точки.



Показывает/скрывает панель для помощи.

17) Кнопка вызова «линейки» статуса.



Показывает/скрывает «линейку» статуса.

#### 18) Кнопка сохранения статуса.



Сохраняет статус панели инструментов в регистре.





Читает статус панели инструментов из регистра.

#### Команда панорама (форма).

Функция «форма» перемещает все объекты модели в любом направлению. Эта функция позволяет вам измените область отображения модели в рабочем представлении (виде). Изменение через эту команду – не постоянное и не сохраняется наряду с моделью. Обработка панорамы выполнена перемещением окна WC-window, относящегося к модели. Это то же самое как перемещение вверх и вниз и слева направо, положения модели окна WC-window.

Вы можете создавать или изменять объекты в модели после того, как вы смещаете панораму. Вы можете ввести в точку для центра нового Окна WC, используя команду [Редактор] - [Ввод].

#### Действие кнопки панорама.

Когда вы нажимаете кнопку <Pan> (панорама), вы получаете возможность выбора нового центра для точки. Щелкнуть на точке в рабочем представлении {виде}. Модель повторно появится в

новом окне WC. Щелкните на кнопке < перенабор вида> (<Reset View>), чтобы отменить панораму.

#### Команда увеличения масштаба изображения.

Увеличением масштаба изображения, в ближнем Окно WC для того, чтобы объекты в рабочем представлении {виде} казались большими. После увеличения масштаба видимы только объекты в середине рабочего вида. Опция [Перенабор вида] отменяет изменение масштаба изображения. Масштаб изображения возвращается к исходному.

Величина изменения масштаба изображения автоматически устанавливается графическим редактором. Изменения через «увеличить масштаб изображения» - не постоянные и не сохраняются вместе с моделью. Вы можете создавать или изменять объекты в модели после того, как вы увеличиваете масштаб изображения.

В зависимости от текста и толщины линии вы определяете увеличение масштаба изображения не, если аппаратные средства ЭВМ не могут приспособиться к изменениям размера.

#### Команда уменьшения масштаба изображения.

Пошаговое уменьшение масштаба изображения окна WC производится так, чтобы появились все необходимые объекты в рабочем представлении {виде} . Опция [Перенабор Вида] ([Reset View]) отменяет изменявшийся масштаб изображения или возвращает масштаб изображения до исходного.

Величина изменения масштаба изображения автоматически определена графическим редактором. Уменьшение масштаба изображения не постоянно и не сохраняется вместе с моделью. Вы можете создать, или изменить объект в модели после того, как вы уменьшите масштаб изображения. В зависимости от текста и толщины линии вы определяете, уменьшение масштаб изображения, если аппаратные средства ЭВМ не могут приспособиться к изменениям размера.

#### Команды изменения масштаба изображения.

Эта команда изменяет масштаб изображения в выделенной области в рабочем представлении {виде}. Для быстрого изменяет масштаб изображения, пользователь выбирает два угла прямоугольника, и щелкает двумя противопоставлениями вокруг области, чтобы изменить масштаб изображения.

Графический редактор регулирует {приспосабливает} прямоугольник так, чтобы выделенное изображение полностью поместилось в рабочем представлении {виде}. Регулирование предотвращает искажение объектов, которые произошли бы если аспект выбранного прямоугольника был маленьким.

Изменение, через «Изменить масштаб изображения» не постоянно и не сохраняется вместе с моделью. Вы можете создавать или изменять объекты в модели после того, как изменили масштаб изображения.

В зависимости от текста и толщины линии вы определяете, величину масштаба изображения, если аппаратные средства ЭВМ не могут приспособиться к изменениям размера.

## Команда перенабора (обновления) представления (вида).

Эта команда отменяет, любое изменение масштаба изображения или панорамы. Рабочее представление {вид} возвращается к его стартовому (начальному) положению. По умолчанию, величина Рабочего Представления {Вида} –противоположные углы (0,0) и (100,75).

#### 20) Команда изменения (перерисовки).

Эта команда изменяет (перерисовывает) рабочее представление {вид}.

## Меню панели инструментов.

Контрольное меню панели инструментов показывает/скрывает строки (линейки) статуса и панели инструментов. Кнопка соответствующая показанной строке инструментов, находится в выдвинутом состоянии.

Меню панели инструментов имеет следующие команды, которые открывают соответствующие панели инструментов:



1) Инструменты создания объекта.

Открывает панель инструментов для создания объектов.

2) Инструменты создания файла.



Открывает панель инструментов для созданияфайла.





Открывает панель инструментов для редакции объектов и файлов.

4) Инструменты Представления {Вида}





Открывает панель инструментов для изменения рабочего вида.

5) Инструменты объекта.



Открывает панель инструментов для открытия объекта и работы с ним.

6) Инструменты динамики.

Открывает панель инструментов для создания динамики объекта.





Открывает панель инструментов для изменения опций.

#### 8) Инструменты палитр.



Открывает панель инструментов для создания и изменения палитры объекта.



Открывает панель инструментов для работы с точкой.



Открывает панель помощи (подсказок)

11) Инструменты статуса.

Открывает панель инструментов для изменения статуса.

#### Панель сохранения статуса.

Эта команда сохраняет статус панели инструментов и строки статуса (показать/скрыть и положение) в регистре. Когда графический редактор запускается повторно, открывается предыдущая панель инструмента и строка статуса.



#### Получение статуса.

Эта команда читает сохраненный статус и строку инструментов из регистра и перемещает строку статуса и панель инструментов в рабочий вид.



## Меню объекта.

Меню объекта используется при работе с группой из множества объектов, или изменения признаков различных объектов.

Group Fill Group ©omplex Fill Group ∐ngroup	
Attributes	<u>G</u> rephica Object <u>N</u> ame <u>Text</u> Object <u>I</u> nfo
Explode Apply Xierm Berrove Xform Dump Set List.	
Objects	× ••••••••••••••••••••••••••••••••••••

Рисунок 3.25 Меню объекта и панель инструментов объекта.

Панель инструментов объекта имеет следующие кнопки от слева направо:



**1** 

Группирует отобранные объекты.



Заполняет открытые объекты.

3) Кнопка комплексного заполнения группы.

Заполняет объекты со сложной границей.

4) Кнопка разгруппирования.



Разгруппировывает сгруппированные объекты.

5) Кнопка признаков.

Открывает диалог графических признаков.

6) Кнопка названия {имени} Объекта.



Устанавливает название {имя} объекта.



7) Кнопка текста.

Изменяет {Заменяет} текст.

8) Кнопка информации об объекте.

Показывает признаки объекта.

9) Кнопка Vis (Показ).

Включает атрибуты отображения.

10) Кнопка Vis Off (Скрыть).

Выключает показ.

Ó



Включает атрибуты выбора.

12) Кнопка «выбор выключить.

Отключает атрибуты выбора.



разрывает подмодель.

14) Кнопка «применить преобразование»



Применяется преобразование.

15) Кнопку «отменить преобразование».

Отменяет преобразование.

16) Кнопку объединение списка.



Сваливает отобранную информацию объекта.

#### Команда группирования.

Эта команда создает групповые объекты (группирует). Когда необходимые объекты отобраны, тогда выбирается опция группировать.

Число объектов в списке выбора изменяется на 1, указывая что выбранные объекты стали единственным {отдельным} объектом. Группа - собрание объектов. Любой объект, который является частью модели, может также быть частью группы. Члены группы действуют как один объект. Например, полная группа может двигаться, изменять цвет, исчезать, или быть заполненной.

Различные типы группы заполняются различными способами, потому что их границы имеют различный интерьер и внешность, а также могут быть разделены прямыми. Любой графический примитив может использоваться как элемент в группы.

#### Команда заполнения группы.

Эта команда подобна, команде группирования за исключением того, что графический редактор соединяет выбранный открытый объект с невидимой границей так, что они могут быть заполнены как один объект.

#### Команда заполнения комплексной группы.

Эта команда подобна, команде группирования за исключением того, что графический редактор создает групповой объект, который может быть заполнен относительно многоуровневых границ (типа "островов" или "отверстие").

#### Команда разгруппирования.

Эта команда расторгает {pacnyckaeт} группу. Щелкают по группе, и выбирают опцию paзгруппировать (ungroup). В списке объектов появится столько объектов, сколько было в группе.

# Меню Признаков

В меню признаков, вы можете выбрать графическую команду, команду названия{имени} объекта, текстовую команду, и команду получения информации об изменении признаков объектов.

#### Графическая команда

Графическая команда открывает диалог графических признаков. За дальнейшей информацией обратитесь к описанию диалога графических признаков в главе 1.

#### Команда названия {Имени} Объекта.

Выбирая это меню, вы можете назвать объект. Название{Имя} – комбинация символов введенных с клавиатуры и используемых для выбора объекта.

Команда названия {имени} объекта используется для каждого индивидуального объекта в списке выбора. Прикладная программа может использовать название{имя}, чтобы определить индивидуальное местонахождение объекта в модели.

Чтобы назвать объект; выберите объект, после выбора команды названия{имени} объекта. Появится диалог ввода имени.

📆 Object name Entry		×
Enter Object Name		
1		
Apply	Reset	Close

Рисунок 3.26 Диалог ввода названия {имени} объекта.

После открытия диалога, введите новое имя объекта или отредактируйте существующее. Щелкните на кнопке <Применить> (<Apply>), чтобы назначить название {имя} для объекта и закрыть диалог.

Чтобы повторно показывать первоначальное название объекта, щелкните на кнопке <Reset>. Чтобы отменить новое, или отредактированное, название объекта название, щелкните на кнопке <Close>.

Название {имя} - любая комбинация символов букв, чисел {цифр}, подчеркиваний, и долларового признака {знака}. Имя должно начаться с буквы. Все объекты, используемые в графическом редакторе должны иметь название.

Имена – составные части различных моделей, так что вы не можете назначить одно и то же имя для различных объектов.

Вы не можете использовать базовые слова SL-GMS, внесенные в список в приложении "SL-GMS сохраненные слова", как имя.

Если вы вводите имя, которое уже существует или - одно из базовых слов SL-GMS, Появляется следующий диалог, побуждающий вас ввести в другое имя.



Рисунок 3.27 Диалог неправильного имени объекта.

#### Команда ввода текста.

Команда ввода текста изменяет содержание текста в текстовом объекте или заполненном прямоугольнике с текстом.

Когда вы выбираете эту команду, после выбора текстового объекта, появляется диалог для редактирования текста.

Edit Text		×
Edit Text:		
abcdefg		
Apply	Reset	Cancel

Рисунок 3.28 Диалог редактирования текста

Введите в текст, и щелкните на кнопке <Apply>, чтобы изменить содержание текстового объекта. Чтобы повторно показывать первоначальный текст, щелкните на кнопке <Reset>. Для отмены, редактируют и закрывают диалог, щелкните на кнопке <Cancel>.

## Команда вывода информации об объекте.

Команда вывода информации об объекте показывает список признаков выбранного объекта. Когда выбраны сложные объекты, эта команда не может быть выполнена. Когда выбрана Подмодель, отображается название подмодели и название объекта.

Display Configuration Into.	×
Object type: G_Spline_30	
Object name:	
Fill direction:0	
Fill type:1	
Fill pattern:1	
Fill percent: 100.00	
Fill attribute:1	
Fill color:9	
Line style:1	
Line width: 1.00	
Line colar.7	
Text alignment X:0	
Text alignment Y:0	
Text type:0	
Text direction:0	
Text font:0	
Text font name:	
Text size X:0	
Text size Y:0	
Text color:0	
	ok 1

Рисунок 3.29 Диалог вывода информации о конфигурации объекта.



Рисунок 3.30 Диалог вывода информации о конфигурации объекта.

## «Vis/Vis Off». Команда изменения атрибутов отображения.

«Vis/Vis Off» команда отображает и изменяет (удаляет) признаки отображения объектов в списке выбора. Вы не можете выбрать «Vis Off» объекта, щелчком на редакторе вида. Вы можете выбрать изменение атрибутов использованием [Редактор] - [Выбор] - [последовательность] или [Редактор]-[Выбор] - [По имени] ([Edit]-[Select]-[Descend] от [Edit]-[Select]-[By Name]).

Пример:

Чтобы создать подмодель, чьи изменения формы объекта зависят от статуса, создайте необходимые объекты заранее и сделайте один из них видимым.

В окружающей среде во время исполнения, только один из согласованных объектов виден (отображается), согласно динамике.

При присоединении подмодели к модели, выберите [Vis Off], чтобы облегчить действие.



Рисунок 3.31 Подмодель, чьё изменение формы зависит от статуса.

## Команда «выбор включить/выбор выключить»

Команда: «выбор включить/выбор выключить» изменяет признак выбора объектов в списке выбора.

Признак выбора управляет возможностью выбора объекта щелчком мыши. При выборе «выбор выключить» объект может быть отображён.. Вы не можете выбрать объект щелчком мыши,при «выбор выключить» (Detect/Off).

Однако, вы можете выбрать объект используя [Редактор] - [Выбор] - [по убывающей] или [Редактор] - [Выбор] - [По имени] ([Edit]-[Select]-[Descend] or [Edit]-[Select]-[By Name]).

#### Пример:

Вы должны установить фоновый цвет объекта при: «выбор выключить». Это предотвращает объекты от того, чтобы быть скрытым из-за случайного щелчка на фоновом цвета.

## Команда разрыва (разгруппировка)

Эта команда разрывает подмодель на копии ее частей. Когда модель используется как подмодель, объект относится к подмодели и устанавливает своё положение в очерёдности моделей.

Сложные объекты могут обратиться к той же самой подмодели. Разрыв подмодели делает копии ее частей и добавляет эти части к текущей модели, удаляя экземпляр объекта. Эти копии больше не связаны с подмоделью и могут быть изменены.

## Применение команды «Xform»

Эта команда применяет преобразование к объектам в списке выбора. Преобразование перемещение {шаг}, вращение, масштабирования, зеркализация, или любая комбинация этих действий.

Преобразования создают специальный объект названный матрицей преобразования, которая показывает преобразованный объект в его новом размере или ориентации.

Когда преобразование применено точки, используемые для ориентации объекта, и матрица преобразования удаляются.

Удаление матрицы преобразования желательно, потому что это ускоряет отображение объектов и уменьшает память, требуемую для их хранения.

Для некоторых объектов, применение, «Хform» не имеет никакого эффекта.

Графический редактор автоматически определяет, может ли быть применено преобразование к точкам объекта.

Например: при вращении прямоугольника преобразование «Xform» не применимо к точкам прямоугольника. Прямоугольники определены двумя противоположными углами и применение преобразования к этим точкам заканчиваются различными прямоугольниками, а не вращением прямоугольника.

Точно так же, если круг неравно масштабирован (изменён) для получения эллипса, преобразование не может быть применено.

Круг определен его центром и точкой на его окружности, и применение преобразования было бы в различных круге, не имеющих форму эллипса.

Пример:

Создайте прямоугольник (0,0) - (10,10), и переместите его в (50,50).

В этом состоянии, величина перемещения {Шага} (50,50) добавляется к (0,0) - (10,10) объекта прямоугольника.

Установка преобразования для этого объекта изменяет прямоугольник на (50,50) - (60,60). Поэтому, ненужные внутренние процессы могут быть опущены так, чтобы ускорить отображение и увеличить память.

#### ССЫЛКА:

Во время работы OIS, оптимизация, включающая, "преобразование Xform" применяется к полной, законченной модели. Поэтому, вам нет необходимости применять команду "Xform". Выполнение "Применяется {Обращается}, Xform" ускоряет скорость показа действий. Мы рекомендуем Вам к

выполните это. Также обратитесь {отнеситесь} к, "Как Создать Плоскую модель".

#### Команда удаления преобразования.

Эта команда удаляет преобразование объектов из списка выбора. Преобразование – это: перемещение {шаг}, вращение, масштабирование, зеркализация, или любая комбинация этих действий.

Преобразования создают специальный объект, названный матрицей преобразования, которая показывает преобразованный объект в его новом размере или ориентации.

Когда преобразование удалено, объект стерт и повторно вызван в его первоначальном местоположении (виде), как было до применения преобразования.

#### Команду вызова списка.

Команда вызова списка показывает структуру объектов в списке выбора в формате «дамп».



Рисунок 3.32 Пример выполнения команды вызова списка.

## Меню Динамики

В меню «динамики» вы исполняете все действия, связанные с динамикой объекта (динамические признаки). Это меню используется, чтобы войти, редактировать, показывать, и изменять название переменных.

Для детального изучения того, как использовать (создавать) динамику каждого объекта, обратитесь к главе 4, "Динамика".



Рисунок 3.33 Меню динамики и панель инструментов динамики.

Панель инструментов динамики имеет следующие кнопки от слева направо.

1) Кнопка редактирования динамики.

ଷ୍ଠା

Редактирует динамические признаки.



2) Кнопка переименования переменных.

Изменение названия переменной.

3) Кнопка отображения переменных.

Отображение названия переменной.



Назначает общественные переменные на переменные.

#### ССЫЛКА:

Для понимания образцов динамики, обратитесь к главе 4, "Динамика".

#### Команду редактора динамики.

Эта команда редактирует динамические описания, связанные с объектами в списке выбора. По этой команде всплывает окно «DYNAMIC PROPERTY» (динамические свойства), которое содержит список действий для создания/изменения динамических описаний объекта.

Edit Dyn		×
Enter Dynamics		A
= 1 call DefVar(&_pVar call DefVar(&_pPnt _pVar	,\$TAGNO\$#dPV, 1, 1, -1, 1, ,\$TAGNO\$#dDCML, 1, 4, 1,	1) 1, 1)
= * call NaNX(_p∀ar, _)	oPnt)	
र		V
OK	Reset	Cancel

Рисунок 3.34 Диалог редактора динамики.

Если объект отобран для установки «Динамики» на {цель}, в списке редактора текста отображается «динамика».

Для объектов, в которых не назначено движение, ничего не отображается в области редакттора. Ведите «динамику» в списке текстового редактора, или отредактируйте существующую динамику.

Нажмите ввод для включения каждой строки результата в соответствующую заявку на описание динамики. Если описания не действительны, появится окно сообщения для изменения описания динамики.

hput erro	r (s) of dynamics
⚠	(_INIT C= 1 ( call DefVer(&_pVar, \$TAGNO\$#dPV, 1, 1, -1, 1, 10) ( call DefVer(&_pPnt, \$TAGNO\$#dDOML, 1, 4, 1, 1, 10) ) ) (pVar C= + ( call NaN2(pVar, _pPnt) ) (=) )
	[OK]

Рисунок 3.35 Окно сообщения ошибки ввода динамики.

Щелкните на кнопке <OK>, чтобы закрыть диалог редактора. Для отказа от отредактированных описаний нажмите кнопку <Cancel>. Чтобы повторно установить содержание начального диалога, щелкните на кнопке <Перенабор> (<Reset>).

#### Команда переименования переменных.

Эта команда переименовывает переменные, связанные с динамическими свойствами Подмоделей. Команда вызывает диалог «ПЕРЕИМЕНОВАНИЕ ПЕРЕМЕННЫХ» (RENAMED VARIABLES), который содержит блок «прокрутки», чтобы переименовать переменные связанные с образцами Подмоделей в списке выбора.

R	Rename Vars.				
	Model Variables	Insert variables			
	LCOLOR	2			
	SIR	FIC			
	LCOLOR	2			
	🔽 Add "" when dropping				
	ОК	Cancel Set Root			

Рисунок 3.36 т Диалог переименования переменных.

Выберите из переменных, показанных в списке диалога, и введите значение, для переименования

переменной в текстовом блоке.

Чтобы подтвердить переименование, нажмите кнопку <Apply>. Чтобы повторно установить первоначальную переменную, щелкните на кнопке (<Reset>) <Перенабор>. Чтобы подтвердить информацию переименования и закрывать диалог, щелкните на кнопке <OK>. Чтобы отказаться от переименования информации и закрыть диалог, щелкните на кнопке <Cancel>.

Вы можете вызвать и уменьшить последовательность, при переименования от другого приложения. При переименовании, необходимое имя, снизу последовательности списка. Добавляющееся переименование автоматически записывается вниз последовательности списка.

#### Команда отображения переменных.

Эта команда показывает название {имя} переменных, которые относятся к выбранному объекту. Название {имя} переменных отображается списке диалога "отображение переменных".

List Information	×
INIT _pVar \$TAGNO\$#dPV _pPnt \$TAGNO\$#dDCML _pType \$TAGNO\$#dTYPE _pLock \$TAGNO\$#dLOCK _pArea \$TAGNO\$#dAREA _pAtyp \$TAGNO\$#dAREA _pAtyp \$TAGNO\$#dALIT _pAlit \$TAGNO\$#dALIT _pStit \$TAGNO\$#dSTIT _pFlit \$TAGNO\$#dFLIT	
	ОК

Рисунок 3.37 Пример выполнения команды показа переменных.

#### Команда назначения переменных.

Эта команда назначает общую переменную для переменных подмодели и устанавливает связь между этой переменной и динамическими моделями и подмоделями. Это назначение автоматически связывает недавно добавленные Подмодели и приложение. Эта команда действительна если описание динамики подмодели имеет переменную, которая начинается с "\_p". Выбор этой команды открывает диалог назначения переменных, и показывает имя переменной, для назначения, когда имя переменной выбранной подмодели начинается с " p".

#### Следующее – процедура:

Как правило, загружается подмодель, для назначения переменной в рабочую область командой открытия подмоделей.

Open		×
File Blower1	Directory	DK
Alm3Sts1Pump almrev	C Local(L)	
almrev_bas almrev_prt And2Pump	🔿 User mode	Cancel
Arrow1 Arrow2 AtomCDataPutBtn	C User submodel	
AtomDataSetBtn Bencury1 Blouce1	Basic model     Basic cubmodel	
Blower2 Blower3		
Boiler3D CircleTank	🗖 Backup	
•		

Рисунок 3.38 Выбор основной подмодели.

После загрузки подмодели в рабочую область, выберите команду назначения переменных. Диалог показывает переменную, которая будет назначена.

Variable Asizignment				×
Private	Public			
_pDI				
		Edit	Cancel	OK

Рисунок 3.39 Команда назначения переменных.

Выберите переменные из списка, для назначения и выбора типа. После выбора переменных, щелкните на кнопке <Edit>, чтобы показать следующий диалог.

Edit Variable Assignm	ent		×
PrivateP Public	DI		
_Туре			
Real	O Intege	o si	tring
	Other		
Size 1	_		
Collect			
Norma	O Initial		OK
	Other		Cancel

Рисунок 3.40 Диалог редактирования назначенных переменных.

Объявление пользовательских переменных осуществляется в этом диалоге.

Внутренние переменные выбранной Подмодели отображены в частном текстовом поле. Введите переменные, чтобы назначить (пользовательские переменные) в общем текстовом поле. Когда каждая внутренняя переменная связанная с пользовательской переменной, " \$ [Строка] \$" используется, чтобы конвертировать {преобразовать} себя к имени переменной структуры прикладных переменных. Переменная связывает внутреннее множество переменные с пользовательской переменной.

Когда подмодель проиллюстрируется к модели или Подмодели, "\$ [Строка] \$" становится адресатом для расширения названия {имени}. "#d", как внутреннее выражение

из """ ".". #n", используется как индекс, который указывает элемент массива, где n - номер. Вы можете ввести строку, включенную в " ".

Вы можете определить "Реальный", "Целочисленный", или "Строка" - тип набора переменной. Вы можете установить размер элемента для массива переменных.

Установка "Collection Type" отражает синхронизацию данных переменной, назначенной на модель, и определяет, используется эта подмодель однажды, при запуске модели, или выполняется циклически.

Когда Вы заканчиваете назначение переменных и сохраняете подмодель, назначеная информация автоматически добавляется к структуре подмодели.

```
mtran0
vis 1
detect 1
TagCommon345: model
. dynprop \
( INIT \
```

```
( = 1 \
(call DefV2(&_pAMON, $TAG$#dAMON)) \
(call DefVar(&_pATYP, $TYPE$#dATYPE, 1,4,1,1,1)) \
(call DefVar(&_pLOCK, $LOCK$#dLOCK1, 1,4,1,1,1)) ))
. userdata "%#uver%#=6"
```

Затем, проинсталлируйте эту подмодель к модели, и переименуйте назначенную переменную.

Rename Vars.		×
Model Variables	Insert variables	
LOCK TAG TYPE		
LOCK		
🔽 Add ''' when drapping		
ок	Cancel Set Rest	

Рисунок 3.41 Диалог переименования переменных.

Когда вы переименовываете назначенную переменную, используя команду «назначенная переменная», " \$ {Переменная} \$" генерируется замена имени переменной, которое используется в приложении.

"\$ [Строка], \$" становится адресатом для изменяющегося имени, и вы может установить каждую индивидуальную часть.

Если там имеется больше одной общей переменной с 1 моделью, и они совместно используют ту же самую "\$ [Строка] \$ ", и обрабатываются как один элемент.

Если общая переменная и переименовывающийся адресат не переписываются, то изменяется расширение только у одного из них. С тех пор как замененные элементы объединены и внедрены, переименовывающая информация каждого элемента теряется. Поэтому, в пользовательской базе данных изменённый элемент запоминается как тот же элемент (в том же месте базы данных).

#### ССЫЛКА:

Для изучения механизма назначения переменных, обратитесь к главе 4, "динамика".

# Меню Вариантов

В меню опций, вы можете определить, фиксацию сетки и точек.



Рисунок 3.42 Меню опций и инструментальная панель опций.

Инструментальная панель опций имеет следующие кнопки слева направо.

1) Переключатель сетки.



Открывает диалог сетки.

2) Кнопка отключения фиксации.



Отменяет режим фиксации.

3) Кнопка быстрой фиксации к сетке.



🔤 фиксация точки к сетке.

4) Кнопка фиксации к объекту.

Фиксация точки к объекту.

#### Опции сетки

Опции сетки открывают диалог, управляющий сеткой, для отображения в рабочем представлении {виде}.

Grid				×
Size(X): 2	Attributes	Style C Dot C Solid	0       1       2       3       4       5       6         8       9       10       11       12       13       14       15         16       17       18       19       20       21       22       23         24       25       26       27       28       29       36       31	

Рисунок 3.43 Диалог управления сеткой.

" Размер (X) " и "размер (Y) " указывают размер сетки. Для изменения формы сетки, «щелкните» на этих значениях, и введите новые значения с клавиатуры.

Если вы нажимаете на флажок Display Grid (показать сетку), сетка переключается между, отображать/скрыть.

Флажок "отображение впереди всего" управляет, отображением сетки перед всеми объектами. Когда модель имеет много заполненных объектов, это действие делает сетку лучше видимой.

## Команда фиксации.

Эта команда запускает режим фиксации так, чтобы вы могли вызывать или перемещать объект на любую указанную позицию.

#### Команда фиксации к сетке.

Команда фиксации к сетке вынуждает каждую выбранную точку, быть перемещенной к самому близкому пересечению сетки.

Команда фиксации, независимо от того, видима или нет сетка, всегда работает с текущим размером сетки.

#### Команда фиксации к объекту.

Когда задана команда фиксации к объекту, «щелчёк» точки около объекта {на объекте} перемещает выбранную точку на объект.

Например: если выбран угол прямоугольника, точка, в углу, используется при создании объекта.

Подсказка при быстрой фиксации к объекту разъясняет соединение объекта и точки. Включение кнопки фиксации к объекту позволяет осуществить выбор точки, для фиксации к самой близкой, предопределенной точке объекта.

# Меню Палитр

Меню палитр применяется, для использования палитр и подмоделей.



Рисунка 3.44 Меню палитр и инструментальная панель палитр.

Инструментальная панель палитр имеет следующие кнопки слева направо.



ф.

Выбор палитры.

2) Кнопка вызова подмоделей.



Выбор подмоделей.

#### Что такое - Палитра?

Палитра - набор подмоделей.

Когда вы открываете палитру, «нажмите» на подмодель в палитре, и щелчке на редакторе вида. Подмодель будет добавлена к модели, которую вы редактируете.

Создавая палитру с часто используемыми подмоделями, вы можете легко читать подмодели не используя диалог.


Рисунок 3.45 Пример палитры.

## Команда получения палитр.

Команда «получить палитру» (Get Palettes) используется, чтобы выбрать элементы модели и отобразить список палитры.

Когда Вы выбираете команду «получить палитру», появляется диалог получения палитр. Выберите имя файла и щелкните на кнопке <OK>, чтобы отобразить окно палитры в графическом редакторе. Вы можете отобразить множество палитр одновременно.



Рисунок 3.46 Диалог получения палитр.

## Команда получения подмоделей.

Команда получения подмоделей (Get Submodels) вставляет образец подмодели в текущую модель. Когда вы выбираете эту команду, появляется диалог выбора подмодели, которую вы хотите вставить.

Get aubmodela		×
File	Directory  Global(G)  C Local(L)	0K
almrev_bas almrev_prt And2Pump Arrow1 Arrow2	C User mode	Cancel
AtomCDataPutBtn AtomDataSetBtn Bencury1 Blower1 Blower2	C Basic node Basic submodel	
Blower3 Boilar3D CircleTank	E Backup	

Рисунок 3.47 Диалог получения подмоделей.

Этот диалог отображает список доступных подмоделей. Нажмите на название {имя} в списке, или введите название {имя} подмодели с клавиатуры. Когда вы нажимаете кнопку <OK>, чтобы вставить выбранную подмодель в рабочую модель, появляется прямоугольник указывающий размер подмодели в рабочей области.

Переместите этот прямоугольник, в желаемую позицию, с помощью мыши, и щелкните его, чтобы присоединить подмодель к рабочей модели.

## ССЫЛКА:

Когда вы установите подмодель, позиция курсора мыши в рабочем виде соответствует началу координат (0,0) первоначальной модели. Поэтому, для примера, внешняя созданная подмодель, использует другую контрольную точку, отличную от начала координат (0,0), вы не можете использовать мышь, для установки подмодели, так как перемещение мыши, ограничивает адрес. В этом случае используйте команду [редактор] - [ввод], чтобы ввести адрес

с клавиатуры.

## Как создавать Палитру

Палитра создается в следующих шагах:

- 1) Создать модель, которая имеет больше чем одну подмодель.
- 2) Сохранить модель.
- 3) Копировать файл модели в каталог (\OIS\SLGMS.NT4\palette).

Вы можете создать сложные палитры, назначая различные имена файлов моделей в палитре.

## Меню точки.

Часто, графическому редактору требуется контрольная точка для объекта, группы, или набора объектов и подсказка пользователю как её выбрать.

Например, все операции в инструментальной панели создания объекта требуют, чтобы контрольная точка завершила операцию.

Это меню может использоваться для определения точки.

<u>M</u> ode Find <u>C</u> enter Find <u>P</u> oint Same Point Changa <u>R</u> ef Reset Ref	<u>P</u> oint ≁ <u>C</u> enter
Points	×  0 0 0

Рисунок 3.48 Меню точки и строка инструментов точки.

Панель инструментов точки имеет следующие кнопки слева направо.

1) Кнопка точки.



Устанавливает режим точки, графического редактора.



Устанавливает режим центра в графическом редакторе.

3) Кнопка поиска середины.

Находит среднюю точку объекта.



ത്ര

4) Кнопка поиска заданной точки.

Осуществляет поиск заданной точки..

5) Кнопка «та же самая точка».

Многократно использует точку, предварительно установленную.

6) Кнопка замены ссылки.

Устанавливает контрольную точку.

7) Кнопка сброса ссылки.

Сбрасывает контрольную точку.

## Меню режима (модификация).

Меню «Mode» устанавливает в графическом редакторе режим центра или режим точки. Режим устанавливает контрольные точки, которые используются для преобразований объектов: перемещение, вращение, масштабирование, и реверс.

Эта команда игнорирует контрольные точки, установленные на объекты с использованием команды «Change Ref» (замена ссылки).

## Команда точки.

В Режиме точки, точки являются предопределены к объекту, самому близкому к выбранной точке.

Центр.

В режиме «центр», перемещение, вращение, или другие операции выполняются относительно центра объекта.

## Поиск центра.

Эта команда находит точку центра объекта. Опция «Find Center» (поиск центра) для кругов приводит точку центра к ранее выбранному центру круга.

Эта опция также находит центр расширения для всех других объектов.

## Поиск точки.

Эта команда находит на выбранном объекте точку, самую близкую к предопределенной. Для Ломаных линий, линий, и кривых, предопределенными точками являются точки, использованные при создании этих объектов. Для круга, предопределёнными точками являются, точки в 12, 3, 6, и 9-часовых позициях (0; 90; 180; 270 градусов).

Для прямоугольника предопределённые точки во всех четырех углах и в середине каждой из сторон.

Для сектора предопределённая точка в конечных точках сектора и в середине сектора.

## Та же самая точка.

Эта команда многократно использует ту же самую точку, которую использовали в предыдущей операции в пределах текущего объекта

Например:

Как «Та же самая точка» многократно используется средняя точка круга как центр следующего круга, или многократно используется вторая точка в круге, представляя радиус, как радиус следующего Круга. Эта операция делает прямоугольники одинакового размера; выбирая первую точку и используя эту опцию, чтобы обеспечить константу смещения для второй точки. Эта команда используется с секторами для нахождения точки того же самого центра или конца.

## Изменение ссылки.

Эта команда устанавливает контрольную точку на объекте.

Графический редактор имеет четыре кнопки редакции; вращать, переместить, полностью изменить, и масштабировать. Операции вращения, перемещения и масштабирования выполняются основанными на контрольной точке, и объект преобразуется {конвертируется} относительно контрольной точки(операция Reverse основана на линии «ссылке.

Объект вращается вокруг контрольной точки. Контрольная точка отображена как маркер и используется с линией ссылки для определения угла вращения {циклического сдвига}. В операции пересылки (перемещения), объект располагается в позиции, которая соответствует контрольной точке, отображенной как маркер.

Для операции масштабирования, есть много путей использования контрольной точки. Когда размер объекта изменяется, все точки, которые определяют объект, перемешаются на расстояние пропорциональное масштабу. Если масштабирование выполнено относительно начала координат, точки перемещаются ближе или дальше, от начала координат и объект отображается с измененным размером и позицией.

Если размер объекта изменяется без определения контрольной точки на объекте, расширение прямоугольника происходит относительно контрольной точки. Контрольная точка отображена как маркер наряду с расширением прямоугольника.

Когда вы изменяете контрольную точку в списке выбора, контрольная точка каждого объекта изменяется независимо (по одной).

Для установки контрольных точек всех объектов к одной и той же точке, установите точку для первого объекта, затем используйте команду «Same Point» (та же самая точка) для каждого объекта.

Заданная по умолчанию контрольная точка - центр объекта. Для комплексных, нерегулярных объектов типа ломаных линии или кривых, центр рассчитывается с использованием центра расширения прямоугольника объекта.

Для масштабирования объекта, центр - единственная используемая контрольная точка, которая не двигается при масштабировании объекта.

Контрольная точка используется для вариантов копирования, перемещения, масштабирования, или вращения из инструментальной панели создания объектов.

Если объект не имеет необходимой контрольной точки, временная контрольная точка рассчитывается как центр расширения объекта.

## Сброс изменения ссылки.

Эта команда сбрасывает новую контрольную точку объекта, к его заданной по умолчанию контрольной точке - центру его расширения степень.

## Справочное меню.

В справочном меню, вы можете проверить информацию о номере версии графического редактора.



Рисунок 3.49 Справочное меню и инструментальная панель справки.

## Команда «версии».

Команда «Version» открывает диалог для отображения информации версии графического редактора.



Рисунок 3.50 Пример информационного диалога о версии.

## 4. Динамика

## Краткий обзор динамики.

Динамика - атрибут, который определяет поведение объекта. Это - язык сценария для изменения цвета и формы, перемещения, и вращения объектов (Прямоугольники, символы, круги, группы, и т.д.). Назначение динамики для объектов модели, заставляет каждую подмодель вести себя по-другому.

#### Формат динамики.

```
Variable(Condition)
Condition
Action
```

Например: переменная закрепляет чтение данных от контроллера. Переменная определяет состояние. Действие определяет поведение объекта (прямоугольник, символ, или круг, и т.д.). Следующее - пример динамики.

Пример:

Когда переменная \_pFI - 1, tcolor 5 - результат. Когда значение \_pFI является отличным от 1, объект отображает начальное значение. В этом примере, следующая динамика имеет то же самое значение:

Вы можете вставить логическое выражение операции на переменном определении. Также, Вы можете использовать а

арифметическое выражение подобно

Пример 2:

```
_pPV
= *
stext _pPV "%6.1f"
```

В качестве переменной «\_pPV» вы можете вставить логическое выражение (=\* means "if changed")., Вы можете, также, использовать арифметическое выражение.

Пример 3:

```
_pVar
= 0 : 9
fcolor 1
= 10
fcolor 2
```

Когда значение \_pVar - от 0 до 9, цвет заливки - 1. Когда значение - 10, цвет заливки -2. Условие 0:9 означает от 0 до 9. Когда одна переменная имеет много условий {состояний}, они могут быть описанны в строке.

fcolor - действие, которое изменяет цвет заливки объекта.

Пример 4:

\_pFI == 1 && \_pPV > 100 = 1 vis0

Когда значение \_pFI - 1, и значение \_pPV является большим чем 100, объект не отображен. Иначе отображен (если начальное состояние объекта - Vis).

Как показано в этом примере, Вы можете иметь множественные логические выражения (операции) для переменной.

Пример 5:

fabs(\_pPV) > 100
 = 1
 scale2

Когда абсолютное значение \_pPV является большим чем 100, размер объекта удваивается.

## Действия.

Действия динамики состоят из динамических действий, предусмотренных как стандарт SLGMS, и добавленных для TOSDIC-CIE как стандартные функции повторного вызова в приложении. Действия, которые добавляются для TOSDIC-CIE, называют функциями повторного вызова. Они описаны с запросом в части действия. Запрос опущен, когда они описаны в переменной части или используются с динамическими действиями.

Пример 1:

```
_pPV
= *
call NaNX(_pPV,_pDCML)
```

При любом изменении \_pPV, символьный объект отображается со значением \_pPV и десятичной точкой \_pDCML.

Пример 2:

```
BitAnd(_pVAL1, _pVAL2)
= 0
fcolor 1
```

Когда операция «AND битов \_pVAL1 и \_pVAL2» приходит к 0, цвет заливки объекта является 1. Хотя «BitAnd» - функция повторного вызова, запрос не нужен, так как он уже находится в переменной.

Пример 3:

\_pVAL1 = \* fcolor(BitAnd(\_pVAL1,\_pVAL2)+2)

Когда значения \_pVAL1 и\_pVAL2 изменяются, цвет заливки объекта результат операции суммы AND с битами \_pVAL1 и \_pVAL2 и 2. Хотя «BitAnd» - функция повторного вызова, запрос не нужен, так как он уже находится в динамической части действия.

## Переменные

Есть следующие три типа переменных, которые определяют поведение объекта:

Глобальные переменные (начинаются с \_\_) Частные переменные (начинаются с \_p) Общие переменные (ни глобальные переменные, ни частные переменные) Для обозначения переменных используется алфавитно-цифровая комбинация. «.\_pPV.» в верхнем и нижнем регистре – различные.

Следующие разделы описывают использование каждой переменной.

## Глобальные переменные.

Глобальные переменные - переменные, назначенные на систему и использование по определению.

INIT	Указывает последовательность инициализации модели. При 5, 3, 1, и 0 заказе, изменяется отображение экрана.
BGCOLOR	Указывает значение для цвета фона.
_ZOOM	Указывает масштаба экрана. Экран изменяется от 0 до 4.
UPDATE	Подсчитывает в интервалы 0.5 секунд.
BLINKCT	Мигающий цикл, который изменяется между 1 и 0. 1 указывает - мигание включить, и 0 указывает - мигание выключить.

#### Частные переменные.

Частные переменные - переменные, которые могут использоваться в пределах модели или подмодели. Если переменная \_pPV используется в подмодели, и подмодель приведена в качестве примера для множества моделей, каждая \_pPV в подмодели обрабатывается как различные переменные.

#### Общие переменные.

Общие переменные - переменные, которые являются адресатами для переименования применяемых подмоделей.

Следующие примеры иллюстрируют использование этих переменных.

Пример 1:

Когда вы создаете модель и пробуете отобразить значение процесса (PV переменная) тэга, вы сначала должны создать символьный объект в модели. Символьный объект отображает значение PV, так как вы описываете последующую динамику.

\_pPV = \* stext \_pPV "%6.1f"

Эта динамика означает, что, когда \_pPV содержит значение процесса Тэга (PV переменная) и значение \_pPV, или значение процесса изменяется, значение отображается в stext с 6 целыми разрядами и одним десятичным (000000.0)

Затем, Вы должны назначить для \_pPV значение тэга процесса из появляющегося меню переменные динамики. В открывающемся меню переменных отображается список частных переменных, используемых в модели. Выберите переменную, которую Вы хотите назначить значением \_pPV в этом случае, и нажмите кнопку <Pедактированиe>, чтобы назначить \_pPV переменную.

Например, чтобы назначить значение для \_pPV: тег номер FIC0001, введите "FIC0001. PV" в общей ячейке (с " ") и выберите реальный тип из списка. Когда модель отображается в OIS, строка отображает значение номера тэга FIC0001 тэга.

Процесс назначения переменных связывает переменную \_pPV и значение тэга процесса. Связывающий их процесс обрабатывается в Динамике.

Если Вы выбираете модель непосредственно в меню [Редактирование] - [Выбор] - [молель]] ([Edit]-[Select]-[Model])и проверяете динамику, назначается следующая Динамика.

\_\_INIT = 1

call DefV2(& pPV, "FIC0001. PV")

При инициализации, \_\_ INIT=1, переменная \_pPV читается к данным тэга FIC0001. PV (когда отображается модель).

Если Вы используете функцию назначения переменной, это автоматически генерирует динамику для модели непосредственно. Вы можете удалить динамику и переместить модель к символьному объекту {цели} как показано ниже. Однако, Вы не должны использовать в этом случае функцию назначения переменной.

```
__INIT
= 1
call DefV2(&_pPV,"FIC0001. PV")
_pPV
= *
stext _pPV "%6.1f"
```

(Пример 2)

Как и в примере 1, пример 2 отображает значение процесса тэга (теговую переменную PV) на модели.

Когда есть больше чем одно значение переменной процесса, эффективно создать подмодель, которая отображает переменную и присоединить её к модели. Так что Вы запускаете создание подмодели подобно примеру 1, создаёте символьный объект {цель} и даёте подобную динамику символьного объекта.

\_pPV = \* stext \_pPV "%6.1f"

Затем, Вы назначаете \_pPV, как значение переменной тега процесса. В примере 1, функция назначения переменной используется, чтобы непосредственно установить номер тэга ("FIC0001. PV").

В этом примере, Вы должны ввести TAGATOM в общей ячейке назначающихся переменных, так как используется подмодель.

Динамика, назначенная на модель, когда Вы это вводите, представляет следующее:

\_\_INIT = 1 call DefV2(& pPV,TAGATOM)

Это отличается тем, что TAGATOM не заключен в " " так, чтобы TAGATOM был определен как общая переменная. Сохраните подмодель, созданную этим способом под именем DispPV как образец новой модели DispPV. Выберите DispPV, и переименуйте переменные динамики. Общая переменная TAGATOM отображается в зарегистрированной переменной.

Выберите TAGATOM, и введите название {имя}, указывающее номер тэга и значение PV. PV переменная тэга, который Вы вводите, обрабатывается в подмодели.

Например, если Вы создаете два образца подмодели DispPV. Первый, названный как "FIC0001. PV" и второй образец названный как "FIC0002. PV ". Первый DispPV отображает значение PV для FIC0001. PV и второй DispPV отображает значение PV для FIC0002. PV.

Если Вы опускаете " " (кавычки), это расценивается как переменная. Когда вы определяете строки типа тэга, Вы должны использовать" ".

(Пример 3)

В этом примере, подмодель примера 2, отображающая значение PV, разработана далее. данные тэга содержат не только значение процесса, но и десятичные разряды (после децимальной точки) для значения процесса.

Десятичные разряды будут также использоваться для отображения строки значения. Измените динамику символьного объекта подмодели с динамикой, которая включает десятичный разряд.

Следующий пример - динамика (NaNX - функция повторного вызова для отображения значения десятичных разрядов).

\_pPV = \* call NaNX(\_pPV,\_pDCML)

Затем, Вы должны назначить \_pPV и \_pDCML для значения тэга процесса и десятичного разряда соответственно.

Подобно к примеру 2, Вы используете функцию назначения переменных. В этом случае {регистре}, наборе

Как показано ниже (PV указывает значение процесса, и DCML - данные тэга, указывающие количество десятичных разрядов).

Переменная	Общая	Тип	Размер	Набор
_pPV	\$TAGNO\$#dPV	Реальная	1	Нормальный
_pDCML	\$TAGNO\$#dDCML	Целое число	1	Нормальный

С этого времени \_pPV и \_pDCML используют один и тот же тэг, заключённый в символы \$. "#d" имеет то же самое значение как и ",".

Этим способом, включая это в \$\$ символах, только TAGNO отображается когда подмодель приведена в качестве примера, и используется функция переименовывающая переменную. Вы только должны определить TAGNO для переименования переменной величины, хотя есть две переменные, которые определены в подмодели.

Например, если Вы определяете, что "FIC0001" для TAGNO при определении переименовывающихся величин, \_pPV переменная обрабатывается как "FIC0001. PV" и \_pDCML переменная обрабатывается как "FIC0001. DCML"внутренне.

#### ССЫЛКА:

Переменные в подмодели, которые были приведены в качестве примера для образцового увеличения

и определения включили в увеличение \$, Вы должны обработать переименовывающиеся Переменные величины в

моделируйте сторону снова. Иначе, процесс переименования недавно добавленных переменных не может быть выполнен.

Пример 4.

В этом примере, подмодель, создающая заполненный прямоугольник, который становится синим (номер цвета 4), когда состояние цифрового сигнала - 1 и коричневого цвета (колер номер 21), когда состояние - 0.

Сначала, создайте коричневый цвет, заполнения прямоугольника. Далее вы должны описать динамику, чтобы изменить {заменить } цвет заливки прямоугольника.

Тэг, который обрабатывает цифровой сигнал, имеет больше чем один входной сигнал (атом), поэтому вы должны определить бит. Следующее - динамика.

\_pFI[BitNo] = 1 fcolor 4

BitNo, описанный здесь обработан как общая переменная и отображен в переименовывающейся переменной образеца подмодели.

Затем, назначьте \_pFI на цифровое входное значение тэга, используя asign функцию переменной. Параметры настройки являются как показано ниже: (размер - 4, так как принято использовать тэг типа PB4, который имеет 4 массива. Если

номер массива не известен, выбирать-1. В этом случае, создайте переменные согласно типу тэга.).

Variable	Public	Туре	Size	Collection
_pFI	STAGNO\$#dFI	Real	1	Normal

Переменная \_pFI обработана как переменная типа массива. В этом примере, номер массива 4. Значение массива в SLGMS начинается 0. В этом примере, доступное значение для общей переменной BitNo - от 0 до 3 то есть 4 массива.

В изменяющейся переменной, которая выполнена в модели, где эта подмодель приведена в качестве примера, TAGNO и BitNo - адресаты для переименования. TAGNO указывает номер тэга, а BitNo указывает разрядную позицию, чтобы отметить сигнал.

Изменяющаяся переменная, которая определена BitNo принимает значение от 0 до 3. Этим способом, если Вы не хотите, чтобы значение было от 0 до 3, Вы можете определить и значение от 1 до 4, исправляя динамику в подмодели как показано ниже.

```
_pFI[BitNo-1]
= 1
fcolor 4
```

Используя BitNo-1 как разрядную позицию, пользователь подмодели может использовать значение от 1 до 4, вместо искусственного значения от 0 до 3.

Вы можете обработать подмодели, созданные в примерах 2 - 4, как будто они - модели и назначают их к графическому экрану. В этом случае вы можете определить изменяющиеся переменные процесса в OIS engineering tool.

Следующее – регистрация названия подмодели созданного в примере 4 - DispDICo1, номера тэга, для отображения - MTR0001, и BitNo - 2.

Title	Model Name	Operation Mode	Variable Rename
Signal Status	DispDICol	Engineer	TAGNO:MTR0001,BitNo:#12

В спецификации переименования переменных величин в OIS engineering tool., Вы можете переименовать переменные в следующий формат:

Для имени переменной строк: Строка

Для имени целых переменных #I Значение

Для имени переменной вещественных чисел: #F Значение

Каждое определение отделено ",". Не используйте никакого пробела.

## Подмодели

Подмодель представляет собой часть, которая может быть разделена между моделями. Подмодели созданы, используя общие переменные и общие переменные переименованы, когда Модель приведена в качестве примера.

Несколько подмоделей обеспечены как стандарт. Обратитесь в список стандартных подмоделей в приложении. Некоторые из них могут использоваться как они есть, без изменений. Вы можете использовать стандартную подмодель как пример, изменив это под другим именем и сохранив

Следующие разделы иллюстрируют, как использовать специальные модели, и примеры использования динамики.

Пример 1

В этом примере используется стандартная подмодель arrow1. Образец arrow1 и переименовывает-Ся, это как показано ниже:

TAGNO VLV0001 BitNo 0 OnColor 11 OffColor 12

Как результат этого процесса переименования, подмодель отображает нежелательное мерцание в цвете 11 когда цифровой вход номер 1 тэга типа PB VLV0001 становится 1, и отображен в цвете

12 когда это становится 0.

Если Вы хотите отобразить подмодель только когда цифровой вход номер 4 тэга VLV0002

становится 1, Вы можете добавить следующую динамику к arrow1, приведенной в качестве примера подмодели.

\_\_INIT = 1 call DefVar(&\_pFI,"VLV0002.FI",1,4,-1,1,1) \_pFI[3] = 0 vis 0

Когда цифровой вход номер 4 тэга VLV0002 становится 0, дисплей подмодели делается невидимый (vis0).

Используя это, можно управлять отображением подмодели в соответствии с состоянием других сигналов, (хотя в этом примере описана функция DefVar повторного вызова, вы можете исполнить назначение с функцией Assign Vars).

## Процедура

Динамика, описанная выше соответствует объектам типа изменяющихся цветов или преобразования.

Динамику, которая исполняет некоторые операции при нажатии мышью, называют процедурой. Формат процеруры показан ниже.

## #

#### Действия

Действия процедуры выражены в функциях повторного вызова, которые разворачивают экраны, отображают тэги, образцы, или диалог конфигурации параметра для тэга.

Процедура - своего рода динамика, описывается заранее и назначается на объекты.

Например, если процедура описана для заполненного прямоугольника, то она выполняется щелчком мыши на заполненном прямоугольнике.

Если процедура описана для незаполненного прямоугольника, должна быть предпринята осторожность.

В этом случае описанная процедура не выполнима, если Вы не щелкните мышью точно на контуре незаполненного прямоугольника.

Пример 1

Если Вы создаете заполненный прямоугольник с динамикой, которая включает процедуру, имеющую функцию развёртки экрана точки тэга, экрана группы, или графического экрана, нажимом на заполненный прямоугольник, то динамика должна быть как показано ниже. Номер тэга определен как переменная которая также используется его в подмодели.

```
#
call DoGraphPoint($TAGNO$,_pSCNO)
___INIT
= 1
call DefVar(&_pSCNO,$TAGNO$#dSCNO,1,4,1,1,1)
```

DoGraphPoint - функция повторного вызова функции развертки точечного экран, экран группы, или графического экрана тэга, обозначенного \$TAGNO\$. \_pSCNO указывает номер экрана который является одним из параметров тэга. Если это значение - не 0, графический экран принят для назначения тэгом так, чтобы расширение графического экрана было результатом номера обозначенного \_pSCNO.

Если \_pSCNO - 0, расширение применяется к экрану группы или экрану точки.

Если тэги, обозначенные \$TAGNO\$, сгруппированы, расширяется сгруппированный экран; в противном случае точечный экран.

В этом примере, нормальная динамика описана под процедурой. Эту динамику определяет переменная \_pSCNO так, чтобы номер экрана, который является параметром тэга, считывался и устанавливал действие.

Как показано в этом примере, вы должны описать процедуру и динамику вместе, приэтом сначала описывается процедура.

Пример 2

Как правило, динамика, которая имеет процесс процедуры, не проста как пример 1. Изображения кнопки часто используются, Нажатая кнопка должна быть отображена вогнутой.

Кроме того, когда кнопка нажата и операция выполнена, нажатие кнопки снова не должно запускать другую операцию так, чтобы функция стала сложной.

В этом примере, TagPtnBtn обеспечения стандартной подмодели раздроблен и отображает содержание. TagPtnBtn подмодель отображает образец дисплея тэга, выбранного в диалоге номера.



Рисунок 4.1 Структура числа TagPtn Btn

TagPtnBtn состоит из четырех объектов. Сгруппирован из двух ломаных линии, одиного заполненного прямоугольника, и одного символа. Процедура назначается на объект группы.

Подробности процесса показаны ниже:

Таблица 4.1 Процесс группового объекта

Описание	Процесс элемента
ButtonHiliteChk	Переключает цвета двух объектов ломаных (верхняя рамка{фрейм} и нижняя рамка{фрейм}) перед группировкой щелчком мыши. Это позволяет отображать кнопку, как нажатую (утопленную), при нажатии.
ButtonMessageChk	Отображает строку в переменном СООБЩЕНИИ на строке состояния когда она нажата.
SetRTVarIntGismoChk	Устанавливает целочисленное значение 1 в переменную TtlDsp используемую подмоделью TagPattern (отображено в диалоге-примере2).

SetRTVarStrGismoChk	Устанавливает строку номера тэга на переменную TAGNO, используемую подмоделью TagPattern (отображено в диалоге-примере2).
DialogInstance2 (диалог-пример 2)	Отображает диалог в позиции (X, Y) в размере (W, H) совместно с моделью TagPattern. Модель TagPattern требует изменения двух переменнх; TAGNO и TtIDsp. Процесс переименования начинается с подмодели, не присоединенной непосредственно к модели, двумя функциями повторного вызова, SetRTVar используются, чтобы исполнить процесс переименования для этих двух переменных.

Для группового объекта, динамика устанавливается помимо процедуры, описанной выше. Динамика использует функцию DefVarInt повторного вызова, чтобы создать целую переменную pChk и присваивает ей 0, как начальное значение.

Эта переменная имеет другое значение, отличное от 0, когда диалог создан в DialogInstance2 и возвращается к 0, когда диалог удален.

Это используют для изменения цвета символов, символьного объекта прежде, чем установлена группировка так, чтобы цвет изменялся (tcolor7 или tcolor13), в зависимости от значения переменной: \_pChk - 0 или нет. Это выражение изменяет цвет символов, делая диалог недоступным, в то время как это отображено.

Точно так же функции повторного вызова в другой процедуре используют значение \_pChk, чтобы предотвратить операции во время отображения диалога.

## Атом тэга процесса

Все данные тэга идентифицируются строкой номера тэга + имя атома.

Пример

Номер Тэга:	"PID0001", "PB40003"
Имя Атома	"PV", "SV", "MV", "FI", "FO"
Значение величины	PID0001 = "PID0001. SV"
Лампа статусного сигнала 1	PB40003 = "PB4003. FI [0] "

Этот раздел описывает имена атомов, часто используемых в Графическом Экране. Для изучения подробностей относительно параметров отображенных в экране сигнализации, дисплее тэга процесса или точки, обратитесь к приложению "Информация о тэге процесса".

Имя атом	Название данных	Тип данных	№ массива	замечания
PV	Текущее Значение	Реальные	1	Технические Данные Модуля
RH	Инструментальный Диапазон Верхний Предел	Реальные	1	Технические Данные Модуля
RL	Инструментальный Диапазон Нижний Предел	Реальные	1	Технические Данные Модуля
DCML	Десятичная точка	Целое число	1	0~5
EU	Технические Данные Модуля	Строка	1	
SV	Значение Набора	Реальные	1	Технические Данные Модуля
MV	Значение Операции	Реальные	1	%-ые Данные

#### Таблица 4.2 Данные Индикатора/Контроллера

Таблица 4.3 Данных команды кнопки выключателя (Push Button Switch)

Имя атом	Название данных	Тип данных	№ массива	замечания
FI	Состояния Лампы	Целое число	8	0 : OFF, 1 : ON
FO	Состояния Кнопки	Целое число	8	0 : OFF, 1 : ON
SNAM	Отображает символ	Строку	8	До 16 Символов
SCON	Цвет дисплея Сигнал ON	Целое число	8	0 ~ 31
SCOF	Цвет дисплея Сигнал OFF	Целое число	8	0 ~ 31

#### ПРИМЕЧАНИЯ

Номер массива (от 0 до 7) соответствует номеру (1 - 8) сигнала.
 Для "FI [0] " и "SNAM [2] ", только один элемент может быть определен (Номер Массива = 1 в этом случае).
 FI [0] до FI [7] может быть идентифицирован от FI 1 до FI 8.
 FO [0] до FO [7] может быть идентифицирован от FO 1 до FO 8.

#### Таблица 4.4 Общих данных Тэга

Имя атом	Название данных	Тип данных	№ массива	замечания
TAG	номер тега	строка	1	До 16 Символов
TYPE	Тип тега	"целое"	1	
NAME	Имя тэга	Строку	1	До 32 Символов
AREA	Номер категории сигнала	Целое число	1	0 ~ 63
SIGN	Метка вывески	Строка	1	До 16 Символов
COLR	Цвета вывески	Целое число	1	0 ~ 31
LOCK	Установка блокировки	Целое число	1	0:включить, 1:отключить
SCNO	автоматическое отображение номера экрана	Целое число	1	0, 1 ~ 4096
MDL	отображение номера образцовой модели	Целое число	1	0, 1 ~ 9
SCN	Установка просмотра	Целое число	1	0: ON, 1: OFF
MOD	Режим управления	Целое число	1	M,A,C,RM (0 ~ 3)
ATYP	Метод формирования монитора тревог	Целое число	1	А, В, С Тип
AMON	Общая установка тревог	Целое число	1	0: ON, 1: OFF
MON	Индивидуальная установка монитора тревог	Целое число	32	0: ON, 1: OFF
AGRD	Тип степени тревог	"целое"	32	Главный, Средний, Незна- чительный Major, Medium, Minor
ALIT	Аварийная ситуация статус тревоги	Целое число	32	0: Норма, 1: Тревога

ATIT	Дисплей статуса тревог	Целое число	32	0: Норма, 1: Тревога
FLIT	Сигнал подтверждения состояния тревоги	Целое число	32	0: Подтверждено, 1: неподтверждено

#### ПРИМЕЧАНИЯ

1. Некоторые атомы не существуют для некоторых типов тэгов.

2. Номер Массива (от 0 до 31) для MON, AGRD, ALIT, STIT, и FLIT, соответствуют элементам сигнализации. Элементы сигнализации изменяются в зависимости от типа тэга.

ALIT: Статус аварии

Возникновение сигнала не зависит от подтверждения сигнала Метод, Установка Монитора, или Тип Степени. STIT: Сигнальное состояние дисплея

Сигнальное состояние дисплея, которое зависит от метода подтверждения сигнала, установки монитора, или тип степени.

FLIT: Сигнал состояния подтверждения

Неподтвержденная аварийная ситуация, вызывает мигание. Операция выключает нежелательное мерцание.

## 5. Создание Подмоделей

Этот раздел описывает, как создать подмодели.

## Изменение символов дисплея

Символы дисплея могут быть изменены согласно состоянию сигнала выключателя кнопки команды (push button switch).

1) Создать символ "Операция останова", используя заполненный текстовый прямоугольник



Рисунок 5.1

2) Зарегистрировать динамику.

Регистрируйте следующую динамику.

```
__INIT
= 1
call DefVar(&_pFI, $TAGNO$#dFI,1,4,-1,1,1)
_pFI[BitNo]
= 1
stext "Start Operation" "%s"
```

Когда \_pFI [BitNo] - 1, символ дисплея изменяется на "Операция начала". Когда \_pFI [BitNo] отличается от 1, условие не выполнено, поэтому отображается начальное значение "Операция Останова".

- 3) Сохранить как подмодель (ExLetterChg).
- 4) Зарегистрировать "ExLetterChg" в Графическом Экране и переименовать переменную.

BitNo: FI номер сообщения (от 0 до 7) ТАGNO: Номер тэга команды кнопки (Push Button)

I	Model Variables	Insert variables
	BitNo	1
	TAGNO	"P <b>B</b> 4_0001"

Рисунок 5.2

Когда состояние лампы (FI2) команды кнопки выключателя - OFF(0), отображена " Операция останова".

Когда состояние лампы (FI2) находится ON(1), отображена "Операция начала"



#### Отображение символов индикатора в сигнальном цвете

Символы индикатора команды кнопки выключателя (push button switch) могут быть отображены в сигнальном цвете.

1) Создать "состояние" (status), используя заполненный текстовый прямоугольник.



2) Зарегистрировать динамику.

Зарегистрируйте следующую динамику

```
__INIT
```

```
= 1
             ; (Чтение переменной регистраторов (Высокоскоростной Тип)
 call SetCnfDefV10(&_pTYPE,$TAGNO$#dTYPE,&_pALIT,TAGNO$#dALIT,
        &_pSTIT, $TAGNO$#dSTIT, &_pFLIT, TAGNO$#dFLIT,
        &_parea, $tagno$#darea, &_patyp, tagno$#datyp,
        &_plock, $tagno$#dlock, 0, 0, 0, 0,
        &_pAGRD, $TAGNO$#dAGRD)
 call DefVar(&_pSNAM, $TAGNO$#dSNAM,1,2,-1,1,1); (Символы чтения индикатора)
_pTYPE
= *
      ; Останавливает нежелательное мерцание сигнала: Регистраторы отображают операцию резюме)
  call SetAlarmConf(AddTagDot($TAGNO$, 0),_pTYPE, &_pAREA, &_pLock,
                                   &_pATYP, &_pALIT, &_pSTIT, &_pFLIT)
pSNAM
 = *
  stext pSNAM[BitNo] "%s"
  tcolor GetAlmColor(GetAlmGrd(&_pSTIT, &_pFLIT, &_pAGRD,BitNo),
                                                             __BLINKCT)
```

- 3) Сохранить как подмодель (ExTagStatus).
- 4) Зарегистрировать 4 "ExTagStatus" подмодели в Графическом Экране и переименовать переменную.

BitNo: Номер сигнала (0, 1, 2, 3)

ТАGNO: Номер тэга команды кнопки (Например: "PB4\_0003")

Четыре типа символов индикатора команды кнопки переключают "PB4\_0003", отображены в соответствующим сигнальным цветом.



Рисунок 5.5

#### Развёртка точечного экрана

Вы можете отобразить аварийную ситуацию тэга и развернуть точечный экран касающийся операции.

1) Создать шесть элементов {рисунков} в приведенном ниже порядке.

- 1. Заполненный прямоугольник (Фон для полной кнопки)
- 2. Верхние и правые ломаные линии (в той же самой позиции как и заполненный прямоугольник (1))
- 3. Нижние и правые ломаные линии (в той же самой позиции как и заполненный прямоугольник (1))
- 4. Заполненный текстовый прямоугольник "ТЭГ" (в фоне (1), том же самом фоне и цвета края)
- 5. Заполненный текстовый прямоугольник "PV" (в фоне (1), том же самом фоне и цвета края).
- 6. Заполненный текстовый прямоугольник "EU" (в фоне (1), том же самом фоне и цвете края).



Рисунок 5.6

2) Группа все шесть объектов {целей} и регистр следующая Динамика.

```
#
 call ButtonHilite(0,0)
                                      ; Операция кнопки Дисплея
call DoPointDisp($TASGNO$, "POINT", 1, 1) ; Развертка точечного экрана
 call ButtonMessage2("Expands to the Point Screen of", $TAGNO$)
___INT
 = 1
              ; Переменная регистраторов; читайте (высокоскоростной тип)
 call SetCnfDefV10(&_pTYPE, $TAGNO$#dTYPE, &_pALIT, TAGNO$#dALIT,
                          &_pSTIT, $TAGNO$#dSTIT, &_pFLIT, TAGNO$#dFLIT,
                          &_parea, $tagno$#darea, &_patyp, tagno$#datyp,
                          &_plock, $TAGNO$#dlock, &_pV, TAGNO$#dPV,
                          &_pDCML, $TAGNO$#dDCML, &_pAGRD, $TAGNO$#dAGRD),
 call DefVar(&_pEU, $TAGNO$#dEU,1,2,1,1,1); чтение регистрции модуля
_pTYPE
 = * ;
          Stops the Alarm Flicker.Registers Display Resume Operation
 call SetAlarmConf(AddTagDot($TAGNO$,0),_pTYPE,&_pAREA,&_pLock,
                                   &_pATYP,&_pALIT,&_pSTIT,&_pFLIT)
```

3) Зарегистрировать динамику для данных дисплея.

\* Заполнить текстовый прямоугольник "ТЭГ" Номер тэга отображен в сигнальном цвете дисплея.

stext \$TAGNO\$ "%s"
call DispAlarmCol(\_pTYPE,&\_pSTIT,&\_pFLIT,&\_pAGRD,1,\_\_BLINKCT)

\* Заполнить текстовый прямоугольник "PV" Данные PV отображены с указанием десятичного места.

```
_pPV
= *
call DispRealDara(&_pPV,_pDCML)
```

\* Заполнить текстовый прямоугольник "EU"

Отображается технический модуль.

```
_pEU
= *
stext_pEU "%s"
```

- 4) Сохранить как подмодель (ExTagData).
- 5) Зарегистрировать "ExTagData" в Графическом Экране и переименовать переменную. ТАGNO: Номер тэга



Рисунок 5.7 Пример дисплея (TAGNO = "PID\_0001")

## Отображение заполненных рисунков

Сложные рисунки заполненные соответственно значению PV.

1) Создать рисунок из восьми строк и восьми секторов по трём точкам, и выполнить группирование.



2) Выбрать «право» для направления заполнения.



Рисунок 5.9

3) Зарегистрировать динамику.

Зарегистрируйте следующую Динамику.

```
_pPV

= _pRL:_pRH

fpercent 0:100

< _pRL

fpercent 0

_pPV || _pRL || _pRH

= * ; Перерисовка изменённых данных

redaw
```

4) Создать фон заполненного прямоугольника.



5) зарегистрировать динамику.

Регистрируйте следующую Динамику.

\_pPV || \_pRH || \_pRL = \* ; Перерисовка изменённых данных redraw

6) Сгруппировать фон и рисунок, и зарегистрировать динамику.

\_\_pPV || \_pRH || \_pRL = \* ; удаление изменённых данных batcherase

7) Назначить переменные PV, RH, и RL (реальная, нормальная группа, размер = 1).

		Edit Varia	ible Assignn	nent	×
Variable Assignmer Private p?V pRH pRL	nt Public STAGNOSØdPM STAGNOSØdRH STAGNOSØdRL	Private name Public name -Type	LaPY STAGNO(#4PV C Intege Other		g CK Cancel
		Edit	Cancel	ок	

Рисунок 5.11

- 8) Сохранить как подмодель (ExTagLevel).
- 9) Зарегистрировать "ExTagLevel" в Графическом Экране и переименовать переменную. ТАGNO: Номер тэга



Рисунок 5.12 Пример дисплея (PV=35 %, 65 %)

## Отображение значение PV используя амперметр

Значение PV может быть отображено в амперметре, используя вращение {Циклический сдвиг}.

1) Создать рисунок в следующем порядке.

1. Заполненный полукруг (три сектора)

Установите это в «невыбираемый» для облегчения более поздних операций.

2. Заполненный Круг (с тем же самым центром как 1)



Рисунок 5.13

3. Группа из одиннадцати строк (масштаб).



Установите контрольную точку Стрелки в 12-часовом направлении и вращайте её по часовой стрелке и против часовой стрелки с 10 степенями свободы (делениями).



Рисунок 5.15

4. Заполненный многоугольник (Игла)

Создайте стрелку в 12-часовом направлении, и вращайте контрольную точку вокруг центра Круг 50 степеней свободы (дискрет)



5. Заполненный текстовый прямоугольник "PV"



Рисунок 5.17

2) Сгруппировать пять объектов, и зарегистрировать следующую динамику.

```
INIT
 = 1
                  ; То же самое как назначение переменной в примере4
  call DefV2(& pPV,$TANGP$#dPV)
  call DefV2(& pRH,$TANGP$#dRH)
  call DefV2(& pRL,$TANGP$#dRL)
_pPV ||_pRH || _pRL
 - *
                 ; Удаление при изменении данных
  batcherase
   3) Зарегистрировать динамику для данных дисплея.
1. Заполненный полукруг
2. Заполненный круг
3. Сгруппировать одиннадцать строк (Масштаб)
_pPV || _pRH || _pRL
                 ; Перерисовка при изменении данных
 =
   *
  batcherase
4. Заполненный многоугольник (Игла)
_pPV
 = pRL: RH
 rotate 0:-100 ; Вращение по часовой стрелке со100 степенями {дискретами} для RL и RH
> pRH
```

rotate -100 ; Вращение по часовой стрелке со 100 степенями (дискретами) для другого чем RH

5. Заполненный текстовый прямоугольник "PV"

```
_pPV
= *
stext _pPV "%4.of"
_pPH || _pRL
= * ; Перерисовка при изменении RH или RL
```

redraw

- 4) Отобразить данные для RH и RL.
  - 1. Заполненный текстовый прямоугольник "RH"

```
_pRH
= *
stext _pRH "%4.of"
2. Заполненный текстовый прямоугольник "RL"
_pRL
= *
stext _pRL "%4.0f"
```

- 5) Сохранить как подмодель (ExTagMeter).
- 6) Зарегистрировать "ExTagMeter" в Графическом Экране и переименовать переменную. ТАGNO: Номер тэга



Рисунок 5.18 Пример дисплея

## Динамический дисплей

Вы можете использовать перемещение, или вращение, чтобы получить динамический дисплей. Когда указанный сигнал идет, Ленточный конвейер (барграф) перемещается, ролик вращается.

- 1) Создать рисунки в следующем порядке.
  - 1. Фон (заполненный прямоугольник)
  - Заполненный прямоугольник в том же самом цвете как фон (Никакой рамки {фрейма}: Тип линии = 0) Установите название {имя} объекта облегчающее более поздние операции.
  - 3. Сгруппировать четыре заполненных прямоугольника (без рамки {фрейма}).
  - 4. Строка в основании.





2) Сгруппировать четыре объекта, и зарегистрировать следующую динамику.

```
__INIT
```

```
= 5
 call DefVarInt(&_pCNT,0) ; Количество движения (от 0 до 7)
  call DefVarInt(&_pCOL,0) ; Цвет дисплея (от 0 до 3)
= 1
 call DefVar(&_pData,TagAtom,1,4,1,1,1)
 _UPDATE && _pData
              ; Выполняется, когда pData - не 0
= 1
 call ChgIntVar(&_pCNT,_pCNT+1)
 call ChgIntVarCond(&_pCOL,_pCOL+1,_pCNT>=8)
 call ChgIntVarCond(&_pCNT,0,_pCNT>=8)
  batcherase
  3) Зарегистрировать динамику для данных дисплея.
         1. Фон (заполненный прямоугольник)
            _pCNT
             = *
                        ; Перерисовка при изменении в количестве движения
              redraw
         2. Заполненный прямоугольник в том же самом цвете как фон
         _ _pCNT
         >= 5
          fcolor BitAnd ( pCOL, 3) ; Отображать, когда количество движения> = 5
         = б
        scalex 2 ; Отображать в удвоенном размере, когда количество движения =6
         = 7
        scalex 3 ; Отображать в утроенном размере, когда количество движения =7
```

3. Установить динамические атрибуты в четырех заполненных прямоугольниках, которые сгруппированы. (Установка слева на право, последовательно)

```
_pCOL

= *

fcolor BitAnd(_pCOL+3,3)

_pCOL

= *

fcolor BitAnd(_pCOL+2,3)
```

```
_pCOL
= *
fcolor BitAnd(_pCOL+1,3)
_pCOL
= *
fcolor BitAnd(_pCOL,3)
_pCNT
= 5
scalex (3./4.) ; отображается 3/4 при перемещении количества - 5
= 6
scalex (2./4.) ; отображается 2/4 при перемещении количества - 6
= 7
scalex (1. 4.) ; отображается 1/4 при перемещении количества - 7
```

Установите динамические атрибуты, чтобы переместить полную группу.

\_pCNT = \* movex \_pCNT

4) Создать ролик, используя заполненный круг и четыре строки.



Рисунок 5.20

• Динамика для заполненного круга

#### \_pCNT

= \* ; Перерисовка при изменении в количестве движения redraw

#### • Динамика для сгруппированных четырех строк

```
_pCNT
```

= \* ; Вращается по часовой стрелке 1/8 90 степеней {дискрет}. rotate(-11.25\*\_pCNT)

• Динамика для заполненного круга и сгруппированных четырех строк сгруппированная снова

```
_pCOL
= *
ecolor BIitAnd(_pCOL,3) ; Тот же самый цвет как в пакете в левый конец.
_pCNT
= *
batcherase
```

5) Копировать левый ролик к правому концу и создать нижнюю строку.



Рисунок 5.21

Измените цвет отображения правого ролика.

PCOL

= \*

color BIitAnd(\_pCOL,3) ; Тот же самый цвет как пакет в левый конец.

\_pCNT

= \*

batcherase

6) Сохранить как подмодель (ExMoveAction).

7) Зарегистрировать "ExMoveAction" в Графическом Экране и переименовать переменную. ТаgAtom: Номер тэга + Атом (Например, " PB4\_0004. FI3")

Когда состояние лампы команды кнопки переключает "PB4\_0004", идетут, шаги пакета на ленточном конвейере и ролик вращаются



Рисунок 5.22

Зарегистрируйте это действие последовательно для много раз так, чтобы пакет перемещался непрерывно.

Чтобы перемещать это всегда независимо от сигнала, регистрируйте TagAtom = "1" (Строка).



рисунок 5.23

# 6. Создание плоских моделей

Создание плоского образцового экрана - быстрый образцовый конверсионный инструмент, который установлен вместе с графический редактором.

Создание плоского образца означает: собрать два или больше отдельных файла подмодели, зарегистрированных в модели, как один образцовый файл и преобразовать это в быстро доступную модель, недоступную для редактирования графическим символьным редактором. Это улучшает загрузку модели. Поскольку подмодели были преобразованы к одной модели и это было далее преобразовано в модель, недоступную для редактирования графическим редактором, другая плоская модель должна быть создана снова, как только модель изменяется (создаётся с другим именем). То есть фиксация изменений с нормальной моделью и плоской моделью осуществляется после того, как отладка закончена.

Когда Вы запускаете плоский образцовый экран создания файла, появляется следующий экран.



#### рисунок 6.1 Экран моделирования плоских моделей

#### 1) Местная модель

Отображает список местных (локальных) моделей, созданных Графическим Редактором. Локальная модель – модель сохраненная Графическим Редактором. Локальные образцовые файлы нахо- в "C:\OIS\SLGMS.NT4\MODELS\LOCAL\MODEL".

#### 2) Пользовательская Модель

Отображает список моделей, к которым установлены, локальные модели, созданные Графическим Редактором.

Пользовательская модель - образцовый файл, который является адресатом эквивалентности среди множественных OIS's.

Команда Install (установить) Графического Редактора перемещает местные (локальные) модели в пользовательские модели. Пользовательские образцовые файлы хранятся в "C:\OIS\SLGMS.NT4\MODELS\USER\MODEL".

#### 3) Плоская Модель

Отображает список плоских моделей, созданных в плоском образцовом экране создания. Расширение образцового файла - m2, что означает, что это - высокоскоростная модель,

Плоские образцовые файлы существуют в "C:\OIS\SLGMS.NT4\MODELS\USER\FLAT".

4) Кнопка конвертирование <Convert>

Если Вы выбираете образцовый файл, чтобы преобразовать в простой файл из местного образцового списка или списка пользовательской модели, кнопка <Convert> становится действующей.

Когда эта кнопка нажата, модели, которые были выбраны, преобразуются в плоские модели. Так как файлы подмодели, которые использует образцовый файл обнаруживаются автоматически, моделируйте конверсионный инструмент применяемый к простому файлу. Список выбора преобразованных {конвертированных} моделей имеет только местные модели и пользовательские модели.

В этом экране создаются только плоские модели так, чтобы образцовые файлы являлись эквивалентом среди OIS's при образцовом обслуживании файла.

## 7. Графический экран дисплея времени

Следующее - экран запрашивающий выполнение когда используется больше чем одна подмодель, показанная в рисунках 4.5 (ExTagStatus) и рисунок 4.7 (ExTagData).

Следующая таблица и граф показывают номер подмоделей и экранный запрос времени работы. Номер подмоделей показывают в приращениях через 100 подмоделей. "100" представляет 50 "ExTagStatus" и 50 "ExTagData". Все взвешенные модели - плоские модели.

Измерение машинной среды

Центральный процессор:Pentium 233 МГЦ (Compaq, Deskpro 2000) Память:128Mbytes

Номер подмоделей	100	200	300	400	500	600
Экран, запрашивающий время	0.418	0.907	1.435	1.975	2.909	3.773
				M	одуль: с	екунда



рисунок 7.1 Экран запроса времени

Как видно из этой таблицы, скорость вызова подмоделей уменьшается при увеличении числа подмоделей (как увеличение числа тэгов). При отображении больших объёмов данных, исполнить экранный дизайн со ссылкой на этому данные.

Эти данные представляют соотношение между количеством подмоделей и экраном, вызывающим время.

Однако, экран, время также имеет влияние на величину динамики описания

подмодели. Экран, называющий запрашивающий также замедляет скорость, когда отображаются большие объёмы данных, больше чем одной подмодели

## 8. Резервная копия и восстановление

Этот раздел описывает процедуры для резервной копии и восстановления.

Когда Вы запускаете Образцовый Экран Резервной копии/Восстановления Файла, следующий экран появляется. Это

экран позволяет Вам выбирать резервную копию или восстановление.



рисунок 8.1 Выбор экрана резервной копии/восстановления

#### Резервная копия

Если Вы выбираете резервную копию, появляется следующий экран.

Backup	×
BATCH_MODEL.ml BoilerFlow.ml btntest.ml chartest.ml DS_System.ml parification_modell.ml	F Flat model F Vser F Local
	○ Model ○ Submodel
*	Beckup Cancel

рисунок 8.2 Экран резервной копии

Когда Вы выбираете User/Local и Model/Submodel плоской модели, файлы моделей отображаются в списке в левой стороне. Выберите из списка файл, для которого Вы хотите выполнить резервную копию. Кнопка возврат <Backup> становится действующей.

В резервной копии, А:\ диск - значение по умолчанию. Когда Вы нажимаете на кнопку <Backup>, появляется диалог спрашивающий Вас о готовности дисковвода. После этого появляется экран для выбор папки, чтобы выполнить резервную копию (Рисунок 8.2).



рисунка 8.3 Экран выбора папки резервной копии

Определите диск и папку для исполнения резервной копии и щелкните на кнопке <Save> (сохранить), чтобы запустить резервное копирование. Для образцового файла, скопированного успешно, отменяется выбор в списке рисунка 4.30.

## Восстановление

Если Вы выбираете восстановление, появляется следующий экран. Однако, если Вы исполняете операцию восстановления в то время как выполняются операции OIS, появляется предупреждение, чтобы запросить Вас установить в образцовый файл обслуживание и следующий экран не появляются, так как образцовые файлы должны быть сделаны эквивалентными.



рисунок 8.4 Экран восстановления

Выберите User/Local и Model/Submodel плоской модели для первого адресата, который нужно восстановить, и нажимать на кнопку <Restore> (восстановить).

При восстановлении, A:\ диск - значение по умолчанию. Когда Вы нажимаете на кнопку Restore>, появляется диалог, спрашивающий Вас, готов ли дисковод для гибких дискет. Тогда появляется экран, чтобы выбрать папку/файл для выполнения восстановления (Рисунок 8.4).

Open	? 🗙
Look jn:	🖃 3½ Floppy (A:)
DEMO_K	AIGAI.1
📃 DEMO_K	AIGAI.2
🔲 DEMO_K	AIGAI.3
I	
File <u>n</u> ame:	*.m1pen
<b>F1</b> ()	
Files of type:	M1 hie (*m1) Cancel

рисунок 8.5 Экран выбора папки/файла

.

Определите папку/файл, чтобы выполнить восстановление и щелкните на кнопке <Open> (открыть), чтобы запустить восстановление. Для Образцового файла, восстановленного успешно, отменяется выбор в списке рисунка 4.32

# 9. Обслуживание образцовых файлов

Экран обслуживания образцовых файлов – экран для резервирования или установки образцовых файлов, используемых SLGMS в графическом экране, не используя редактора. В этом экране, выберите резервную копию или установку и щелкните на кнопке <Execute> (выполнить), чтобы запустить функции.

Model file mainter	nance		×
	-Operation © Backup	O Install	
	Execution	Cancel	

Рисунок 9.1 Экран моделирования обслуживания файла

## • Функция резервирования

Резервная копия - функция, которая копирует образцовые файлы, установленные в OIS, запуская экран выбора других носителей типа гибких дисков.

• Функция установки

Установка – функция для установки файлов из других носителей типа гибких дисков в основой модуль.

Установка аналогична функции установки графического редактора, где образцовые файлы копируются ко всем OIS в той же самой сети.

## Резервная копия

Когда Вы выбираете резервную копию в экране обслуживания образцовых файлов, появляется следующий диалог.

Backup	×
Backup model type	
📀 Plat model	
C Model	
C Submodel	Select
Selection	
Backed up to	
	8s, set
No executor	Cancel

рисунок 9.2 Диалог резервной копии

Чтобы выполнять резервную копию, следуйте за шагами ниже.

1) Выбрать одну из плоских моделей, моделей, или подмоделей для типа образцового файла.

2) Нажать на кнопку <Select> (выбор).

3) Результат появляется в списке результата выбора.

4) Выбрать каталог устройства, чтобы сохранить модель, выбранную в шагах 1 - 3, используя кнопку < Выбор адресат резервной копии >( <Select Backup Destination>).

Результат появляется в окне редактирования.

5) Нажать на кнопку <Execute> (выполнить).

Резервные запуски. Каждый раз резервная копия завершает содержание списка результата выбора, который становится модифицированным.

#### ССЫЛКА:

Вы не можете использовать выбор адресата резервной копии, если Вы не выбрали файл модели. Вы не можете использовать кнопку <Execute>, если Вы не выбрали образцовый файл и адресата резервирования.

#### Инсталляция

Когда Вы выбираете установку в экране обслуживания образцовых файлов и выполняете это, появляется следующий диалог.

Install
-Install model selection
@ Flat model
C Nodel
C Submodel
Installed to
Selection
Selection
Baccupe Cancel

Рисунок 9.3 Диалог установки

Чтобы выполнять инсталляцию, следуйте за шагами ниже.

1) Выбрать одну из плоских моделей, моделей, или подмоделей для типа образцового файла.

- 2) Нажать на кнопку <Select Model File> (< Выбор файла модели >).
- 3) Результат появляется в списке результата выбора.

4) Нажать на кнопку <Execute> (выполнить).

Инсталляционные запуски. Каждый раз инсталляция завершает содержание список результата выбора, становящегося модифицированным. Функция установки, устанавливает выбранный образцовый файл во все OIS's в той же самой сети.

#### ССЫЛКА:

Вы не можете использовать кнопку < Execute>, если Вы не выбрали образцовый файл.
## Приложение 1. Динамическая ссылка

### Динамическое действие

Динамические функции выполнены через действия, которые определяют изменения {замены}, приложенные к объектам {целям} в соответствие с изменениями {заменами}, сделанными в прикладных переменных. Действия, используемые в динамических атрибутах упомянуты ниже.

#### Таблица А1.1 Динамические Действия (1)

Категория	Действие	Описание Действия
Изменяемый атрибут	bcolor int closed int detect int Ecoloer int estyle int ewidth real fcolor int foir int	атрибут цвета фона закрытие объекта{цели} края. 0 = не закрытый, 1 = закрытый. возможность обнаружения 0 = не возможно, 1 = возможно. цвет рамки {фрейма} стиль рамки {фрейма} ширина Рамки{Фрейма} заполняющий цвет
верхний		руководство заполнения . 0 – нижний край, т – левый, 2 –
-	finter int	край, 3 = право. образец заполнения 0 = передача, 1 = отдельный цвет, 2 =
образец,		
	fpercent real fstyle int filled int mcolor int msize real msytle int stext [int]real[string]	<ul> <li>степень заполнения. От 0 до 100 %</li> <li>стиль заполнения. Зависит от рабочего места.</li> <li>Заполнение закрытого объекта. 0 = не заполнено, 1 = заполнено.</li> <li>цвет маркера.</li> <li>размер маркера.</li> <li>стиль маркера.</li> <li>Текстовое содержание. Параметр вещественного чиспа</li> </ul>
отформати-	Stext [intredistring]	текстовое содержание. Параметр вещественного числа,
	.format-строка. talign x y tcolor int tfont int theight real tpath int tprec int trepl [int real string] .form at-string. tsize x y vis int vispart int	рованный .format-строкой. тип должен совпасть с кодом в строке символа управления форматом. Выравнивание текста (x, y) (параметр целый) цвет текста шрифт текста высота текста направление текста 1 = право, 2 = вниз. точность текста. 0 = растр, 1 = вектор, 2 = Hershey (семиугольник) То же самое как в stext текстовый размер (x, y) (целое число) отображение: 0 = невидимый, 1 = видимый, 2 = принудительное обновление. Отображены части индексированные int. Индексиро- ванные части в группе будут отображены. Другие части не будут отображены.

#### Таблица А1.1 Динамические действия (2)

Категория	Действие	Описание Действия
деформация	arclegth real	реальный угол, показывает длину дуги или круга
	move x y	относительное движение (параметр реальное число).
	movex real	Относительное движение к оси х
	movey real	Относительное движение к оси у
	radius real	Радиус круга, круга или дуги
	rotate real	Относительное выравнивание по центру на контрольной точке
	rtoz real	Так же, как при вращении.
	scale real	Однородное расширение или сжатие, центрирующееся на контрольной точке
	scalex real	Однородное расширение или сжатие направления X оси только центрированное на контрольной точке

	scaley real	Однородное расширение или сжатие направления <b>у</b> оси только центрированное на контрольной точке
	scaley real	Угол, чтобы стать основой дуги или круга.
Указанные не	bacherase	Стирает и перерисовывает части в группе или модели. Примитив
действия	call functionptr()	затрагивается функциями Call, уже определенными в приложении, типа sin (),abs
(),	dbflag int	или функции, определенные пользователем в приложении. Программная двойная буферизация допускается (1) или отключена
	dynarray	Абсолютный индекс массива сделан пригодным для
использования.	Dynerase noerase,	G_DYNARRAY флажок набора. То же самое как в batcherase если сделаны решительные изменения{замены}, устанавливает
метку	redraw userdata string userword int	к объектам до стирания Перерисовка без стирания. Устанавливает пользовательское поле данных в строке. Устанавливает пользовательское поле данных в int.

Двойной буферный флажок сохранен вместе с объектом в файле. После того, как сделают набор, флажки должны быть сброшены. Поэтому, относительно двойной буферизации, повторное выполнение этого действия бессмысленно, если объект отмечен. Другими словами, длительное выполнение динамического атрибута

dbfag1 с объектом {целью} бессмысленно. Исполните эту операцию в течение инициализации если возможно.

#### Строка символа управления форматом

Команда добавления динамики к текстовому объекту - ввод в следующем формате: stext<argument>.format-string.

Введите переменную ссылку или выражение после ключевого слова .stext.. Альтернативная переменная ссылка может быть включена в выражения. Ключевое слово .trepl. может использоваться вместо .stext..

Строки символа управления форматом всегда заключаются в двойные кавычки и могут содержать два

типа элементов, символьные строки непосредственно и формат кодирования, чтобы печатать значения переменных или результаты выражения. Символьные строки будут напечатаны как отображено, а код формата определит метод печати значений параметра.

.l. (символ нижнего регистра алфавита L) будет использоваться с кодами d, я, u, o, х и X с двойными целыми переменными точности.

Таблица ниже кратко описывает их коды.

Тип	код	описание
Целое Переменные	%d, %l	Описывает целую переменную как десятичные числа с положительным или отрицательным символом.
	%u	Описывают целую переменную как десятичные числа без положительного или отрицательного символа.
	%o	Описывает целую переменную как восьмеричное число без незначащего 0
	%#o	Описывает целую переменную как восьмеричное число с незначащим 0
	%x,&X,%p	Описывает целую переменную как шестнадцатеричные числа, используя .abcdef. для %х и %р и .ABCDEF. для %Х.
	%#X	Описывает целую переменную как шестнадцатеричные числа{номера},
используя		
		.ABCDEF. и префиксы .0X. к другим результатам кроме 0.

#### Таблица А1.2 Формат управляющих кодов (1)

%с	Описывает как символы.
%ld, %li	Описывает двойную целую переменную точности как десятичные числа с положительным или отрицательный символ.
%lu	Описывает двойную целую переменную точности как десятичные числа без 0 в головке (незначащий ноль).
%lo	Описывает двойную целую переменную точности как восьмеричные числа без
	головке (незначащий ноль).
%lx, %lX	Описывает двойную целую переменную точности как шестнадцатеричные
	используя .abcdef. для %х и .ABCDEF. для %Х.
%f с плавающей	Описывает переменную с плавающей точкой как десятичные числа с точкой и положительным или отрицательным символом.
%s, %E	Описывает переменную с плавающей точкой в показателе экспоненты, типа 1.2345е+01 для %е и 1.2345Е+01 для %Е.
%g, %G	Описывает меньшую переменную с плавающей точкой, используя %f или %e
	%Е для %G).
%s	Описывает как символьная строка до .\0. достигнут или число символов, показанное точность напечатана.
%% \ \n	Описывает символ процентного отношения. «Escapes». кавычки в символьных строках. создание несколько строк текста.
	%c %ld, %li %lu %lo %lx, %lX %f плавающей %s, %E %g, %G %s %s %% %s

ОБРАТИТЕ ВНИМАНИЕ: символьный строковый массив Размера 1 должен использовать %1.1s как код формата.

Минимальная ширина поля и невысокая точность могут быть определены после символа процента

и перед символами типа переменной. Определение минимальной ширины поля, обнуляет пробелы, если не достаточно ширины для печати числового значения. Количество цифр, отображенных после десятичной точки, выбирается при определении точности.

Например, минимальная ширина, включая десятичную точку установлена пять цифр, и две цифры после десятичной точки установлены для %5.2f.

#### Операторы выражения

Операторы, содержавшиеся в динамических атрибутах и эффектах, показаны ниже.

#### Таблица А1.3 Операторы выражения

Категории Оператора	оператор	описание
Арифметические	+	Добавление
Операторы	-	Вычитание
	*	Умножение
	**	Умножение мощности
	/	Деление
	%	Остаток (деление модуля)
Логические	!	NOT (логическое HE)
Операторы	>	Больше чем
	<	Меньший чем
	> =	Больше чем или равный (больше чем)
	<=	Меньший чем или равный (меньше чем)
	==	Равный
	! =	Не равный
	&&	И
	11	Логическое ИЛИ
Операция адреса	&	Адрес of

& оператор в синтаксисе динамических атрибутов передает переменный адрес. В интерпретации процедуры (GISMO), параметры функций пользовательского повторного вызова запрашивают от процедуры (GISMO) ссылку на передачу библиотеки SL-GMS. Вызов повторный и вызов значения (только функции С) поддерживаются для пользователя функции повторного вызова после действий запроса. В динамических атрибутах пользовательской функции повторного вызова будет выполнен вызов по ссылке, если & символ не предустановлен параметр. Если & символ предустановлен, будет выполнен повторный вызов.

#### ССЫЛКА:

& оператор подчинен к временным ограничениям. & оператор подобен С языку программирования.

В динамических атрибутах SL-GMS, & оператор будет необходим также при выдаче адреса переменной типа массив.

#### Арифметика и Логические операторы

Арифметические операторы устанавливают результаты из арифметических операций. Логические операторы устанавливают .1. если объект {задача} истинный (да) (например, 10>1), или 0 если объект {задача} ложный (нет).

## Функции библиотеки для С стандарта, распознанные{признанные} динамическим атрибутом

Те функции, которые можно назвать внутренней динамикой и описать, упомянуты ниже:

Таблица А1.4 Функции биб	пиотеки для С стандарта
Функция	устанавливаемое значение

sin(arg)	РЕАЛЬНЫЙ установлен
cos(arg)	РЕАЛЬНЫЙ установлен
tan(arg)	РЕАЛЬНЫЙ установлен
asin(arg)	РЕАЛЬНЫЙ установлен
acos(arg)	РЕАЛЬНЫЙ установлен
atan(arg)	РЕАЛЬНЫЙ установлен
atan2(arg1,arg2)	РЕАЛЬНЫЙ установлен
exp(arg)	РЕАЛЬНЫЙ установлен
log(arg)	РЕАЛЬНЫЙ установлен
log10(arg)	РЕАЛЬНЫЙ установлен
pow(arg1,arg2)	РЕАЛЬНЫЙ установлен
sqrt(arg)	РЕАЛЬНЫЙ установлен
floor(arg)	РЕАЛЬНЫЙ установлен
ceil(arg)	РЕАЛЬНЫЙ, установлен
fabs(arg)	ЦЕЛОЕ ЧИСЛО установлен
hypot (arg1, arg2)	РЕАЛЬНЫЙ установлен
strcmp (arg1, arg2)	ЦЕЛОЕ ЧИСЛО установлен
strncmp (arg1, arg2, n)	ЦЕЛОЕ ЧИСЛО установлен
strlen (arg1)	ЦЕЛОЕ ЧИСЛО установлен

5

# Приложение 2. Зарезервированные слова SL-GMS

Ниже упомянуты SL-GMS зарезервированные слова. Эти зарезервированные слова не могут использоваться как названия {имена} объекта {цели} или имена переменной.

act	col	ecoler	fpercent
align	color	endc	fpie
amove	comp	endcomp	flie2
amovex	coordbase	endif	fpoly
arclength	coordscale	endg	frec
arotz	csave	endgroup	free
ascale	cspline	endm	fsec2
asclex	ctb	endmodel	free
ascaley	current	endv	fsect
bqackgrflag	cwflag	endview	fsty
batcherase	ebflag	estyle	fstyle
bcol	deact	etime	fuect
bcolor	decomp	ewidth	get
bitmap	deface	extcenter	graphaxis
bitmapflag	defaults	extent	graphtrace
cd	defu	face	grid
center	degroup	fcir2	group
cgmget	delay	feire	gsave
cgmsave	demask	fcol	height
chdir	detect	fcolor	immu
cir2	direction	fdir	input
circ	dump	fdirection	inst
circle	dupe	fgroupflag	inst2
cl	dynarray	fgrouptype	lcol
clear	dyninit	fiiled	lcolor
clip	dynprop	fillgroup	line
clone	dynswich	fint	lsty
closed	dyntext	finter	lstyle
closergroun	dynupdate	ont	lwid
lwidth	offset	rmovey	updd
mlget	pan	rotz	upde
mlsave	path	rrect	userdata
mlxget	pause	rrtoz	upde
m2get	pcfrect	rscale	valuelimits
m2optsave	percent	rscalex	vfile

Таблица A2.1 SL-GMS Зарезервированные слова (1)

Таблица A2.1 SL-GMS Зарезервированные слова (2)

m2save	pie	rscalry	view
m2xget	pie2	save	vis
majorspacing	pitdata	scale	vrest
mark	pitdata	sec2	vrest
marker	poly	seය	wscale
mask	polygon	sect	wind
mcol	port	sector	workst
kdelay	preview	set	wsport
merge	proj	sfgtouptype	wswind
minorspacing	projevt	size	xget
model	quit	spline	xpoints
modist	radius	startangle	xtran
modinst2	rastop	status	xvaluelimits
moffset	rect	stext	yvaluelimits
move	rectheight	stress	zoom
mrotz	rectwidth	t∞l	zoomin
mscale	redraw	t∞lor	aoomount
msize	refpoint	text	zpr
msty	renamedvars	theight	zps
mstyle	reschk	time	
mtran	resol	time	
mtran0	rfrect	tran0	
nil	rmove	upd	
noerase	rmovex	update	

# Приложение 3. Стандартная подмодель

Эта глава перечисляет стандартные подмодели и описывает то, что они отображают по сигналам ошибок доступа тэга.

### Список стандартных подмоделей

В этом разделе перечислены описания функций для подмоделей, которые обеспечены как базовые, их образцовые имена, имена файлов, диаграммы моделей, описания переменных.

Что описано в списке переменных, может быть установлено, используя переменную,

переименовывающую функцию графического редактора. Есть два типа переменных; числовой тип и строковый тип. Для числового типа, установленные числовые значения переменной,

переименовыва-ют функцию. С другой стороны, для строкового типа должны использоваться строки, для переимено-вания функции.

#### Таблица А3.1 Стандартные подмодели (1)

Имя модели	имя файла	страница
Дисплей сигнального списка	almrev	151
Дисплей аварийной ситуации насоса	alm3StsPump	151
Изменение цвета состояния насоса	And2Pump	152
Отображение стрелки ошибки 1 (тонкая)	Arrow1	153
Отображение стрелки ошибки 2 (толстая)	Arrow2	153
Кнопка записи константы атома тэга	AtomCDataPutBtn	154
Кнопка установки произвольного атома тэга	AtomDataSetBtn	155
Труба Вентури 1	Bencury1	156
Турбина 1	Blower1	156
Турбина 2	Blower2	157
Турбина 3	Blower3	158
Катушка 1	Coil 1	158
Конденсатор 1	Condenser1	159
Резервуар круглый	Circle Tank	159
Пересечение ворот 1	CrossingGate1	160
Пересечение ворот 2	CrossingGate2	160
Пересечение ворот 3	CrossingGate3	161
Пересечение ворот 4	CrossingGate4	162
Дисплей значения 1	Cur_Valur1	162
Дисплей значения 2 (с фоновым цветом)	Cur_Value2	162

#### Таблица А3.1 Стандартные подмодели (2)

Имя модели	имя файла	страница
Дисплей значения 1 с CNF процессом	CurV_AlmCNF1	163
Дисплей значения 2 с CNF процессом (с фоновым цветом)	CuVr_AlmCNF2	163
Дисплей значения 3 с CNF процессом	CurV_AImCNF_Plus1	164
Дисплей значения 4 с CNF процессом (с фоновым цветом)	CurV_AlmCNF_Plus2	165
Дисплей значения 5 с CNF процессом	CurV_AImCNF_PlusR1	165
Дисплей значения 6 с CNF процессом (с фоновым цветом)	CurV_AlmCNF_PlusR2	166
Цветное сигнальное значение 1	CurV_AlmCol1	167
Цветное сигнальное значение 2 (с фоновым цветом)	CurV_AImCol2	167
DE	DE1	168
Измеритель плотности	Densitymeter1	168
Измеритель плотности 2 (с Датчиком)	Densitymeter2	169
Кнопка дисплея окна диалога	DialogBtn	169
DO кнопка выхода из диалога	DialogDO	170
DO кнопка вывода	DirectDO	172
Заземление	Earth	173
Имя модели	имя файла	страница

Вентилятор 1	Fan1	173
Электромагнитный измеритель потока 1	Flowmeter_EM1	174
Электромагнитный измеритель потока 2 (с Датчиком)	Flowmeter_EM2	174
Измеритель потока (диафрагма) 1	Flowmeter_OF1	175
Газотурбинный измеритель потока 1	Floemeter_TB1	175
Горизонтальная гистограмма (барграф)	HbarGraph	176
Бункер 1	Hoppa1	176
Исторический граф тенденции 1 для произвольного тэга	HTrd_Entry1	177
Исторический граф тенденции (тренд)	1 HTrd1	178
Исторический дисплей графа тенденции	2 HTrd2	179
Отдельный генератор мощности 1	IndependPW1	179
Дисплей ошибки уровня 1	LevelTraiangle1	180
MOF 1	MOF1	181
Горизонтальный ползунок	MVPTslider	181
Кнопка расширения экрана	NewScrnBtn	182
OR изменение цвета состояния насоса	Or2Pump	183
Отображение тревоги точки	PointAlarm	183
Указатель кнопки расширения экрана	PontBtn	184
Дисплей параметра точки	PointParm	184
Объединение 1 (Установить)	Pool_Fix1	185
Объединение 2 (Установить)	Pool_Fix2	186
Разъединение 1 (Установить)	Pool_FixDoom1	186
Объединение 1 (Переменная)	Pool_Var1	187
Объединение 2 (Переменная)	Pool_Var2	187
Разъединение 1 (Переменная)	Pool_VarDoom1	188

#### Таблица АЗ.1 Стандартные Подмодели (3)

Имя модели	имя файла	страница
Кнопка набора данных (для установки параметра точки)	PTParmSetBtn	188
Граф тенденции 1 для произвольного тэга	PtTrd Entry1	189
Граф тенденции 1 для дисплея точки	PtTrd1	190
Граф тенденции 2 для дисплея точки	PtTrd2	191
Машинный насос1	Pump Engine1	191
Моторный Насос 1	Pump Motor1	192
Hacoc 1	Pump1	192
Hacoc 2	Pump2	193
Hacoc 3	Pump3	194
Три цветных вертикальных графа	PValmBar	194
Получение сигнала окончания	ReciveEL1	195
Датчик 1	Sensor1	196
Датчик 2	Sensor2	196
Вертикальный ползунок	Svslider	197
Переключатель 1	Switch1	197
Переключатель 2	Switch2	198
Образец дисплея тэга	TagPattern	198
Дисплей измерителя	TagPVmeter,	199
	TagPFImeter	
Кнопка отображения диалога моделей дисплея тэга	TagPtnBtn	200
Левая кнопка запроса модели тега	TagPtnBtnL8	200
Правая кнопка запроса модели тега	TagPtnBtnR8	201
Кнопка отображения расширения тега	TagScrnBtn	202
Трансформатор 1	Transformer1	202
Трансформатор 2	Transformer2	203
Трансформатор 3	Transformer3	204
Трансформатор 4	Transformer4	204
Трансформатор 5	Transformer5	205
Трансформатор 6	Transformer6	206
Перевозка (транспортировка)1	Truck1	206
Запирающий (отсекающий) клапан 1	Valve_Cutoff1	207
Запирающий (отсекающий) клапан 2	Valve_Cutoff2	207
Регулирующий клапан 1	Value_Var1	208
Имя модели	имя файла	страница
Широкий клапан 1	Value_Wide1	208
Клапан 1	Valve1	209
Вертикальная гистограмма (барграф)	VbarGraph	209

Кнопка отображения диалога ХОК модели тэга	XorTagPtnBtn	210
Граф ломаной линии (4 ломаных линии)	Polyline4XY	211
Граф ломаной линии (заполненный + 3 ломаных линии)	PolyFill_3XY	212
Граф ломаной линии (гистограмма + 3 ломаных линии)	PolyBar_3XY	214

#### Таблица А3.1 Стандартные Подмодели (4)

Имя модели	имя файла	страница	
Радиальная диаграмма	radial	215	
Граф ломаной линии (6 ломаных линий)	Polyline6XY	216	
ISA символ	ISÁ	217	
Трехмерный канал	pipe3D	221	
Трехмерный насос	pump3D	222	
Трехмерный клапан	valve3D	222	
Трехмерный датчик	sensor3D	223	
Трехмерный котел	Boiler3D	223	
Трехмерный реактор	ReactorK	224	

## Сигнальный дисплей списка

### almrev

• Диаграмма модели

255	75-4-	K a s	(1 / 60)	75一直整生器合計:1251	
	p84,0111				4
- 1	p84,0821				_
1	100,000000000000		上上開具来	Calif pant	
+	pb 4,4 822				
6	ph 6,6 822				
	program	IN BUILD OF STREET			
2	ph 6,6870		<b>a</b> k		
2	program	IN BUSIC OFFICE	- <b></b>		
	projento	IN BUILD OF DET	- <b></b>		
18	projense				
	program	IN BUILD OF FIT	- <b></b>		
12	DE 4.0 854	IN READ OF STREET			
12	projentes	IN BUILD OF FT	- <b></b>		
16	004.0851	IN READ OF STREET	- <b></b>		
18	ph 4.4 855	and the Address	46		
18	004.0852	IN READ OF STREET			
17	ph 4.4 855	and a second state of the	46		
12	p84(014	and a star fulfater	<b>36</b>		
15	044.072	1999 Aug 1999	46		
24	264,0890	and the same fulfation	46		▼.

- Установка переменной
- Описание Функции

Отображает сигнальный дисплей списка.

## Дисплей аварийной ситуации насоса Alm3Sts1Pump

• Диаграмма модели



• Установка Переменной

AomTAGNO	Строковый
AlmBit1	Целочисленная
AlmBit2	Целочисленная
AlmBit3	Целочисленная

Номер тэга дисплея сигнальный Отображает тревогу бит номер 1 (от 0) Отображает тревогу бит номер 2 (от 0) Отображает тревогу бит номер 3 (от 0) 
 StsTAGNO
 Строковый

 StsBit
 Целочисленный

 ONColor
 Целое число

 OFFColor
 Целое число

Номер тэга дисплея состояния Номер бита дисплея состояния (от 0) Цвет включен Цвет выключен

• Описание Функции

Отображает насос в сигнальном цвете дисплея самого высокого приоритета для трех уровней тревоги

тэгов дисплея. Если все три уровня нормальны, отображение насоса с ВКЛ\ВЫКЛ цветом состояния определяется тэгом дисплея состояния. Любой тэг может быть определен как сигнального тэг дисплея. Тэги, где существуют DI сигналы (Flatom), могут быть определены как тэга дисплея состояния.

### AND изменения цвета состояния насоса

And2Pump

Arrow1

• Диаграмма модели



• Установка Переменной

ONColor	Целое число
OFFColor	Целое число
DATA1	Строка
Bit1	Целое число
DATA2	Строка
Bit2	Целое число

Цвет включен Цвет выключен Атом тега 1 Атом тега 1 номер бита (от 0) Атом тега 2 Атом тега 2 номер бита (от 0)

• Описание Функции

Отображает насос в различных цветах, основанных на И условии этих двух переменных. Когда оба

ОN, насос находится в ОN Цвете. Если один из них OFF, цвет дисплея выключен. Для атома тэга, FI, FO, СПУСКАЛСЯ, STIT, или МЕЛЬКАТЬ, может использоваться. (Пример) .PB40001. FI.

## Отображение стрелки ошибки 1 (Тонкая)

• Диаграмма модели



• Установка Переменной

TAGNO	Строковый
BitNo	Целочисленный
OnColor	Целочисленный
OffColor	Целочисленный

номер тега задания проверки состояния номер вита (от 0) атом тега задания включающий цвет состояния атом тега задания выключающий цвет состояния

• Описание Функции

Отображается уровень ошибки, типа ошибок уровня водны.

Должен быть определен номер тэга, являющегося заданием проверки состояния операции, и номер бита (от0) FI атома. Если состояние указанного атома тэга ON, используется OnColor. Если OFF, используется OffColor. При появлении сигнала тревоги символ отображен, мигая.

(Примечание) Должен быть определен номер цифрового тега для задания проверки состояния. Для BitNo, любой номер от 0. Для FI атомов тэга задания может быть определён - 1.

## Отображение стрелки ошибки 2 (Толстая)

Arrow2

• Диаграмма модели



• Установка Переменной

TAGNO	Строковый
BitNo	Целочисленный
OnColor	Целочисленный
OffColor	Целочисленный

номер тега задания проверки состояния номер вита (от 0) атом тега задания включающий цвет состояния атом тега задания выключающий цвет состояния

• Описание Функции

Отображается уровень ошибки, типа ошибок уровня водны.

Должен быть определен номер тэга, являющегося заданием проверки состояния операции, и номер бита (от0) FI атома. Если состояние указанного атома тэга ON, используется OnColor. Если OFF, используется OffColor. При появлении сигнала тревоги символ отображен, мигая.

(Примечание) Должен быть определен номер цифрового тега для задания проверки состояния. Для BitNo, любой номер от 0. Для FI атомов тэга задания может быть определён - 1.

## Атом тэга кнопки записи константы

## AtomCDataPutBtn

• Диаграмма модели

#### • Установка Переменной

TITLE	строка
HEIGHT	вещественная
MSG	строка
TAGATOM1	строка
DATA1	вещественная
TAGATOM2	строка
DATA2	вещественная
TAGATOM3	строка
DATA3	целое
TAGATOM4	строковый
DATA4	целое

кнопки строки заголовка размер символа заголовка состояние (статус) строки сообщения имя атома входного тега 1 константа для атома входного тэга 1 имя атома входного тега 2 константа для атома входного тэга 2 имя атома входного тега 3 константа для атома входного тэга 3 имя атома входного тега 4 константа для атома входного тэга 4

• Описание Функции

Пишет константу указанному атому тэга. Возможна запись до двух реальных атомов тэга и двух целочисленных атомов тэгов. Если имя атома тэга является NULL, запись не выполняется.

## Кнопка установки произвольного атома тэга AtomDataSetBtn

• Диаграмма модели



#### • Установка Переменной

строка	кнопка символа заголовка
Реальный	размер символа заголовка
строка	состояния символа сообщения
строка	имя атома входного тега
целое	тип данных
целое	количество входных цифр
целое	десятичные разряды
целое	тип проверки ввода
Реальное	максимальное значение для проверки входа (максимальное
	значение символа, когда тип строка)
Реальное	минимальное значение для проверки входа (минимальное
	значение символа, когда тип строка)
	строка Реальный строка строка целое целое целое целое Реальное Реальное

#### • Описание Функции

Отображение и проверка данных, устанавливающих диалог с данным атомом тэга, основана на проверке метода ввода, входного типа данных, максимального и минимального значения для входа.

#### Тип данных

1: Целочисленный Тип 2: Реальный Тип 3: Строковый Тип

Количество входных разрядов

Используемое, если тип данных - 1 или 2. Если тип данных - 3, максимальное количество вводимых символов (для строки) - количество входных разрядов (DTSIZE).

Десятичный разряд Используемый, если Тип данных - 2.

Метод проверки 1: ≤ минимума X ≤ Макс 2: Минимум <X <Максимум 2: ≤ минимума X <Макс 4: Минимум <X ≤ Макс

(Примечание) Это определение для строкового типа конечно (ограничено). Во время отображения диалога установки данных,

кнопка заблокирована. Для одной кнопки могут быть отображены только одни данные. В этой подмодели не рассматриваются - операция отключения состояния(БЛОКИРОВКА) (LOCK), (SCAN) (ПРОСМОТР), и режим (модификация) операции. Будьте внимательным, когда используете эту кнопку для тэга процесса.

## Труба Вентури 1

Bencury1

• Диаграмма модели



• Установка Переменной

None - нет переменной

• Описание Функции

Устанавливается модель {Рисунок}.

## Турбина 1

• Диаграмма модели



• Установка Переменной

TAGNO	Строковый
BitNo	Целочисленный

Проверка номера тега задающего состояние бит номер 1 (от 0)

• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.



#### • Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

## Blower1

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.



• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

## Катушка 1

• Диаграмма модели



• Установка Переменной

нет переменной

• Описание Функции

Устанавливается модель {рисунок}.

## Конденсатор 1

• Диаграмма модели



• Установка Переменной

#### Нет переменной

• Описание Функции

Condenser1



Устанавливается модель {рисунок}.

## Круглый резервуар

## CircleTank

• Диаграмма модели



• Установка Переменной

ТАGNO строка номер тега

• Описание Функции

Отображает уровень воды в резервуаре. Основано на значении тэга процесса, указанного TAGNO. Высокий уровень воды взят как 100 %, а низкий уровень взят как 0 %, в зависимости от текущего значенияпроцесса.

## Пересечение ворот 1

• Диаграмма модела



• Установка Переменной

TAGNO	Строковый
BitNo	Целочисленный

Проверка номера тега задающего состояние бит номер 1 (от 0)

CrossingGate1

• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

## Пересечение Вворот 2



• Диаграмма модели



• Установка Переменной

TAGNO BitNo Строковый Целочисленный Проверка номера тега задающего состояние бит номер 1 (от 0)

#### • Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

## Пересечение ворот 3

• Диаграмма модели



• Установка Переменной

TAGNO	Строковый
BitNo	Целочисленный

Проверка номера тега задающего состояние бит номер 1 (от 0)

• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

## Пересечение ворот 4

• Диаграмма модели



\* \* \* \* \* \* \* \* \* \*

• Установка Переменной

Нет переменной

• Описание Функции

Устанавливается модель {Рисунок}.

## Дисплей значения 1

- Диаграмма модели
- Установка Переменной

TAGNO DispColor

строка целое номер тега задания цвет дисплея

CrossingGate4

CrossingGate3

Cur\_Value1

#### • Описание Функции

Отображает текущее значение в цвете указанного тэга, указанном DispColor. Используется для набор значения десятичного разряда тэга. Числовое значение отображается с правым выравниванием в области дисплея.

(Примечание) В качестве задания номера тэга должен быть определен аналоговый тэг.

## Дисплей значения 2 (с фоновым цветом) Cur\_Value2

• Диаграмма модели



TAGNO	строка
DispColor	целое
BackColor	Целое

номер тега задания цвет дисплея цвет фона

• Описание Функции

Отображает текущее значение в цвете указанного тэга, указанном DispColor. Фоновая область дисплея числового значения отображена в цвете, указанном в BackColor. Для десятичного разряда числа, используется набор значения для заданного тэга. Числовое значение отображено с правым выравниванием в области дисплея.

(Примечание) В качестве задания номера тэга должен быть определен аналоговый тэг.

## Дисплей Значения 1 с CNF процессом



• Диаграмма модели

\* \* \* \* \* \* \* \* \* \*

• Установка Переменной

ТАБОО строка

номер тега задания

• Описание Функции

Отображает текущее значение указанного тэга в текущем цвете аварийной ситуации. Это задание для отображения CNF/LO процесса.

Для заданного тэга используется набор значения десятичных разрядов. Числовое значение отображается с правым выравниванием в области дисплея.

(Примечание) В качестве задания номера тэга должен быть определен аналоговый тэг.

## Дисплей значения 2 с CNF процессом (с цветом фона)



• Диаграмма модели



• Установка Переменной

TAGNO

строка

BackColor Целое • Описание Функции цвет фона

Отображает текущее значение указанного тэга в текущем цвете аварийной ситуации. Фон области дисплея числового значения отображается в цвете, указанном BackColor. Это задание для отображения CNF/LO процесса.

Для заданного тэга используется набор значения десятичных разрядов.

Числовое значение отображается с правым выравниванием в области дисплея.

(Примечание) В качестве задания номера тэга должен быть определен аналоговый тэг.

## Дисплей значения 3 с CNF процессом CurV\_AlmCNF\_Plus1

• Диаграмма модели

\* \* \* \* \* \* \* \* \* \*

• Установка Переменной

TAGNO	строка	номер тега задания
SCOL	целое	включение цвета просмотра
MCOL	целое	выключение цвета сигнального монитора

• Описание Функции

Отображает текущее значение указанного тэга в цвете дисплея согласно следующим приоритетам: неподтвержденная тревога> просмотр установок> подтвержденная тревога> сигнальный монитор OFF> неподтвержденная нормально тревога и нормально подтвержденная тревога, отображены в цвете аварийной ситуации. Установка просмотра OFF, отображена в цвете, указанном SCOL, и сигнальный монитор OFF, отображен в цвете определенном MCOL. Это задание для отображения CNF/LO процесса. Для заданного тэга используется набор значения десятичных разрядов. Числовое значение отображается с правым выравниванием в области дисплея.

(Примечание) В качестве задания номера тэга должен быть определен аналоговый тэг.

## Дисплей значения 4 с CNF CurV\_AlmCNF\_Plus2 процессом (с Цветом фона)

• Диаграмма модели



• Установка Переменной

TAGNO	строка
BackColor	Целое
SCOL	целое
MCOL	целое

номер тега задания цвет фона включение цвета просмотра выключение цвета сигнального монитор

• Описание Функции

Отображает текущее значение указанного тэга в цвете дисплея согласно следующим приоритетам. неподтвержденная тревога> выключение просмотра > подтвержденная тревога> сигнальный монитор выключен> неподтвержденная нормально тревога. «Подтвержденная тревога» и «норма», отображены в цвете аварийной ситуации. Установка «просмотр OFF», отображен в цвете, указанном SCOL, а «сигнальный монитор OFF» отображен в цвете заданном MCOL. Фон и область дисплея числового значения отображены в цвете заданном BackColor. Это задание для отображения CNF/LO процесса. Для заданного тэга используется набор значения десятичных разрядов.

Числовое значение отображается с правым выравниванием в области дисплея.

(Примечание) В качестве задания номера тэга должен быть определен аналоговый тэг.

## Дисплей значения 5 с CNF процессом

CurV\_AlmCNF\_PlusR1

• Диаграмма модели



• Установка Переменной

строка	номер тега задания
целое	просмотр, установленного выключенного цвета
целое	просмотр установленного включенного цвета
целое	отключение цвета сигнального монитора
целое	включение цвета сигнального монитора
	строка целое целое целое целое

• Описание Функции

Отображает текущее значение цвета аварийной ситуации указанного в тэге. В правой части числового дисплея отображен заполненный прямоугольник, который указывает сигнальное состояние монитора (верхняя часть) и просмотр, установленного состояние (нижняя часть).

Сигнальное состояние монитора: Отображается в цвете, указанном MOFCOL, когда сигнальный

	монитор выключен Огг.
	Отображается в цвете, указанном MONCOL, когда сигнальный монитор включен ON.
	Если общедоступная установка сигнальный монитор OFF, или
там	
	существует не менее одного OFF состояния сигнальных
параметров	
	настройки индивидуального монитора, сигнальное состояние
мони-	
	тора определяется как OFF.
Просмотр статуса:	Отображается в цвете, указанном SOFCOL, когда установка просмотра OFF.
	Отображается в цвете, указанном SONCOL, когда установка просмотра ON.

Это задание для отображения CNF/LO процесса. Для заданного тэга используется набор значения десятичных разрядов.

Числовое значение отображается с правым выравниванием в области дисплея.

(Примечание) В качестве задания номера тэга должен быть определен аналоговый тэг.

## Дисплей значения 6 с CNF процессом (с цветом фона)

CurV\_AlmCNF\_PlusR2



#### • Установка Переменной

TAGNO	строка	номер тега задания
BackColor	Целое	цвет фона
SOFCOL	целое	просмотр, установленного выключенного цвета
SONCOL	целое	просмотр установленного включенного цвета
MOFCOL	целое	отключение цвета сигнального монитора
MONCOL	целое	включение цвета сигнального монитора

• Описание Функции

Отображает текущее значение цвета аварийной ситуации указанного в тэге. В правой части числового дисплея отображен заполненный прямоугольник, который указывает сигнальное состояние монитора (верхняя часть) и просмотр, установленного состояние (нижняя часть).

Сигнальное состояние монитора: Отображается в цвете, указанном MOFCOL, когда сигнальный монитор выключен OFF. Отображается в цвете, указанном MONCOL, когда сигнальный монитор включен ON. Если общедоступная установка сигнальный монитор OFF, или там существует не менее одного OFF состояния сигнальных параметров настройки индивидуального монитора, сигнальное состояние монитора определяется как OFF. Просмотр статуса: Отображается в цвете, указанном SOFCOL, когда установка просмотра OFF. Отображается в цвете, указанном SONCOL, когда установка просмотра ОЛ. Фон и область дисплея числового значения отображены в цвете, указанном BackColor.

Это задание для отображения CNF/LO процесса. Для заданного тэга используется набор значения десятичных разрядов.

Числовое значение отображается с правым выравниванием в области дисплея.

(Примечание) В качестве задания номера тэга должен быть определен аналоговый тэг.

## Сигнальный цветной дисплей значения 1

CurV AlmCol1

• Диаграмма модели

\* \* \* \* \* \* \* \* \* \*

- Установка Переменной
- ТАGNO строка задаёт номер тега

• Описание Функции

Отображает текущее значение указанного тэга в текущем цвете аварийной ситуации. Для десятичных разрядов используется набор значения для целевого тэга. Числовое значение отображается с правым выравниванием в области дисплея.

((Примечание) В качестве задания номера тэга должен быть определен аналоговый тэг.

## Сигнальный цветной дисплей значения 2 (с Цветом фона)



• Диаграмма модели



• Установка Переменной

TAGNO строка BackColor Целое задаёт номер тега цвет фона

• Описание Функции

Отображает текущее значение указанного тэга в текущем цвете аварийной ситуации. Для десятичных разрядов используется набор значения для целевого тэга. Числовое значение отображается с правым выравниванием в области дисплея. Фон и область дисплея числового значения отображены в цвете, указанном BackColor

((Примечание) В качестве задания номера тэга должен быть определен аналоговый тэг.



• Установка Переменной

нет переменной

• Описание Функции

Устанавливает модель {Рисунок}.

#### измеритель плотности

• Диаграмма модели



• Установка Переменной

нет переменной

• Описание Функции

Устанавливает модель {Рисунок}.

## Измеритель плотности 2 (с датчиком)

Densitymeter2

Densitymeter1

• Диаграмма модели



• Установка Переменной

нет переменной

• Описание Функции

Устанавливает модель {Рисунок}.

## Кнопка отображения окна диалога

## DialogBtn

• Диаграмма модели



• Установка Переменной

TITLE	строка	кнопка титульной строки
HEIGHT	Реальная	размер символов титульной строки
MESSAGE	строка	статус панели сообщений
Х	целое	позиция дисплея диалога Х координата
Y	целое	позиция дисплея диалога Ү координата
W	целое	ширина диалога
Н	целое	высота диалога
BTN	целое	тип кнопки панели
SCRN	строка	экран меню или клавиша {Ключ} Элемента {Пункта}
NO	целое	номер экрана

• Описание Функции

Отображает окно диалога и указанный в диалоге экран. Символы на кнопке-строке, указываются в соответствии с ЗАГОЛОВКОМ. Размер символов - размер, указанный в HEIGHT. Когда кнопка нажата, символы отображенные в строке состояния указываются в соответствии с MESSAGE. Позиция дисплея из диалога - (X, Y) - (X+W, Y+H). Модуль определен количеством точек. Значения X и Y- относительные позиции от верхнего левого угла окна, в котором расположена кнопка. ВTN указывает тип кнопки на инструментальной панели в диалоге.

-1: Ни один, 0: CNF/LO/EXE/ Дисплей предупреждения, 1: FW/BW/GB/ Дисплей предупреждения, 2: CNF/LO/EXE/FW/BW/GB/ Дисплей предупреждения, 3: Дисплей предупреждения, 4: CNF/LO/EXE/ Дисплей предупреждения /закрыть, 5: FW/BW/GB/П Дисплей предупреждения /закрыть,

6: CNF/LO/EXE/FW/BW/GB/Предупреждение Отображают/Закрывают, и 7: Дисплей предупреждения /закрыть.

Если значение, указанное W является меньшим чем полный размер кнопок на инструментальной панели, некоторые кнопки могут не отображаться . Экран, отображенный в диалоге определенный SCRN и NO. SCRN, указывает ключ экрана меню или клавиша {ключ} элемента {точки} и Экранный номер. Если клавиша {ключ} экрана меню определена в SCRN, экранный номер - номер

страницы экрана меню. Клавиши {ключи} экрана меню и клавиши {ключи} элемента {точки} из стандартного экрана.

ALMS	Дисплей списка тревоги (клавиша {ключ} пункта {точки})
HTRD	Экран Тенденции (клавиша {ключ} элемента {пункта пункта {точки})
PONT	Экран о отображения пункта {точки} (клавиша {ключ} элемента {пункта})
GRAM	Экран меню графических символов (клавиша {ключ} экрана меню)
GRAP	Графический экран (клавиша (ключ) пункта (точки))
MENU	Экран меню (клавиша {ключ} экрана меню)

Во время отображения диалога, кнопка заблокирована. Когда кнопка Dialog Display назначена на другой экран, предыдущий диалог удаляется в то же самое время.

### DO Кнопку вывода дисплея диалога

DialogDO

• Диаграмма модели



#### • Установка Переменной

TITLE	строка	кнопка титульной строки
HEIGHT	реальная	размер символов титульной строки
MESSAGE	строка	статус панели сообщений
Х	целое	позиция дисплея диалога Х координата
Y	целое	позиция дисплея диалога Ү координата
W	целое	ширина диалога
Н	целое	высота диалога
TAGNO_L1 ~ L4	строка	номера тега от 1 до 4 налево от, DO кнопки вывода.
NUM_L1 ~ L4	целое	DO номер от 1 до 4 налево от DO кнопки вывода.
TAGNO_R1 ~ R4	строка	номера тега от 1 до 4 направо от, DO кнопки вывода.
NUM_R1 ~ R4	целое	DO номер от 1 до 4 направо от DO кнопки вывода.
MSG_L	строка	статус строки сообщения налево от DO кнопки вывода.
MSG_R	строка	статус строки сообщения направо от DO кнопки вывода.

#### • Описание Функции

Отображает, DO окно диалога вывода (образцовое имя DirectDO2). Символы на кнопке указаны в соответствии со строкой заголовка. Размер символов - размер, указанный высотой HEIGHT. Когда кнопка нажата, символы, указанные в соответствии с сообщением отображены в строке состояния. Позиция дисплея из диалога - (X, Y) - (X+W, Y+H). Модуль определен в точке №. Значения X и Y-относительные позиции от верхнего левого угла окна, в котором расположена кнопка. В инструментальной панели диалога отображена кнопка предупреждения.

Если значение, указанное W является меньшим, чем полный размер кнопок на инструментальной панели, некоторые кнопки могут быть не отображены. Во время отображения диалога кнопка заблокирована. Когда назначается развёртка другого экрана, данный диалог удаляется в то же самое время.

DO окно диалога вывода (используется модель: DirectDO2) отображается, как показано ниже.

P P	84_0001
運転	故障

В этом диалоге, две кнопки, которые могут определить вывод максимум четырех DO.s, изображений.

Номер тэга и DO (выходной) номер выбран в следующих комбинациях от TAGNO\_L1 до L4, от NUM\_L1 до L4, от TAGNO\_R1 до R4 и от NUM\_R1 до R4. DO номер NUM\_L1 для номера тэга TAGNO\_L1

DO номер NUM R4 для номера тэга TAGNO R4

Номер тэга TAG\_L1 используется для области заголовка диалога. Символы на левой кнопке символы индикатора, указаны в TAGNO\_L1 и NUM\_L1 (имя атома SNAM), и

символы на правой кнопке - символы индикатора, указаны в TAGNO\_R1 и NUM\_R1 (имя атома SNAM). Кнопки в диалоге не отображаются как заблокированные, если весь указанный тэг

является DO выхода состояния. Если не один номер тэга не определен, это задание для DO вывода.

## DO Кнопка выведите

DirectDO

• Диаграмма модели



• Установка Переменной

HEIGHT	реальная	размер символа заголовка
TAGNO1 ~ 4	строки	номер тега от DO вывод 1, до DO вывод 4
NUM1 ~ 4	целое	DO номер от DO вывод 1, до DO вывод 4
MSG	строка	статус строки сообщения

#### • Описание Функции

Исполняет, DO вывод из указанного тэга. Символы на кнопке - символы индикатора (SNAM) указанного TAG1 и NUM1. Размер символов - размер, указанный высотой HEIGHT. Когда кнопка нажата, символы, указанные в соответствии с сообщением отображены в строке состояния.

Тэг номер и DO номер определен в следующих комбинациях. DO номер NUM1 для номера тэга TAGNO1

DO номер NUM4 для номера тэга TAGNO4

Кнопки в диалоге отображены как заблокированные, если весь указанный тэг не встречается, DO выводном состоянии. Если не один номер тэга не определен, это задание для DO вывода. (Примечание) Когда указанный тэг находится в неподтвержденном состоянии, кнопка останова нежелательного мерцания (квитирования) на инструментальной панели становится доступной.

## Заземление Earth

• Диаграмма модели



• Установка Переменной

нет переменной

• Описание Функции

Установка модели рисунка.

## Вентилятор 1 Fan1

• Диаграмма модели



• Установка Переменной

TAGNO	Строковый	ном
BitNo	Целочисленный	бит

номера тега задающего проверку состояния бит номер 1 (от 0)

#### • Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

## Электромагнитный измеритель потока 1 Flowmeter\_EM1

• Диаграмма модели



• Установка Переменной

нет переменных

• Описание Функции

Устанавливается модель (рисунок).

Электромагнитный измеритель Flowmeter\_EM2 потока 2 (с датчиком) Flowmeter\_EM2



• Установка Переменной

Нет переменной

• Описание Функции

Установка модели (рисунок).

## Измеритель потока диафрагменный 1 Flowmeter\_OF1

• Диаграмма модели



• Установка Переменной

Нет переменных

• Описание Функции

Устанавливается модель {рисунок}.

## Газотурбинный измеритель потока 1 Flowmeter\_TB1

• Диаграмма модели



• Установка Переменной

Нет переменных

• Описание Функции

Устанавливается модель {рисунок}.

## Горизонтальная Гистограмма

• Диаграмма модели



• Установка Переменной

BackColor	целое	цвет фона гистограммы
BarColor	целое	цвет дисплея гистограммы
Data1	реальная/строка	данные дисплея 1 (реальная константа или атом тэга)
Data2	реальная/строка	данные дисплея 2 (реальная константа или атом тэга)
LowRange	реальная/строка	нижняя граница диапазона (реальная константа или атом
тега)		
UpRange	реальная/строка	верхняя граница диапазона (реальная константа или атом
тэга)		

• Описание Функции

Отображает данные дисплея в горизонтальной гистограмме (барграфе) между нижней границей диапазона и верхней границей диапазона.

Если данные 1 = нижняя граница диапазона, гистограмма левее.

Если данные 2 = верхняя граница диапазона, гистограмма правее.

Если BothData 1 и 2 определены, может быть создан диапазон разбиения, или граф МН-МL.

Бункер 1

Hoppa1

• Диаграмма модели



• Установка Переменной

Нет переменных

• Описание Функции

Устанавливается модель {рисунок}.

## Исторический граф тенденции (тренд) 1 Htyd\_Entry1 для произвольного тэга

• Диаграмма модели



• Установка Переменной

TAG1 строка PenColor1 целое тэг исторического тренда для пера графа 1 значение рисующее цвет для пера графа 1

TAG8	строка	тэг исторического тренда для пера графа 8
PenColor8	целое	значение рисующее цвет для пера графа 8
RangeLo	реальное	нижний диапазон графа
RangeHi	реальное	верхний диапазон графа

#### • Описание Функции

Отображает исторический граф тенденции для произвольного тэга (до восьми указанных точек). Цвет отображения исторического рисунка и графа для тэга, может быть определен для каждого пера рисунка. Диапазон отображенного графа также определен.

(Примечание) Для цифровых тэгов, позиция дисплея установлена согласно перьевому номеру.

## Исторический граф тенденции (тренд)1

HTrd1

• Диаграмма модели



• Установка Переменной

Группа целое Историческая тенденция номер группы

• Описание Функции

Отобразите указанную историческую группу тенденции тем же самым способом как граф тенденции, отображенный на экран исторической тенденции. Цвет рисунка для исторического тэга и графа для отображения определен в информации для каждой исторической группы тенденции.

(Примечание) Для цифровых тэгов, позиция дисплея установлена согласно перьевому номеру.

## Исторический дисплей графа тенденции 2

Htrd2



• Установка Переменной

Группа целое Историческая тенденция номер группы

#### • Описание Функции

Отобразите указанную историческую группу тенденции тем же самым способом как граф тенденции, отображенный на экран исторической тенденции. Цвет рисунка для исторического тэга и графа для отображения определен в информации для каждой исторической группы тенденции. Установленная информация пера номер 1 используется для выбора пера. Поэтому граф для исторического тэга использует указанное перо номер 1.

(Примечание) Для цифровых тэгов, позиция дисплея установлена согласно перьевому номеру.

## Отдельный генератор мощности 1



• Диаграмма модели



• Установка переменной

Строка	задание состояние номер тега
Целое	бит номер (от 0)

#### • Описание функции

Отображает состояние операции устройства. Номер тэга, который является заданием для проверки состояние операции и номер бита (от 0), FI атома, должны быть определены. Если состояние указанного атома тэга ON, устройство отображено красным цветом. Если состояние OFF, устройство отображено зеленым цветом.

(Примечание) Для задания проверки состояния должен быть определен номер цифрового тэга. Для BitNo, может быть определен любой номер от 0 до номера FI атома заданного тэга - 1.

## Дисплей ошибки уровня

1 LevelTraiangle1



• Установка переменной

TAGNO	строка	номер тега проверки статуса состояния
BitNO	целое	бит номер (от 0)
OnColor	целое	атом тега задание ON цветное отображение статуса
OffColor	целое	атом тега задание OFF цветное отображение статуса

#### • Описание Функции

отображаются ошибки уровней, типа ошибок уровня воды. Должны быть определены: номер тэга, который является заданием для проверки состояния операции и бит номер (от 0), FI атома. Если состояние указанного атома тэга ON, используется OnColor. Если состояние указанного атома тэга OFF, используется OffColor. Как только происходит тревога, символ отображается мигая.

(Примечание) Должен быть определен для адресата (задания) проверки состояния номер цифрового тэга.

Для BitNo может быть определен любой номер от 0 до номера FI атома заданного тэга - 1.

## MOF 1 MOF1

• Диаграмма модели



• Установка переменной

Нет переменных

• Описание Функции

Установка модели {Рисунка}.

### Горизонтальный ползунок

## **MVPTslider**

• Диаграмма модели



• Установка Переменной

целое
целое
реальное/строка
реальное/строка
реальное/строка

цвет отображения метки слайдера (ползунка) данные отображения (реальная константа или атом тэга) нижняя граница диапазона (реальная константа или атом тэга) верхняя граница диапазона (реальная константа или атом тэга)

#### • Описание Функции

Отображает данные дисплея в горизонтальном ползунке, с меткой скольжения, от нижней границы диапазона до верхней границы диапазона.

цвет фона

## Кнопка расширения экрана

## NewScrnBtn

• Диаграмма модели



• Установка переменной

TITLE	строка	кнопка символа заголовка
HEIGHT	реальное	размер символа заголовка
MSG	строка	статусная панель символов сообщения
SCRN	строка	экранное меню или клавиша {ключ} элемента {точки}
NO	целое	экранный номер

• Описание функции

Это - кнопка для экранного расширения. Символы на кнопке - строка, указанная в соответствии с заголовком

Размер символов - размер, указанный высотой (HEIGHT). Когда кнопка нажата, символы определенные в соответствии с сообщением отображены на строке состояния. Экран,

отображенный в диалоге определен SCRN, илиNO. SCRN указывает клавишу {ключ} меню экрана или клавишу {ключ} элемента {пункта}, а NO указывает экранный номер.

Если клавиша {ключ} экрана меню определена в SCRN, экранный номер - номер страницы экранного меню.

Клавиши {ключи} меню экрана и клавиши {ключи} элемента {пункта} стандартного экрана следующие:

Клавиша {ключ} элемента {пункта} отображения сигнального списка

Клавиша {ключ } элемента {пункта } точечного экрана

Клавиша {Ключ} экранного меню

Клавиша {Ключ} элемента {пункта} графического экрана

Клавиша {Ключ} экранного меню

(Примечание), чтобы развернуть экран, используйте TagScrnBtn или PointBtn.

#### OR изменение цвета состояния насоса

Or2Punp

• Диаграмма модели



• Установка Переменной

ONColor	целое	ON отображаемый цвет
OFFColor	целое	OFF отображаемый цвет
DATA1	строка	тег атом 1
Bit1	целое	тег атом 1 номер бита атома (от 0)
DATA2	строка	тег атом 2
Bit2	целое	тег атом 2 номер бита атома (от 0)

#### • Описание Функции

Отображает насос в различных цветах, основанных на ИЛИ условии этих двух переменных. Когда один из них ON, насос находится в отображаемом цвете. Если оба OFF, насос находится в OFF отображаемом цвете. Для атома тэга может использоваться FI, FO, ALIT, STIT, или МЕЛЬКАТЬ.

(Пример) "PB40001. FI"

## Отображение тревоги в точечном дисплее

PointAlarm

• Диаграмма модели



• Установка Переменной

ТАС строка номер тега

• Описание Функции

Отображает информацию тревоги пункта (точки) указанного номера тэга.

### Кнопка расширения точечного экрана

## PointBtn

• Диаграмма модели



• Установка Переменной

TITLE	строка
HEIGHT	реальная
MSG	строка
TAGNO	строка

кнопка строки заголовка размер символа заголовка статус панели строки сообщений номер тега

• Описание Функции

Это - кнопка для расширения точечного экрана. Символы на кнопке определяют символы строки заголовка. Размер символов - размер, указанный высотой (HEIGHT). Когда кнопка нажата, символы, указанные в соответствии с сообщением, отображаются на строке состояния.

(Примечание) Чтобы расшириться до графического экрана, экрана группы, или указать экран в зависимости от регистрации тэга, используйте TagScrnBtn.

## Дисплей параметра точки



•	Установка	Переменной
---	-----------	------------

ТАGNO строка номер тега

• Описание Функции

Отображает информацию параметра точки указанного номера тэга. Информация параметра точки напоминает следующее:

8-0 REGRAD	9-9	94 310K94	9-2
1 5569	r64.000	7 批批法示赦	**
2 2759	a a 3 1 4 0 0 MB 2 3	<ul> <li>NH#322平</li> </ul>	409123
2 471.00		第 75-4万次	975
4 他类图(8)	X		01
5 888749	10	🔟 Sui-susts	
5 法门实上	21	100000000000000000000000000000000000000	

## Объединение 1 (Установка)

• Диаграмма модели



• Установка переменной

Нет переменных

- Описание функции
- Установка модели {Рисунка}.

## Бассейн 2 (Установка)

• Диаграмма модели



• Установка переменной

Нет переменных

• Описание функции

Установка модели (Рисунка)

Pool\_Fix2



## Закрытый бассейн 1 (Установка) Pool\_FixDoom1

• Диаграмма модели



• Установка переменной

Нет переменных

• Описание функции

Установка модели (Рисунка)

## Бассейн 1 (Переменная)

## Pool\_Var1

• Диаграмма модели



• Установка Переменной

ТАGNO строка задание номера тега

• Описание Функции

Отображает уровень воды бассейна в гистограмме. Номер тега определяет значение уровня воды. Верхний предел диапазона взят за 100 %, а низкий предел диапазона взят как 0 % когда уровень воды отображен основанным на текущем значении.

(Примечание) Аналоговый тэг должен быть определен для заданного номера тэга.

## Объединение 2 (Переменная)





• Установка Переменной

ТАGNO строка задание номера тега

• Описание Функции

Отображает уровень воды бассейна в гистограмме. Номер тега определяет значение уровня воды. Верхний предел диапазона взят за 100 %, а низкий предел диапазона взят как 0 % когда уровень воды отображен основанным на текущем значении.

(Примечание) Аналоговый тэг должен быть определен для заданного номера тэга.

# Закрытый бассейн 1 (Переменная) PTParmSetBtn • Диаграмма модели • Установка Переменной ТАGNO строка задание номера тега

• Описание Функции

Отображает уровень воды бассейна в гистограмме. Номер тега определяет значение уровня воды. Верхний предел диапазона взят за 100 %, а низкий предел диапазона взят как 0 % когда уровень воды отображен основанным на текущем значении.

(Примечание) Аналоговый тэг должен быть определен для заданного номера тэга.

## Кнопка набора данных (для установки параметра точки)

PTParmSetBtn

• Диаграмма модели

• Установка переменной

строка	кнопка титульной строки
реальная	размер символа заголовка
строка	статус строки сообщения
строка	ввод имени тега
строка	ввод имени атома
	строка реальная строка строка строка

#### • Описание Функции

Отображает диалог установки данных, данного атома тэга и проверяет входные данные точки в базе данных. Во время установки данных кнопка заблокирована. Для одной кнопки может быть отображен только один диалог установки данных.

## Граф тенденции 1 для произвольного тэга

PtTrd\_Entry1

• Диаграмма модели



• Установка Переменной

TAG1	строка	назначение атома для пера графа 1
PenColor1	целое	цвет для пера графа 1
TAG8	строка	назначение атома для пера графа 8
PenColor8	целое	цвет для пера графа 8
RangeLo	реальное	нижнее ограничение диапазона графа
RangeHi	реальное	верхнее ограничение диапазона графа

• Описание Функции

Отображает граф тенденции для произвольного атома тэга (до восьми указанных пунктов {точек}).

Для каждого тега может быть определён цвет каждого пера. Пределы диапазона (шкалы) также определены

(Примечание) Для цифровых тэгов, позиция дисплея установлена согласно перьевому номеру.

## Граф тенденции 1 для дисплея точки


ATOM	an ao ao aminina 🖌 🖌	
ATON OCCORDOCCO ATON	······	
	ייייים את האת האת אחר איייים	
	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	5

ТАGNO строка номер тега

• Описание Функции

Отображает тег так же как в точечном дисплее Для каждого типа тэга определяется (задаётся) цвет графика. Используя ограничение диапазонов, определяют расположение графика тега.

(Примечание) Для цифровых тэгов, позиция дисплея установлена согласно перьевому номеру.

### Граф тенденции 2 для дисплея точки



• Установка Переменной

ТАGNО строка номер тега

• Описание Функции

Отображает тег так же как в точечном дисплее

Для каждого типа тэга определяется (задаётся) цвет графика. Используя ограничение диапазонов, определяют расположение графика тега.

(Примечание) Для цифровых тэгов, позиция дисплея установлена согласно перьевому номеру.

### Машинный насос 1

Pump\_Engine1

PtTrd2



TAGNO	Строковый
BitNo	Целочисленный

номера тега задающего состояние бит номер 1 (от 0)

Pump\_Motor1

• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

### Моторный насос 1

• Диаграмма модели



• Установка Переменной

TAGNOСтроковыйBitNoЦелочисленный

номера тега задающего состояние бит номер 1 (от 0)

Pump1

• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

## Hacoc 1

• Диаграмма модели



• Установка Переменной

Строковый

Целочисленный

TAGNO BitNo

номера тега задающего состояние бит номер 1 (от 0)

### • Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.



• Диаграмма модели



• Установка Переменной

TAGNO	Строковый
BitNo	Целочисленный

номера тега задающего состояние бит номер 1 (от 0)

• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

### Hacoc 3

• Диаграмма модели



• Установка Переменной

TAGNOСтроковыйBitNoЦелочисленный

номера тега задающего состояние бит номер 1 (от 0)

• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

### Pump3

Pump2

## Трёхцветный вертикальный граф

### PvalmBar

• Диаграмма модели



#### • Установка переменной

HHLLColor	Целое	PHH-PLL цвет области дисплея
HLCColor	Целое	PH-PL цвет области дисплея
HMLColor	Целое	Нормальный цвет области дисплея
RH	Реальная/Строка	RH данные (вещественная константа или атом тэга)
RL	Реальная/Строка	RL данные (вещественная константа или атом тэга)
RHH	Реальная/Строка	РНН данные (вещественная константа или атом тэга)
PH	Реальная/Строка	РН данные (вещественная константа или атом тэга)
MH	Реальная/Строка	МН данные (вещественная константа или атом тэга)
PLL	Реальная/Строка	PLL данные (вещественная константа или атом тэга)

### • Описание Функции

Отображает тревогу в зависимости от значения (PHH, PH, MH, PLL) PV в вертикальной гистограмме в трех цветах.

### Остановка

• Диаграмма модели



• Установка Переменной

нет переменных

• Описание Функции

Установка модели (рисунка).

## Датчик 1

• Диаграмма модели

Sensor1

### ReciveEL1



нет переменных

• Описание Функции

Установка модели (рисунка).

## Датчик 2

• Диаграмма модели



• Установка Переменной

нет переменных

• Описание Функции

Установка модели (рисунка).

### Вертикальный ползунок

• Образцовая Диаграмма



• Установка Переменной

BackColor	целое
MarkColor	целое
MVPT	реальное/строка
LowRange	реальное/строка
UpRange	реальное/строка

цвет фона цвет отображения метки слайдера (ползунка) данные отображения (реальная константа или атом тэга) нижняя граница диапазона (реальная константа или атом тэга) верхняя граница диапазона (реальная константа или атом тэга)

• Описание Функции

Отображает данные дисплея в вертикальном ползунке, с меткой скольжения, от нижней границы диапазона до верхней границы диапазона.

Переключатель 1

Sensor2

Svslider

Switch1

• Диаграмма модели



• Установка Переменной

нет переменных

• Описание Функции

## Установка модели (рисунка). Переключатель 2

• Диаграмма модели



• Установка Переменной

нет переменных

• Описание Функции

Установка модели (рисунка).

### Образец дисплея тэга

• Диаграмма модели



• Установка переменной

TAGNO Строка TtlDsp Целое

Номер тега Флажок дисплея заголовка

### • Описание Функции

Отображает образец указанного тэга. Если TtlDsp - 0, никакой заголовок не отображен. Если TtlDsp - 1, номер тэга отображен как экранный заголовок. Часть образца дисплея тэга изменяется в зависи-мости от типа тэга.

(Примечание) Для дисплея заголовка: заголовок, отображенный на экране последним, имеет более высокий приоритет.

### Дисплей измерителя

TagPVmeter, TagPFImeter

Switch2

### TagPattern

• Диаграмма модели



• Установка переменной

TAGNO Строка Номер тега

• Описание Функции

Отображает значение PV тэга, указанного TAGNO, в интервале между верхним и нижним пределом

диапазона. Если Тип Тэга - измеритель коэффициента мощности (PFI), используется модель TagPFImeter. Для тэгов с другим атомом PV, кроме этого, используется модель TagPVmeter.

### Кнопка отображения диалога модели дисплея тэга

## TagPtnBtn

• Диаграмма модели



• Установка Переменной

строка	титульная строка
реальный	Размер символа заголовка
строка	Панель статуса строки сообщения
целое	Х координата позиция дисплея диалога
целое	Ү координата позиции дисплея диалога
целое	Ширина диалога
целое	Высота диалога
строка	Номер тега
	строка реальный строка целое целое целое целое строка

• Описание Функции

Отображает окно диалога и указанный образец дисплея тэга в диалоге. Символы на кнопке строка, указанная в соответствии с заголовком. Размер символов - размер, указанный высотой (HEIGHT) .

Когда кнопка нажата, символы, указанные в соответствии с сообщением отображаются в строке состояния.

Отобразите позицию диалога - (X, Y) - (X+W, Y+H). Модуль определен в No точек. Значения из Х и У - относительные позиции от верхнего левого угла окна, в котором расположена кнопка. Отображается дисплей тэга, который удовлетворяет условию типа тэга, указанного TAGNO. В диалоге отображения модели тэга, кнопка CNF/LO/EXE и кнопка отображения предупреждения – всегда отображены на инструментальной панели. Если значение, указанное W является меньшим чем полный размер кнопок на инструментальная панель, некоторые кнопки не могут быть отображены. Во время отображения диалога кнопка заблокирована.

При развертке другого экрана, предыдущий диалог удаляется.

### Кнопка «влево» запроса диалога TagPtnBtnR8 моделей отображения тэга

• Диаграмма модели



• Установка Переменной

TITLE	строка
HEIGHT	реальная
MSG	строка
TAGNO	строка

кнопка символов заголовка размер символа заголовка статус строки сообщения номер тега

### • Описание Функции

Отображается восемь тэгов в порядке вызова, слева направо, для удаленного пульта оператора с 8 контурами управления. Ширина диалога - 1/8 ширины окна оператора. Символы на кнопке - строка, указанная в соответствии с заголовком. Размер символов - размер, указанный ВЫСОТОЙ (HEIGHT). Когда кнопка нажата символы, указанные в соответствии с сообщением отображаются в строке состояния. В то время как отображается диалог, кнопка заблокирована. Этот диалог может

будьте удален, используя кнопку Delete. Запросы после удаления отображены в позиции где диалог удален.

## Кнопка вызова «вправо» диалога моделей дисплея тэга

### TagPtnBtnR8

• Диаграмма модели



• Установка Переменной

TITLE	строка	кнопка символов заголовка
HEIGHT	реальная	размер символа заголовка
MSG	строка	статус строки сообщения
TAGNO	строка	номер тега

• Описание Функции

Отображается восемь тэгов в порядке вызова, справа налево, для удаленного пульта оператора с 8 контурами управления. Ширина диалога - 1/8 ширины окна оператора. Символы на кнопке - строка, указанная в соответствии с заголовком. Размер символов - размер, указанный ВЫСОТОЙ (HEIGHT). Когда кнопка нажата символы, указанные в соответствии с сообщением отображаются в строке состояния. В то время как отображается диалог, кнопка заблокирована. Этот диалог может

будьте удален, используя кнопку Delete. Запросы после удаления отображены в позиции где диалог удален.

### Кнопка развертки экрана тега

TagScrnBtn

- Диаграмма модели
- Установка Переменной

TITLE	строка	кнопка символов заголовка
HEIGHT	реальная	размер символа заголовка
MSG	строка	статус строки сообщения
TAGNO	строка	номер тега

• Описание Функции

Это - кнопка для развёртки на экране экрана тэга. Если любой графический экран зарегистрирован для тэга и экран существует, экранная развёртка выполняется в графическом экране. Если не сущест-вует никакого графического экрана, но группа зарегистрирована, развёртка экрана выполняется к экрану группы.

Если никакой графический экран не существует, и никакая группа не зарегистрирована, экранная развёртка выполняется в точечном экране.

Символы на кнопке - строка, указанная в соответствии с заголовком. Размер символов - размер, указанный ВЫСОТОЙ (HEIGHT). Когда кнопка нажата, отображаются символы, указанные в соответствии с сообщением в строке состояния.

(Примечание), чтобы всегда разворачивать точечный экран, используйте PointBtn.

## Трансформатор 1

### Transformer1

• Диаграмма модели



• Установка Переменной

UpperColor	Целое
часть)	
LowerColor	Целое
• Описание Фун	нкции

Цвет отображения первичной стороны (обмотки) (верхняя

Цвет отображения вторичной стороны (обмотки) (нижняя часть)

Устанавливает модель {Рисунок}.

Отображает первичную сторону (обмотку) (верхний круг) в цвете, указанном UpperColor. Отображает вторичную сторону (обмотку) (нижний круг) в цвете, указанном LowerColor.

### Трансформатор 2

Transformer2



UpperColor	Целое	Цвет отображения первичной стороны (обмотки) (верхняя
часть)		
LowerColor	Целое	Цвет отображения вторичной стороны (обмотки) (нижняя часть)

• Описание Функции

Устанавливает модель {Рисунок}.

Отображает первичную сторону (обмотку) (верхний круг и рисунок в нём) в цвете, указанном UpperColor.

Отображает вторичную сторону (обмотку) (нижний круг и рисунок в нём) в цвете, указанном LowerColor.

## Трансформатор 3

## Transformer3

• Диаграмма модели



• Установка Переменной

UpperColor часть)	Целое	Цвет отображения первичной стороны (обмотки) (верхняя
LowerColor	Целое	Цвет отображения вторичной стороны (обмотки) (нижняя часть)

### • Описание Функции

Устанавливает модель {Рисунок}.

Отображает первичную сторону (обмотку) (верхний круг и рисунок в нём) в цвете, указанном UpperColor.

Отображает вторичную сторону (обмотку) (нижний круг и рисунок в нём) в цвете, указанном LowerColor.

## Трансформатор 4

Transformer4



UpperColor	Целое	Цвет отображения первичной стороны (обмотки) (верхняя
часть)		
LowerColor	Целое	Цвет отображения вторичной стороны (обмотки) (нижняя часть)

• Описание Функции

Устанавливает модель {Рисунок}.

Отображает первичную сторону (обмотку) (верхний круг и рисунок в нём) в цвете, указанном UpperColor.

Отображает вторичную сторону (обмотку) (нижний круг и рисунок в нём) в цвете, указанном LowerColor.

## Трансформатор 5

### Transformer5

• Диаграмма модели



• Установка Переменной

UpperColor	Целое	Цвет отображения первичной стороны (обмотки) (верхняя
LowerColor	Целое	Цвет отображения вторичной стороны (обмотки) (нижняя часть)

### • Описание Функции

Устанавливает модель {Рисунок}.

Отображает первичную сторону (обмотку) (верхний круг и рисунок в нём) в цвете, указанном UpperColor.

Отображает вторичную сторону (обмотку) (нижний круг и рисунок в нём) в цвете, указанном LowerColor.

## Трансформатор 6

## Transformer6



UpperColor	Целое	Цвет отображения первичной стороны (обмотки) (верхняя
LowerColor	Целое	Цвет отображения вторичной стороны (обмотки) (нижняя часть)

### • Описание Функции

Устанавливает модель {Рисунок}. Отображает первичную сторону (обмотку) (верхний круг и рисунок в нём) в цвете, указанном UpperColor. Отображает вторичную сторону (обмотку) (нижний круг и рисунок в нём) в цвете, указанном LowerColor.

### Перевозка 1

• Диаграмма модели



• Установка Переменной

нет переменных

• Описание Функции

Установка модели {Рисунка}.

### Отсекающий клапан 1

• Диаграмма модели





• Установка Переменной

нет переменных

## Truck1

• Описание Функции

Установка модели {Рисунка}.

### Отсекающий клапан 2

• Диаграмма модели





• Установка Переменной

нет переменных

• Описание Функции

Установка модели {Рисунка}.

### Регулирующий клапан 1

• Диаграмма модели

Valv\_Var1



• Установка Переменной

TAGNOСтроковыйBitNoЦелочисленный

номера тега задающего состояние бит номер 1 (от 0)

• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

### Широкий клапан 1



• Диаграмма модели



• Установка Переменной

TAGNO BitNo Строковый Целочисленный номера тега задающего состояние бит номер 1 (от 0)

### • Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

### Клапан 1

• Диаграмма модели





• Установка Переменной

TAGNO	Строковый
BitNo	Целочисленный

номера тега задающего состояние бит номер 1 (от 0)

• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, модель отображена красным. Если состояние ВЫКЛЮЧЕНО OFF, модель отображена зеленым.

(Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга. Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

### Вертикальная гистограмма (барграф) VbarGraph

• Диаграмма модели



• Установка Переменной

BackColorцелоеBarColorцелоеData1реальная/строкаData2реальная/строкаLowRangeреальная/строкатега)телона

цвет фона гистограммы цвет дисплея гистограммы данные дисплея 1 (реальная константа или атом тэга) данные дисплея 2 (реальная константа или атом тэга) нижняя граница диапазона (реальная константа или атом UpRange тэга) реальная/строка

• Описание Функции

Отображает данные дисплея в вертикальной гистограмме (барграфе) между нижней границей диапазона и верхней границей диапазона.

Если данные 1 = нижняя граница диапазона, гистограмма ниже.

Если данные 2 = верхняя граница диапазона, гистограмма выше.

Если BothData 1 и 2 определены, может быть создан диапазон разбиения, или граф МН-МL.

## Кнопка вызова диалога XOR модели дисплея тэга XorTagPtnBtn

• Диаграмма модели



• Установка Переменной

иия
диалога
диалога

### • Описание Функции

Отображает окно диалога и указанный образец дисплея тэга в диалоге. Символы на кнопке строка, указанная в соответствии с заголовком. Размер символов - размер, указанный высотой (HEIGHT).

Когда кнопка нажата, символы, указанные в соответствии с сообщением отображаются в строке состояния.

Отобразите позицию диалога - (X, Y) - (X+W, Y+H). Модуль определен в No точек. Значения из X и Y - относительные позиции от верхнего левого угла окна, в котором расположена кнопка. Отображается дисплей тэга, который удовлетворяет условию типа тэга, указанного TAGNO. В диалоге отображения модели тэга, кнопка CNF/LO/EXE и кнопка отображения предупреждения – всегда отображены на инструментальной панели. Если значение, указанное W является меньшим чем полный размер кнопок на инструментальной панели, некоторые кнопки не могут быть отображены.

Во время отображения диалога кнопка заблокирована. При развёртке другого экрана старый диалог закрывается

Если используются несколько XorTagPtnBtn.s, отображен только один диалог. Для модели TagPtnBtn, XOR дисплей диалога не выполняется. Однако, для XorTagPtnBtn, XOR выполнена.

### Граф ломаной линии (4 ломаных линии) Polyline4XY



BackCol	Целое	Цвет фона графа
GridCol	Целое	Фоновый цвет отображения сетки
PolyMAX	Целое	Максимальное значение данных ломаной линии [n]
PolyTAG1	Строка	Спецификация 1 пример: Тэга Данных. % float_1. ДАННЫЕ [0-63].
PolyTAG2	Строка	Спецификация 2 пример: Тэга Данных. % float_2. ДАННЫЕ [0-5].
PolyTAG3	Строка	Спецификация 3 пример: Тэга Данных. % float_3. ДАННЫЕ [0-5].
PolyTAG4	Строка	Спецификация 4 пример: Тэга Данных. % float_3. ДАННЫЕ [6-11].
Xformat	Строка	Х формат координат отображения метки (значение по умолчанию:. %g.)
XlineCol	Целое	Цвет отображения координат Х
XMAX	Реальное	Максимальное значение координаты X [Образцовая Диаграмма = 30]
XMIN	Реальное	Минимальное значение координаты Х [Образцовая Диаграмма = 0]
XmPoint	Реальное	Масштаб координаты X [Образцовая Диаграмма = 5]
XsPoint	Реальное	Под масштаб координаты Х [Образцовая Диаграмма = 5]
Yformat	Строка	У формат координат отображения метки (Значение по умолчанию:. %5.
1f.)	_	
YLineCol	Целое	Цвет отображения координат Ү
YMAX	Реальное	Максимальное значение координаты Ү [Образцовая Диаграмма = 500]
YMIN	Реальное	Минимальное значение координаты У [Образцовая Диаграмма = 0]
YmPoint	Реальное	Масштаб координаты Ү [Образцовая Диаграмма = 100]
YsPoint	Реальное	Под масштаб координаты Ү [Образцовая Диаграмма = 50]

• Описание Функции

Отображает максимум четырех пары данных ломаной линии в графе. Структура данных тэга - как показано ниже.

(заполненный + 3 ломаных линии)

ДАННЫЕ [0] = Цвета отображени ДАННЫЕ [2] = X1	я графа ДАННЫЕ[1] = Номер ДАННЫЕ [3] = Y1	данных ломаной линии [n]
ДАННЫЕ [2n] = Xn	ДАННЫЕ [2n+1] = Yr	1
DATA[0] = Graph Display Color DATA[2] = X1	DATA[1] = No. of Polyline Data [n] DATA[3] = Y1	
DATA[2n]= Xn	DATA[2n+1] = Yn	
Граф ломаной линии		PolyFill_3XY



BackCol	Целое	Цвет фона графа
GridCol	Целое	Фоновый цвет отображения сетки
PolyMAX	Целое	Максимальное значение данных ломаной линии [n]
PolyTAG1	Строка	Спецификация 1 пример: Тэга Данных. % float_1. ДАННЫЕ [0-63].
PolyTAG2	Строка	Спецификация 2 пример: Тэга Данных. % float_2. ДАННЫЕ [0-5].
PolyTAG3	Строка	Спецификация 3 пример: Тэга Данных. % float_3. ДАННЫЕ [0-5].
PolyTAG4	Строка	Спецификация 4 пример: Тэга Данных. % float_3. ДАННЫЕ [6-11].
Xformat	Строка	X координата отображения метки (Диаграмма модели = . %02g:00.)
XlineCol	Целое	Цвет отображения координат Х
XMAX	Реальное	Максимальное значение координаты X [Диаграмма модели = 30]
XMIN	Реальное	Минимальное значение координаты Х [Диаграмма модели = 0]
XmPoint	Реальное	Масштаб координаты X [Диаграмма модели = 5]
XsPoint	Реальное	Под масштаб координаты Х [Диаграмма модели = 1]
Yformat	Строка	Y координата отображения метки (Диаграмма модели =. %g KW.)
YLineCol	Целое	Цвет отображения координат Ү
YMAX	Реальное	Максимальное значение координаты Ү [Диаграмма модели = 500]
YMIN	Реальное	Минимальное значение координаты Ү [Диаграмма модели = 0]
YmPoint	Реальное	Масштаб координаты Ү [Диаграмма модели = 50]
YsPoint	Реальное	Под масштаб координаты Ү [Диаграмма модели = 25]

### • Описание Функции

Отображает максимум четырех пара данных ломаной линии в графе. Данные 1 заполнены, а данные 2, 3, и 4 отображены ломаными линиями. Структура тэга данных та же самая как .PolyLine4XY..

### Граф ломаной линии (дисплей гистограммы + 3 ломаных линии)

## PolyBar\_3XY

• Диаграмма модели



• Установка Переменной

BackCol Целое Цвет фона графа

GridCol	Целое	Фоновый цвет отображения сетки
PolyMAX	Целое	Максимальное значение данных ломаной линии [n]
PolyTAG1	Строка	Спецификация 1 пример: Тэга Данных. % float_1. ДАННЫЕ [0-63].
PolyTAG2	Строка	Спецификация 2 пример: Тэга Данных. % float_2. ДАННЫЕ [0-5].
PolyTAG3	Строка	Спецификация 3 пример: Тэга Данных. % float_3. ДАННЫЕ [0-5].
PolyTAG4	Строка	Спецификация 4 пример: Тэга Данных. % float_3. ДАННЫЕ [6-11].
Xformat	Строка	Х формат координат отображения метки (значение по умолчанию:. %g.)
XlineCol	Целое	Цвет отображения координат Х
XMAX	Реальное	Максимальное значение координаты Х [Диаграмма модели = 30]
XMIN	Реальное	Минимальное значение координаты Х [Диаграмма модели = 0]
XmPoint	Реальное	Масштаб координаты Х [Диаграмма модели = 5]
XsPoint	Реальное	Под масштаб координаты X [Диаграмма модели = 5]
Yformat	Строка	У формат координат отображения метки (Значение по умолчанию:. %5.
1f.)		
YLineCol	Целое	Цвет отображения координат Ү
YMAX	Реальное	Максимальное значение координаты Ү [Диаграмма модели = 500]
YMIN	Реальное	Минимальное значение координаты Y [Диаграмма модели = 0]
YmPoint	Реальное	Масштаб координаты Ү [Диаграмма модели = 100]
YsPoint	Реальное	Под масштаб координаты Ү [Диаграмма модели = 25]

#### • Описание Функции

Отображает максимум четыре пары данных ломаной линии в графе. Данные 1 отображены в области

графа гистограммой, а данные 2, 3, и 4 отображены ломаными линиями. Ось X отображена в левой стороне. Структура тэга данных та же самая как .PolyLine4XY..

### Радиальная диаграмма

### radial

• Диаграмма модели



• Установка Переменной

BKCOLOR	Целое	Цвет фона графа круга
PolyTag	Строка	Номер тэга данных
XMAX	Целое	Максимальное значение

• Описание Функции

Отображает данные в радиальной диаграмме. Структура тэга данных - как показано ниже: ДАННЫЕ [0] = Цвета отображения графа ДАННЫЕ [2] = X1

ДАННЫЕ [2n] = Xn

ДАННЫЕ[1] = Номер данных ломаной линии [n] ДАННЫЕ [3] = Y1

ДАННЫЕ [2n+1] = Yn

Значения тэга данных являются координатными данными в пределах области с радиусом XMAX Линии координатной сетки сделаны так, чтобы они делите круг на 12 равных частей, но нет никакого ограничения для количества координатных данных.

### Граф ломаной линии (6 ломаных линий)

### PolyLine6XY

• Диаграмма модели



• Установка Переменной

BackCol	Целое	Цвет фона графа
GridCol	Целое	Фоновый цвет отображения сетки
PolyMAX	Целое	Максимальное значение данных ломаной линии [n]
PolyTAG1	Строка	Спецификация 1 пример: Тэга Данных. % float_1. ДАННЫЕ [0-63].
PolyTAG2	Строка	Спецификация 2 пример: Тэга Данных. % float_2. ДАННЫЕ [0-5].
PolyTAG3	Строка	Спецификация 3 пример: Тэга Данных. % float_3. ДАННЫЕ [0-5].
PolyTAG4	Строка	Спецификация 4 пример: Тэга Данных. % float_3. ДАННЫЕ [10-15].
PolyTAG5	Строка	Спецификация 5 пример: Тэга Данных. % float_3. ДАННЫЕ [0-5].
PolyTAG6	Строка	Спецификация 6 пример: Тэга Данных. % float_3. ДАННЫЕ [10-15].
Xformat	Строка	X координата метки (Диаграмма модели =. ~time \n00: ~ секунда.)
XlineCol	Целое	Цвет отображения координат Х
XMAX	Реальное	Максимальное значение координаты Х [Диаграмма модели = 30]
XMIN	Реальное	Минимальное значение координаты Х [Диаграмма модели = 0]
XmPoint	Реальное	Масштаб координаты Х [Диаграмма модели = 5]
XsPoint	Реальное	Под масштаб координаты X [Диаграмма модели = 1]
Yformat	Строка	У формат координат отображения метки (Диаграмма модели:=. %5.1f
KW.)		
YLineCol	Целое	Цвет отображения координат Ү
YMAX	Реальное	Максимальное значение координаты Ү [Диаграмма модели = 500]
YMIN	Реальное	Минимальное значение координаты Y [Диаграмма модели = 0]
YmPoint	Реальное	Масштаб координаты Ү [Диаграмма модели = 100]
YsPoint	Реальное	Под масштаб координаты Ү [Диаграмма модели = 50]

### • Описание Функции

Отображает максимум шесть пар данных ломаной линии в графе. Структура тэга данныхта же самая как .PolyLine4XY..

### ISA символ ISA\_

	Диаграмма	Имя	Пояснения
--	-----------	-----	-----------

	ISA_ACTR	ISA_ACTR Релейный Привод головок
	ISA_MATR	ISA_MATR Ручной Привод головок
$\Box$	ISA_TACT	ISA_ТАСТ Удушение Привода головок
<b></b>	ISA_BVLV	ISA_BVLV Клапан - бабочка
$\overline{\mathbf{b}}$	ISA_RVLV	ISA_RVLV Клапан Помощи{Облегчения}
	ISA_CVLV	ISA_CVLV Проверяют {Отмечают} Клапан
$\square$	ISA_VLV3	ISA_VLV3 3Way Тело Клапана
K	ISA_VLV3_ACTR	ISA_VLV3_ACTR 3Way Клапан с Релейным Приводом головок
M	ISA_VLV3_MATR	ISA_VLV3_MATR 3Way Клапан с Ручным Приводом головок
$\bowtie$	ISA_VLVE	ISA_VLVE Тело Клапана
	ISA_VLVE_ACTR	ISA_VLVE_ACTR Клапан с Релейным Приводом головок
Модель	Имя	Пояснения

k	ISA_VLVE_MATR	ISA_VLVE_MATR Клапан с Ручным Приводом
- F	ISA_VLVE_TACT	головок
	ISA_DTWR	ISA_VLVE_TACT Клапан с Удушением Привода головок
$\square$	ISA_RCTR	ISA_DTWR Башня Дистилляции
	ISA_VSSL	ISA_RCTR Реактор
Ŏ	ISA_JVSL	ISA_VSSL Судно
$\square$	ISA_ATNK	ISA_JVSL Покрытое кожухом Судно
	ISA_BINN	ISA_AINK Atomspheric Kopnyc {Pesepbyap}
	ISA_FTNK	ПЗА_ВПИК Лоток памяти {дранения}
	ISA_GHDR	ISA_FTNК Плавающий Корпус {Резервуар} Крыши
ρ	ISA_PVSL	ISA_GHDR Газовый Держатель
$\sum_{i=1}^{n}$	ISA_WHPR	
		ISA_WHPR Весят Карман

Модель	Имя	Пояснения

	ISA BLWR RT	
$\bigcirc$		ISA_BLWR_RT Трубач - Выход{Вывод} на Праве
$\overline{\bigcirc}$	ISA_BLWR_LF	ISA_BLWR_LF Трубач - Выход{Вывод} на Левом
R	ISA_PUMP_RT	ISA_PUMP_RT Накачка - Выход{Вывод} на Праве
$\mathbb{R}$	ISA_PUMP_LF	ISA_PUMP_LF Накачка - Выход{Вывод} на Левом
	ISA_TURB	ISA_TURB Турбина
	ISA_CMPR	ISA_CMPR Компрессор
	ISA_RECP	ISA_RECP Оплата Компрессора
-	ISA CNVR	ISA_CNVR Конвейер
$\overline{\otimes}$	ISA_RFDR	ISA_RFDR Ротационный Подающий лоток
- 00-	ISA_RSTD	ISA_RSTD Позиция Рулона
$\langle \square \rangle$	ISA_MILL	ISA_MILL Завод
	ISA_AGIT	ISA_AGIT Агитатор
Î	ISA_AGIT_MOTR	ISA_AGIT_MOTR Агитатор с Двигателем
		ISA IMIX Встроенный Смеситель
	ISA_IMIX ISA_SCNV	
	ISA_SUNV	ISA_SCNV Вворачивают Конвейер

Модель	Имя	Пояснения

	ISA CBRK	ISA CBRK Выключатель
Å	ISA_DLTA	ISA_DLTA Подключение{Связь} Дельты
	ISA_FUSE	ISA_FUSE Плавкий предохранитель
_/ <b>_</b>	ISA_MCTR	ISA_MCTR Руководство Contactor
$\bigcirc$	ISA_MOTR	ISA_MOTR Двигатель
$\square$	ISA_MOTR_BASE	ISA_MOTR_BASE Двигатель с Ядром
	ISA_STAT	ISA_STAT Заявляют Индикатор
ЗE	ISA_XFMR	ISA_XFMR Трансформатор
	ISA_WYEC	ISA_WYEC Подключение{Связь} УАЯ
	ISA_LFLT	ISA_LFLT Жидкий{Ликвидный} Фильтр
	ISA_EPCP	ISA_EPCP Электростатический Осадитель
	ISA_SCBR	ISA_SCBR Скребок
	ISA_VFLT	ISA_VFLT Vacum Фильтруют .ВР
$\Box$	ISA_CSEP	ISA_CSEP Разделитель Циклона
6	ISA_RSEP	ISA_RSEP Ротационный Разделитель
	ISA_SDRY	ISA_SDRY Распыляют Сушилку
$\overline{\diamond}$	ISA_XCHG1	ISA_XCHG1 Теплообменник

Модель	Имя	Пояснения

Ŕ	ISA_XCHG2	ISA_XCHG2 Теплообменник
	ISA_FURN	ISA_FURN Печь
	ISA_FAXR	ISA_FAXR Принудительный Воздушный
	ISA_KILN	Геплоооменник
$\overline{}$	ISA_CTWR	ISA_KILN Ротационная Печь
		ISA_CTWR Охлаждающаяся Башня
	ISA_EVPR	
	ISA_FNXR	
		ISA_FNXR Finned Теплоооменник

нет переменных

• Описание Функции

Установка модели {Рисунка}.

### Изображение трёхмерного канала

pipe3D\_

• Диаграмма модели



• Установка Переменной

нет переменных

• Описание Функции

Установка модели {Рисунка}.

## Изображение трёхмерного насоса pump3D\_



ТАG\_АТОМ строка

Название {имя} атома тега

• Описание Функции

Вращается, если данные, обозначенные атомом тэга, указанным TAG\_ATOM становятся 1. Если 0, вращение {циклический сдвиг}остановлен. Атом тэга определен в цифровом типе определен.

(Примечание) атом тэга должен быть атомом, который указывает состояние цифрового тэга.

## Изображение трёхмерного клапана valv3D\_

• Диаграмма модели



• Установка Переменной

TAGNO

номера тега

• Описание Функции

Отображает клапан в сигнальном цвете, указанном на клапане TAGNO. Если Вы не хотите отображать сигнальный цвет, Вы можете оставить TAGNO неопределенным.

### Изображение трёхмерного датчика sensor3D\_

• Диаграмма модели



строка



sensor3D H



sensor3D V

• Установка Переменной

TAGNO

строка

номера тега (только sonsor $3D_H$  и sensor $3D_V$ )

• Описание Функции

Отображает клапан в сигнальном цвете, указанном на датчике TAGNO. Если Вы не хотите отображать сигнальный цвет, Вы можете оставить TAGNO неопределенным.

### Изображение трёхмерного котла Boiler3D



FTAG	Строка	атом тэга отображения пламени
LTAG	Строка	номер тэга отображающий уровень воды

• Описание Функции

Отображает значение процесса как уровень воды в пределах диапазона тега, обозначенного LTAG. Если данные атома тэга, обозначенного FTAG, становятся 0, отображение пламени удаляется. Отображение пламени (включено/выключено) определяется состоянием цифрового тэга.

(Примечание) Аналоговый тип должен иметь только номер тэга, определенный для LTAG. Тэг атом определенный для FTAG, должен иметь цифровой тип.

### Изображение трёхмерного реактора

### ReactorK

• Диаграмма модели



• Установка Переменной

BitNo	Целое	бит номер (от 0)
TAGNO	Строка	номера тега

• Описание Функции

Отображает состояние операции устройства. Должны быть определены номер тэга и номер бита (от 0) FI атома, который является заданием для проверки состояния операции. Если состояние указанного атома тэга ON, реактор вращается. (Примечание) Заданием проверки состояния должен быть определен номер цифрового тэга.

Для BitNo может быть определен любой номер от 0 до Номера FI атомов тэга задания - 1.

## Подробный дисплей подмоделей по ошибкам доступа тэга

Этот раздел описывает подробности отображения для каждой подмодели, когда происходит ошибка доступа тэга.

В основном, когда происходит ошибка доступа тэга, кнопка отображения ошибки в окне, которое использует подмодель, становится доступной Если вы нажимаете на кнопку отображения ошибки, появляются список номеров тэга с ошибкой доступа и их кодами ошибки.

Ниже объяснена синхронизация возникновения ошибки доступа тэга, отображенных подробностей, и классификация подмоделей.

. Ошибка доступа тэга происходит, в то время как экран начинает отображаться.

. Ошибка доступа тэга происходит, когда экран используемой подмодели отображен.

Так как невозможно выполнить доступ, когда экран отображен, отображение подробного экрана тэга остается в начальном состоянии. Начальное состояние - состояние рисунка, когда подмодель только развёрнула в графическом редакторе. Числовое значение становится 0.

# Приложение 4. Стандартные функции повторного вызова

Эта глава описывает функционирование повторного вызова с динамическими свойствами, которые будут добавлены к различным объектам SL-GMS.

Есть три типа переменных; частные переменные, общественные переменные, и глобальные переменные. Они определены следующим образом:

### \* Частная переменная

Переменные, которые начинаются с \_р и эффективны только в модели. Если то же самое название {имя} используется в двух моделях, переменные рассматривают {считают} как различные переменные.

#### \* Общая переменная

Переменные, которые не начинаются с \_p и разделены (используются) среди моделей. Если они используются в подмодели, то обрабатываются, как изменение (переименование) адресата (задания). Если переменная создана в символах \$, символ (имя) в \$'s - переименовывающийся адресат (задание).

### \* Глобальная переменная

Переменные, которые начинаются с \_р и зарезервированы для системы. Это приложение описывает

глобальные переменные, зарезервированные для системы.

Как одно из действий, есть, функция повторного вызова. Функция повторного вызова описана как функция Языка С, где описываются имя функции и параметры.

Для объяснения определения параметра для каждой функции, используются следующие примечания:

Void	Переменная с неопределенным типом
void*	Указатель на переменную с неопределенным типом
int	Переменная целочисленного типа
int*	Указатель на переменную целочисленного типа
real	Переменная реального типа
real*	Указатель на переменную реального типа
char*	Указатель на переменную строкового типа

Некоторые действия допускаются только, когда используется кнопка мыши, а другие действия допускаются при изменении переменных.

Те, которые допускаются только, когда используется кнопка мыши, называются как специальная функция процедуры.

### Таблица А4.1 Глобальные переменные

Переменная INT Инициализация	страница 176
BGCOLOR Фоновый цвет	177
OPEMODE Режим доступа	177
ZOOM	177

Изменение масштаба окна

\_\_\_UPDATE Обновление

\_\_BLINKCT Мигание

## Таблица A4.2 Callbac Variable (переменные обратного вызова)

177

177

Переменная	страница
AddTagDot(char *tagno, int pos) DelTagDot(char *tagno) Добавление к строке номера тега	178
BitAnd(int bit1, int bit2) BitOr(int bit1, int bit2) BitXor(int bit1, int bit2) Результат логической операции	178
BreakString(char *str, int chars, int lines) Разделение строки на линии	178
ButtonChange(int color1, int color2) Изменение цвета рамки	178
ButtonHilite(int color1, int color2) ButtonHiliteChk(int color1, int color2, int exechk,) ButtonChk3(int color1, int color2, int backcol, int dispchk, int exechk) Отображение вкл. /выкл. кнопки	179
ButtonMessage(char *title1) ButtonMessage2(char *title1, char *title2) Описание функций кнопки	179
CheckVarInit(void *_pData) Проверка корректности чтения переменной	180
ChkDirect(int *_pChk, char *tag1, int num1,, char *tag4, int num4) Проверка статуса 4х выходов DO	180
ChgIntVar(void *_pSetPoint, int SetData) Набор целочисленной переменной	180
ChkMoniGrp(int area) Проверка категории сигнальной группы	180
ChkOPWMode(int opemode) Проверка операции доступа	180
DefTitleModel(int no, void *_pno, void *_ptitle) Проверка регистрации переменной	181
DefTitleModelChk(int no, void *_pchk)	181
DefVar(void *_pvar, char *tagatom, 1, int type, int array, int col, 1) DefV2(void *_pvar, char *tagatom) Чтение данных атома тега	181
DefVarInt(void *_pvar, int val) DefVarReal(void *_pvar, real val) DefVarStr(void *_pvar, char *val) Инициализация значения переменной	182
DialogInstance1(char *NewModel, int pX, int pY, int pW, int pH) Отображение указанной модели	182
DialogInstance2(char *NewModel, int pX, int pY, int pW, int pH, int pUsrBtn,int pDelflg, void *_pDlgBody)	182

Отображение модели с инструментальной панелью

Di *	alogInstance3(char *opwType, int no, int pX, int pY, int pW, int pH, int pUsrBtn,int pDelflg, void pDlgBody) гображение модели с инструментальной панелью	183
Di Ot	spAlarmCol(int type, int *_pSTIT, int *_pFLIT, int *_pAGRD, int TLPfg, int BlkCt) гображение сигнальной информации самого высокого приоритета	184
Di _p Ot	spAlarmColPlus(int type, int *_pSTIT, int *_pFLIT, int *_pAGRD, int _pSCN, int _pAMON, int* MON, int Scol, int Mcol, int TLPfg, int BlkCt) гображение сигнальной информации высокого приоритета	185
Di ** Ci	spAllAlarm(int type, int *_pSTIT, int *_pFLIT, int *_pAGRD,int *_pAMON, int *_pMON, char _pSNAM, int BlkCt) игнальные подробности экрана точки	186
Di 3a	spAreaName(int area) мена строкового объекта	186
Di Ot	spBlinkCol(int col1, int col2, int Cflg, int Bflg, int TLPfg, int BlkCt) гображение цвета дисплея	186
Di Bi Bi	spBlinkColSP(int col1, int col2, int *Stit, int *Flit, int BitNo1, int BitNo2, int BitNo3,int BitNo4, int tNo5, int BitNo6, int BitNo7, int BitNo8,, int TLPfg,int BlkCt) ыбор условий отображения	187
Di Ot	spCntrlMode(int *_pMod, int *_pMods, int is_Mods, int BkCol, int BlkCt) гображение символов контура регулирования	187
Di Гo	spHbar(real dt1, real dt2, real RLdt, real RHdt, real xmin, real xmax) ризонтальная гистограмма	188
Di Гa	spHslid(real Hdt, real RLdt, real RHdt, real xmin, real xmax) ризонтальный слайдер	188
Di Пр	spMofFrame(int type, int *_pAMON, int *_pMON, int BkCol) роверка переключения контроля	188
Di Ot	spRealData(real *_pData, int decml) гображение реальных данных	189
Di Di Be	spVbar(real dt1, real dt2, real RLdt, real RHdt, real ymin, real ymax) spVbarPFI(real dt1, real dt2, real RLdt, real RHdt, real ymin, real ymax) ертикальная гистограмма	189
Di Di Be	spVslid(real Vdt, real RLdt, real RHdt, real ymin, real ymax) spVslidPFI(real Vdt, real RLdt, real RHdt, real ymin, real ymax) ертикальный слайдер	189
Do Pa	oGraphPoint(char *tagno, int num) азвёртка графического экрана	190
Ge Ci	etAlmCol(int almgrd, int Blinkct) игнальный цвет уровня тревоги	190
Ge 3⊦	etAlmGrd(int *_pSTIT, int *_pFLIT, int *_pAGRD, int bitno) іачение уровня сигнализации	190
Ge Ha	etAreaName(int area) азвание категории сигнала	191
Ge 3⊦	etBlinkCol(int col1, int col2, int Cflg, int Bflg, int BlkCt) іачение кода цвета	191
Ge Bi Ko	etBlinkColSP(int col1, int col2, int *_pStit, int *_pFlit, int BitNo1, int BitNo2, int tNo3,int BitNo4, int BitNo5, int BitNo6, int BitNo7, int BitNo8, int BlkCt) дировка цвета	191
Ge Пр	etDIBitValue(int *_pFI) реобразование разрядных данных	192

GetMaxInt(int dat1, int dat2) GetMinInt(int dat1, int dat2) Сравнение данных	192
GetMonOff(int type, int _pAMON, int* _pMON) Проверка состояния сигнального монитора	192
InitSprintf() SetSprintfInt(char *fmt, int dat) SetSprintfStr(char *fmt, int dat) SetSprintfDouble(char *fmt, double dat) GetSprintf () Конвертирование строки	192
GetTagAlarmCol(int type, int *_pSTIT, int *_pFLIT, int *_pAGRD, int defCol, int BlkCt) Цветное значение кода тревоги	193
GetTagAlarmColPlus(int type, int *_pSTIT, int *_pFLIT, int *_pAGRD, int _pSCN, int _pAMON,int* _pMON, int Scol, int Mcol, int defcol, int BlkCt) Значение кода цвета тревоги	194
GetTagModel(int tagtype, int modelno) Имя образца модели	194
GetTagTYPLET(int tagtype) Сокращенная строка типа тега	194
InitCntrlDO(char *tag, int *_pMOD,int *_pMODS, int *_pFO, int *_pPB, char **_pSNAM) Регистратор команды кнопки	195
IsTagAlarm(int type, int *_pSTIT, int *_pFLIT, int *_pAGRD) Проверка тревог в теге	195
NaNX(real data, int decml) Отображение реальных данных	195
NewMenuModel(char *menu, int no) Развёртка заданного экрана	196
NewModel(int no) Экран регистрации нового меню	196
RuntimeInstance(char *model) Присоединение модели к объекту	197
SendWinMessage(int cmd, int wParm, int IParm) Посылка сообщения	197
SetAlarmConf(char *tag, int tagtype,int *_parea, int *_plock, int *_patyp, int *_palit, int *_pstit, int *_pflit) Регистрация тега	197
SetCnfColDefV(void *_pType, char *type, void *_pAlit, char *alit,void *_pStit, char *stit, void *_pFlit, char *flit, void *_pArea, char *area, void *_pAtyp, char *atyp, void *_pLock, char *lock, void *_pVar, char *pv, void *_pDcml, char *dcml, void *_pAgrd, char *agrd, void *_pScn, char *scn, void *_pAmon, char *amon, void *_pMon, char *mon) Чтение и набор данных атома тега	198
SetCnfDefV10(void *_pType, char *type, void *_pAlit, char *alit,void *_pStit, char *stit, void *_pFlit, char *flit, void *_pArea, char *area, void *_pAtyp, char *atyp, void *_pLock, char *lock, void *_pVar, char *pv, void *_pDcml, char *dcml, void *_pAgrd, char *agrd) Чтение и набор данных атома тега	199
SetDirectDO(char *tag, int num, int _pChk) Вывод DO тега	200
SetDirectStart() SetDirectEnd() Исполнение DO вывода	200

SetPutDataDbl(char *tagAtom, double data) SetPutDataInt(char *tagAtom, int data) Запись значения установки атома тега	200
SetRTVarInt(char *name, int val) SetRTVarReal(char *name, real val) SetRTVarStr(char *name, char *val) Установка значения заданной переменной	201
SetTGATM(char *tag, char *atom) Строка подключения тега и атома	201
SetTitleText(char *str) Отображение символа заголовка	201
SetVarChk(char *TagAtom, int TagType, int _pBtn, void *_pDlgBody) Диалог установки данных	201
SetVarAtom(char *TagAtom, double Max, double Min, int type, int size, int dtype, int dcml, int _pBtn, void *_pDlgBtn) SetVarAtomSP(char *TagAtom, double Max, double Min, int type, int size, int dtype, int dcml, int _pBtn, void *_pDlgBtn, char *atmName) Отображение данных диалога установки	202
SetVarInt(char *TagAtom, int Max, int Min, int type, int size, int _pBtn, void *_pDlgBtn) SetVarIntSP(char *TagAtom, int Max, int Min, int type, int size, int _pBtn, void *_pDlgBtn, char *atmName) Диалог установки целых чисел	203
SetVarReal(char *TagAtom, real Max, real Min, int type, int size, int dcml, int _pBtn, void *_pDlgBtn, char *atmName) SetVarRealSP(char *TagAtom, real Max, real Min, int type, int size, int dcml, int _pBtn, void *_pDlgBtn, char *atmName) Диалог установки реальных данных	204
SetVarStr(char *TagAtom, int Max, int Min, int type, int _pBtn, void *_pDlgBtn) SetVarStrSP(char *TagAtom, int Max, int Min, int type, int _pBtn, void *_pDlgBtn, char *atmName) Диалог установки строк	204
StartPutData( ) EndPutData( ) Операция записи данных	205
TankLevelBar(real level, real lmin, real lmax) Заполнение модели ёмкости	205
ValueString (real *_pData, int Type, int decml) Создание строки реальных данных	206
MakeAlStsBlock(void *block, void *trigger, char *tagno) Интеграция доступа тега	206
DispAlBlinkStdCol(void *block, int obj, int blink) Отображение изменений аналогового тега	206
MakeDIStsBlock(void *block, void *trigger, char *tagno int bit1, int bit2, int bit3, int bit4, int bit5, int bit6, int bit7, int bit8) Интеграция доступа тега	207
DispDIBlinkStdCol(void *block, int obj, int blink) DispDIBlinkUsrCol1(void *block, int offcol, int obj, int blink) DispDIBlinkUsrCol2(void *block, int oncol, int offcol, int obj, int blink) Отображение изменения аналогового тега	208
MakeDIStsNBlockStart(void *block, void *trigger, int num) MakeDIStsNBlockEnd(void *block, void *trigger) AddDIStsNBlock(void *block, void *trigger, char *tag, int bit1, int bit2, int bit3, int bit4, int bit5, int bit6, int bit7, int bit8) DispDINBlinkStdCol(void *body, int obj, int blink) DispDINBlinkUsrCol1(void *body, int offcol, int obj, int blink)	208

DispDINBlinkUsrCol2(void \*body, int oncol, int offcol, int obj, int blink) Расширенная версия отображения

### Папка моделей

Этот раздел описывает папки, где хранятся модели. Конфигурация папок следующая:

\OIS\SLGMS\MODELS\LOCAL\MODEL \OIS\SLGMS\MODELS\LOCAL\SUBMODEL \OIS\SLGMS\MODELS\USER\FLAT \OIS\SLGMS\MODELS\USER\MODEL \OIS\SLGMS\MODELS\USER\SUBMODEL \OIS\SLGMS\MODELS\STANDARD\FLAT \OIS\SLGMS\MODELS\STANDARD\MODEL \OIS\SLGMS\MODELS\STANDARD\SUBMODEL

\* \OIS\SLGMS\MODELS\LOCAL

Папка, где существуют модели/подмодели, созданные/сохраненные с графическим редактором. МОДЕЛЬ указывает файл модели, и ПОДМОДЕЛЬ указывает файл подмодели. Папки МОДЕЛЬ и ПОДМОДЕЛЬ имеют резервную папку для резервной копии максимума трех старых поколений (порождений).

### \* \OIS\SLGMS\MODELS\USER

Папка, где существуют модели/подмодели, установленные графическим редактором. FLAT (КВАРТИРА) указывает оптимизированный файл модели, МОДЕЛЬ указывает файл модели, и ПОДМОДЕЛЬ указывает файл подмодели. Инсталляция заменяет все OIS, выполняющиеся на той же самой локальной сети с тем же самым файлом. В то же самое время, файлы в \OIS\SLGMS\MODELS\LOCAL удаляются.

Когда OIS инициализированы, выполняется зеркализация от другого OIS.s. Папки МОДЕЛЬ и ПОДМОДЕЛЬ имеют Резервную папку для резервной копии максимума трех старых поколений (порождений).

\* \OIS\SLGMS\MODELS\STANDARD

Папка для стандартных экранных файлов модели/подмодели.

В графике OIS и стандартных экранах, поиск модели и файлов подмодели осуществляется в изложенном выше порядке. Поэтому, модель или файлы подмодели, изменяемые графическим редактором в \OIS\SLGMS\MODELS\LOCAL\ папке находятся и читаются перед другими папками.

Изменение файлов моделей, используемых в стандартном экране и размещение их в \OIS\SLGMS\MODELS\USER папке приводит к тому же самому эффекту как настройка стандартного

экрана.

### Описание переменных

Этот раздел описывает Глобальные переменные.

### INIT

- Тип переменной INIT целое число

• Описание переменной

Переменная инициализации модели. Переменную используемую для инициализации обрабатывают при отображении модели и выходе из процесса при закрытии модели. Значение этой переменной изменяется, как показано ниже.

INIT	5	Инициализация 1	
	3	Инициализация 2	
	1	Инициализация 3	¥.,
	0	Начало операции	
	10	Выход	¥

### BGCOLOR

• Тип переменной

\_ BGCOLOR (ФОНОВЫЙ ЦВЕТ) Целое число

• Описание Переменной

Указывает цвет фона модели. Устанавливается при каждой загрузке модели.

### OPEMODE

• Тип переменной ОРЕМОDE (режим доступа) Целое число

• Описание переменной

Указывает текущий режим доступа в OIS (0:Locked, 1:Operator, 2:Administrator, 3:Engineer).

### \_ZOOM

• Тип переменной

ZOOM (ИЗМЕНИТЬ МАСШТАБ ИЗОБРАЖЕНИЯ) Целого числа

• Описание переменной

Устанавливает изменение коэффициента масштабирования изображения модели.

0	$1 \times$
1	$2 \times$
2	$4 \times$
3	$8 \times$
4	16×

### UPDATE

• Тип переменной UPDATE (ОБНОВЛЕНИЯ) Целое число

• Описание переменной

Счетчик увеличивается каждый раз, когда данные модифицированы.

## BLINKCT

• Тип переменной

\_\_ BLINKCT (МИГАНИЕ) Целое число

• Описание переменной

Операция мигающего индикатора. 0 указывает- мигание выключено, и 1 указывает - выключен цикл мигания.

### Описание функций

### AddTagDot (char\*tagno, int pos) DelTagDot (char\*tagno)

• Описание параметра

tagnoстрока номера тэгаposстроковая позиция (от конца строки)

• Описание функциональных операций AddTagDot добавляется "." к позиции, которая указывает на роз после строки номера тэга.

указанной в tagno.

DelTagDot удаляется "."в конце строки номера тэга, обозначенной tagno.

• возвращаемое значение Строка "." добавляется к AddTagDot, и строка ".", удаляется из DelTagDot.

### BitAnd (int bit1, int bit2) BitOr (int bit1, int bit2) BitXor (int bit1, int bit2)

• Описание Параметра bit1 Целочисленные Данные 1 bit2 Целочисленные Данные 2

• Описание функциональных операций

Возвращает результат из логической операции AND, OR, XOR (И, ИЛИ, неэквивалентность) двух целых чисел.

• возвращаемое значение

Результат логической операции (И, ИЛИ, или неэквивалентности).

### BreakString (char \*str, int chars, int lines)

• Описание Параметра str (строка) отображают chars (символы) количество lines (линия) количество

отображают строку количество символов в одной строке количество отображенных строк

 Описание функциональных операций Делит строку на несколько линий и отображает их. Символы сверх указанного количества опущены с" ~".

• возвращаемое значение Нет.

## ButtonChange (int color1, int color2)

• Описание Параметра color1 Код цвета 1 color2 Код цвета 2

• Описание функциональных операций

Изменяет {заменяет} цвет внешней рамки {фрейма} кнопки команды на указанный цвет.

Код цвета 1: цвет верхней рамки {фрейма}, код цвета 2: цвет более низкой рамки {фрейма} Если код цвета 1 = код цвета 2 =0, цвет отображения экранного устройства инвертирован.

 возвращаемое значение Нет

ButtonHilite (int color1, int color2) ButtonHiliteChk (int color1, int color2, int exechk) ButtonChk3 (int color1, int color2, int backcol, intdispchk, int exechk)

• Описание Пара	метра
color1	Код цвета 1
color2	Код цвета 2
backcol	Цвет фона
exechk	Условие {Состояние} выполнения
dispchk	Условие {Состояние} отображения

• Описание функциональных операций

Отображает вкл\выкл состояние кнопки команды с изменением {заменой} внешнего цвета рамки {фрейма} кнопки команды операции.

Код цвета 1: цвет верхней рамки {фрейма} в ОN состоянии, код цвета 2: цвет более низкой рамки {фрейма} в ОN состоянии.

Если код цвета 1 = код цвета 2 =0, используется цвет отображения экранного устройства. Не выполняется, если условие {состояние} выполнения = 0.

Если условие {состояние} отображения = 0, используется цвет фона (кнопка удалена). Объекты - группы больше чем из трех модулей (1: фон, 2: верхняя рамка {фрейм}, 3: нижняя рамка {фрейм}).

Цветной код ButtonChk3 должен быть определен.

• возвращаемое значение

Нет

• Замечание

Gismo специальная функция.

## ButtonMessage (char\*title1) ButtonMessage2 (char\*title1, символ \*title2)

• Описание Параметра

title1	-	строка описания 1
title2		строка описания 2

• Описание функциональных операций

Выводит строку, описывающую функцию кнопки в строке состояния, когда кнопка нажата. Этот функция обратного вызова - Gismo специальная функция. В отличие от другой Gismo специальная функции, эта работает при нажатии кнопки мыши. ButtonMessage () отображает символ, обозначенный title1, a ButtonMessage2 () отображает символы, обозначенные title1 и title2 в строке состояния.

• возвращаемое значение Нет.

• Замечание Gismo специальная функция.
## CheckVarInit (void \*\_pData) (пусто \*\_pData)

• Описание Параметра

\_pData

Переменная

• Описание функциональных операций Проверка, если инициализация и чтение переменной нормальны (проверка на корректность).

• Возвращаемое значение

Если инициализация и чтение переменной нормальны (корректны), возвращается 1.

# ChkDirect (int \*\_pChk, символ \*tag1, int num1..., символ \*tag4, int num4)

• Описание Параметра

_pChk	Переменная
tag1 ~ tag4	Номер тэга
num1 ~ num4	DO номер

• Описание функциональных операций

Проверка статуса четырёх основных выходов. Если DO выход разрешён для всех из них, то перемен-ная \_pChk устанавливается в1, иначе переменная \_pChk устанавливается в 0. Номер тэга должен иметь "."в конце.

Адресаты для DO проверяют состояние максимум четырех типов, DO выходов, с комбинациями типа номера тэга tag1 и DO номер num1, и так далее.

Когда вы хотите проверить состояние двух типов DO, определяют tag1, tag2, num1, и num2 и устанавливают весе другие параметры в 0.

• Возвращаемое значение Нет

### ChgIntVar (void \*\_pSetPoint, int Set Data)

• Описание Параметра pSetPoint Переменная

SetData

Устанавливаемое значение

• Описание функциональных операций

Данные для набора переменной целочисленного типа.

• Возвращаемое значение Нет

### ChkMoniGrp (int area)

• Описание параметра Категория сигнала

#### • Описание функциональных операций

Проверка категории отображения сигнальной группы, если определено снижение в пределах области

• устанавливаемое значение

1: в пределах сигнальной группы монитора, 0: вне сигнальной группы монитора

#### ChkOPWMode (int opemode)

• Описание Параметра Режим операции (доступ)

• Описание функциональных операций

Проверка режима операции (доступа) (0: Locked, 1: Operator, 2: Administrator, 3:Engineer) (0: Блокированный, 1: Оператор, 2: Администратор, 3:Инженер) указанного в оретоde - режиме, описывающего как в настоящее время отображенный экран.

• Возвращаемое значение

1: Допускаемый, 0: Заблокированный

• Замечание

Как правило, глобальная переменная \_OPEMODE, которая указывает режим операции (доступа) OIS.s, определена для opemode.

## DefTitleModel (int no, void \*\_pno, void \*\_ptitle)

•	Описание	Параметра
		**

no	Номер
_pno	Номер Меню
_ptitle	Название {Имя} Заголовка

• Описание функциональных операций

Устанавливает номер меню для переменной, обозначенной \_pno и строку заголовка для \_ptitle, основанную на обозначенном номере, и текущей странице меню. Когда ничего не зарегистрировано в меню, ничего не устанавливается

Чтобы определить, зарегистрировано ли что - нибудь или нет, функция использует

DefTitleModelChk (). Эта функция обратного вызова должна быть выполнена при инициализации с INIT=5 или INIT=1.

• Возвращаемое значение Нет

#### DefTitleModelChk (int no, void \*\_pchk)

• Описание Параметра

no номер \_pchk проверяемое значение

• Описание функциональных операций

Установки, зарегистрирована переменная, обозначенная \_pchk в меню или нет, базировались на обоз-начении номера не текущей страницы меню. Функция устанавливает 1, если есть регистрация и 0 если нет.

Эта функция обратного вызова должна быть выполнена при инициализации с \_INIT=5 или \_INIT=1.

• Возвращаемое значение Нет

#### DefVar (void \*\_pvar, char \*tagatom, 1, int type, int array, int col, 1) DefV2 (void \*\_pvar, char \*tagatom)

• Описание Параметра

_pvar	Указатель на переменную, куда введен результат выборки тэга.
tagatom	Название {Имя} атома тэга
type	Тип переменной. 1: реальный, 2: строка, 4: целое число
array	Номер массивов. Если-1 определен, это откорректировано к типу атома.
Col	Тип набора. 1: Собранный циклически, 3: Собранный только в первый раз.

Чтения в данных атома тэга, указанного в tagatom и \_pvar. Тип данных, которые читаются определен типом переменой. В типе массива определен номер массивов для обращения к данным. Переменная \_pvar переопределена в типе и массиве, указанном в type и array. Col определяет тип коллекции.

Эта функция обратного вызова должна быть выполнена при инициализации с \_INIT=5 если набор тип 3 (собран только в первый раз) или \_INIT=1, если набор тип 1 (собрано периодически). DefV2 () эквивалентен DefVar (\_pvar, tagname, 1,1,1,1)

• Возвращаемое значение Нет

DefVarInt (void \*\_pvar, int val) DefVarReal (void \*\_pvar, real val) DefVarStr (void \*\_pvar, char \*val)

• Описание Параметра

\_pvar указатель на переменную val значение переменной

• Описание функциональных операций

Инициализирует значение переменной, обозначенной как \_pvar. DefVarIni () используется, если уста-новлено целое число, DefVarReal () используется, если установлено реальное значение , и DefVarStr () используется, если установлена строка.

Эта функция обратного вызова должна быть выполнена при инициализации с \_INIT=5 или \_INIT=1.

 Возвращаемое значение Нет.
 DialogInstance1 (char \*NewModel, int pX, int pY, int pW, int pH)

• Описание Параметра

NewModel	Название {Имя} модели
pX	Х координата верхнего левого угла окна диалога
pY	У координата верхнего левого угла окна диалога
pW	Ширина окна диалога
pН	Высота окна диалога

• Описание функциональных операций

Отображает указанную модель в окне диалога.

Позиция дисплея и размер окна диалога зависят от параметров.

Если для Х и координат Ү дается -1, диалог отображен от позиции, где Вы щелкнули мышью.

• Возвращаемое значение Всегда 1.

• Замечание Gismo специальная функция.

DialogInstance2 (char \*NewModel, int pX, int pY, int pW, int pH, int pUsrBtn, int pDelflg, void \*\_pDlgBody)

• Описание Параметра	
NewModel	Название {Имя} модели
px	Х координата верхнего левого угла окна диалога
pY	Ү координата верхнего левого угла окна диалога
pW	Ширина окна диалога
pН	Высота окна диалога
pUsrBtn	Спецификация инструментальной панели для окна диалога
pDelflg	Условие очистки окна диалога
_pDlgBody	Состояние отображения окна диалога

Отображает указанную модель в окне диалога с инструментальной панелью.

инструментальная панель, отображенная в окне диалога зависит от следующей спецификации инструментальной панели:

0: CNF/LO/EXE/Warning

1: BW/FW/GB/Warning

- 2: CNF/LO/EXE/BW/FW/GB/Warning 3: Warning (Предупреждение)
- 4: CNF/LO/EXE/Warning/Deletion
- 5: BW/FW/GB/Warning/Deletion
- 6: CNF/LO/EXE/BW/FW/GB/Warning/Deletion
- 7: Warning/Deletion (Предупреждение/Удаление)

Если 20 - 27 определен, используется инструментальная панель в меньшем размере. Условия удаления окна диалога следующие:

0: Удаляется при первичной развертке экрана.

1: Не удаляется даже при первичной развертке экрана.

2: Диалоги, начатые в 2 не отображаются одновременно. Более поздний имеет более высокий приоритет.

Условия отображения окна диалога следующие:

0: Окно диалога не отображено.

Не 0: окно диалога отображается.

Позиция отображения и размер окна диалога зависят от параметров.

Если дается для X и координат Y -1, диалог отображен от позиции, где Вы щелкнули мышью.

• Возвращаемое значение

Всегда 1.

• Замечание

Gismo специальная функция.

# DialogInstance3 (char \*opwType, int no, int pX, int pY, int pW, int pH, int pUsrBtn, int pDelflg, void \*\_pDlgBody)

Int posrBth, int poeifig, void "\_poigBody)

• Описание Параметра	
opwType	Клавиша экрана меню или клавиша элемента {точки}
no	Экранный номер
pX	Х координата верхнего левого угла окна диалога
рY	У координата верхнего левого угла окна диалога
pW	Ширина окна диалога
pH	Высота окна диалога
pUsrBtn	Спецификация инструментальной панели для окна диалога
pDelflg	Условие очистки окна диалога
pDlgBody	Состояние отображения окна диалога
0 1	v

• Описание функциональных операций

Отображает указанную модель в окне диалога с инструментальной панелью.

ОрукТуре и не используется для экрана, отображенного в диалоговом окне. См. описание NewMenuModel для экранной клавиши типа стандартных экранов. Клавишу экрана меню и клавиша элемента {точки} показывают ниже.

- 0: CNF/LO/EXE/Warning
- 1: BW/FW/GB/Warning
- 2: CNF/LO/EXE/BW/FW/GB/Warning
- 4: CNF/LO/EXE/Warning/Deletion
- 5: BW/FW/GB/Warning/Deletion
- 6: CNF/LO/EXE/BW/FW/GB/Warning/Deletion

7: Warning/Deletion (Предупреждение/Удаление)

3: Warning (Предупреждение)

Если 20 - 27 определен, используется инструментальная панель в меньшем размере.

Условия удаления окна диалога следующие:

0: Удаляется при первичной развертке экрана.

1: Не удаляется даже при первичной развертке экрана.

2: Диалоги, начатые в 2 не отображаются одновременно. Более поздний имеет более высокий приоритет.

Условия отображения окна диалога следующие:

0: Окно диалога не отображено.

Не 0: окно диалога отображается.

Позиция отображения и размер окна диалога зависят от параметров.

Если дается для X и координат Y -1, диалог отображен от позиции, где Вы щелкнули мышью.

• Возвращаемое значение Всегла 1.

• Замечание

Gismo специальная функция.

#### DispAlarmCol (int тип, int \*\_pSTIT, int \*\_pFLIT, int \*\_pAGRD, int TLPfg, int BlkCt)

• Описание Параметра

type	Тип тэга
pSTIT	STIT переменная
	FLIT переменная
pAGRD	AGRD переменная
TLPfg	Отображение спецификации
BlkCt	Спецификация мигания

• Описание функциональных операций

Отображает сигнальную информацию с самым высоким приоритетом, основанным на типе тэга, сигнальном состоянии возникновения (STIT), неподтвержденное состояние ((FLIT) мигает), и степени (AGRD).

Спецификации дисплея следующие:.

0: Отображение аббревиатуры сигнала.

32: Название {имя } дисплея тревог (сигнализации)

(Для DI1 - DI8, сигнальная информация не отображена, а цвет отображения текста изменен.) 0 бит = 1

(1, 3, 5, 7): Изменяется цвет текстового дисплея.

1 бит = 1

(2, 3, 6, 7): Изменяется цвет отображения строки.

2 бит = 1

(4, 5, 6, 7): Изменяется заполняющий цвет дисплея.

• Возвращаемое значение

Возвращает тип сигнала.

0: Никакой тревоги

1: Другая тревога

2: PV- зависимая тревога

3: MV- зависимая тревога

4 ~ 11: тревога от D11 до D18

• Замечание

Чтобы получить сигнальный номер цвета, используйте GetTagAlarmCol ().

#### DispAlarmColPlus (int type, int \*\_pSTIT, int \* pFLIT, int \*\_pAGRD, int \_pSCN, int \_pAMON, int \*\_pMON, int Scol, int M col, int TLPfg, int BlkCt)

#### • Описание Параметра

type	Тип тэга
pSTIT	STIT п еременная
pFLIT	FLIT (МЕЛЬКАЕТ) п еременная
pAGRD	AGRD переменная
pSCN	SCN переменная
pAMON	АМОН переменная
pMON	МОН переменная
SCol	Просмотр OFF Цвет
MCol	просмотр тревог OFF Цвета
TLPfg	Отображает требование
BlkCt	Мигающее требование

• Описание функциональных операций

Отображает сигнальную информацию в цвете самого высокого приоритета, основанного на типе тэга, возникновении сигнала состояния (STIT), не подтвержденном состоянии (FLIT), степени (AGRD), просмотре, установленном (SCN), сигнализирует в общедоступном мониторе спецификацию (AMOH), и контролирует индивидуальную тревогу спецификации (MOH), в порядке, показанном ниже:

Неподтвержденная тревога> Просмотр OFF> Подтвержденная тревога>просмотр OFF сигнального монитора > Норма

Цвета, используемые как цвет дисплея следующие.

Просмотр OFF.

Просмотр OFF:	Просмотр выключения используемого цвета.
Просмотр сигнального монитора OFF:	Просмотр выключения сигнального монитора.
Другой	Цвет аварийной ситуации.

Спецификации дисплея следующие.

0 : Сигнальный дисплей аббревиатуры

32: Сигнальный дисплей названия {имени}

(Для DI1 - DI8, сигнальная информация не отображена, а цвет отображения текста изменен.) 0 бит = 1

(1, 3, 5, 7): Изменяется цвет текстового дисплея.

1 бит = 1

(2, 3, 6, 7): Изменяется цвет отображения строки.

2 бит = 1

(4, 5, 6, 7): Изменяется заполняющий цвет дисплея.

• Возвращаемое значение

Возвращает тип сигнала.

0: Никакой тревоги

1: Другая тревога

2: PV- зависимая тревога

3: MV- зависимая тревога

4 ~ 11: тревога от D11 до D18

• Замечание Чтобы получить сигнальный номер цвета, используйте GetTagAlarmCol ().

# DispAllAlarm (int type, int \*\_pSTIT, int \*\_pFLIT, int \*\_pAGRD, int \*\_pAMON, int \*\_pMON, char \*\*\_pSNAM, int BlkCt)

• Описание Параметра

type	Тип тэга
_pSTIT	STIT п еременная
_pFLIT	FLIT (МЕЛЬКАЕТ) п еременная
_pAGRD	AGRD переменная
_pSCN	SCN переменная
_pAMON	АМОН переменная
_pMON	МОН переменная
BlkCt	Спецификация мигания

• Описание функциональных операций

Отображает сигнальные подробности экрана пункта {точки}. Эффектор объекта, который содержит множественный номер - сгруппированные символы с рамкой {фреймом}. Функция также отображает информацию номера групп в порядке тревоги.

• Возвращаемое значение Нет

#### DispAreaName (int area)

• Описание Параметра area Сигнал категории

• Описание функциональных операций

Заменяет строковый объект названием категории сигнала, указанной областью. Название {имя} категории сигнала отображено в голубом.

• Возвращаемое значение

Нет

• Замечание

Чтобы получить строку для названия {имени} категории сигнала, используйте GetAreaName ().

# DispBlinkCol (int col1, int col2, int Cflg, int Bflg, int TLPfg, int BlkCt)

• Описание Параметра

col1	Цвет 1
col2	Цвет 2
Cflg	Выбор цвета
Bflg	Выбор мигания
TLPfg	Отображение спецификации
BlkCt	спецификация мигания

• Описание функциональных операций

Отображает цвет дисплея 1 если выбор цвета = 1, и цвет дисплея 2 если выбор цвета = 0. Если выбор мигания = 1 и спецификация мигания = 0, отображение в черном (мигающий дисплей). Символ (0 битов), линия (1 бит), или заполнение (2 бита) определяется спецификацией дисплея.

• Возвращаемое значение Нет.

• Замечание

Чтобы получить только цветной код, используйте GetBlinkCol ().

# DispBlinkCoISP (int col1, int col2, int \*Stit, int \*Flit, int BitNo1, int BitNo2, int BitNo3,

int BitNo4, int BitNo5, int BitNo6, int BitNo7, int BitNo8, int TLPfg, int BlkCt)

• Описание Параметра		
Цвет 1		
Цвет 2		
Значение атома STIT (массив)		
Значение атома Flit (массив)		
Разрядный номер (от 0 до 7)		
Отображает спецификацию (требование)		
спецификация мигания		

• Описание функциональных операций

BitNo1 - 8 определяет разрядный номер, который используется как условия. Для спецификации номера бита, используйте их начиная с наименьшего номера, BitNo1.

Выбор цвета OR (ИЛИ) между STIT и FLIT (МЕЛЬКАТЬ) из указанного BitNo. Цвет отображения дисплея 1, если выбор цвета =1, и цвет отображения дисплея 2, если выбор цвета =0.

Выбор Мигания – OR (ИЛИ) между FLIT.s указанного BitNo. Если выбрано мигание = 1 и специфи-кация мигания = 0, отображает в черном (мигающий дисплей).

Символ (0 битов), строка (линия) (1 бит), или Заполнение (2 бита) определяется спецификацией Дисплея.

• Возвращаемое значение Нет.

• Замечание

Чтобы получить только цветной код, используйте GetBlinkColSP ().

# DispCntrlMode (int \*\_pMod, int \*\_pMods, int is\_Mods, int BkCol, int BlkCt)

• Описание Параметра

_pMod	MOD переменная
_pMods	MODS переменная
is_Mods,	наличие MODS
BkCol	цвет фона
BlkCt	спецификация мигания

• Описание функциональных операций

Подсвечивает символы режима управления (С, А, и М).

С: желтый, А: зеленый, М.: красный, отдаленный М.: красный мигающий.

Если, ли MODS существуют или not=1 а MODS=0, это удаляет их в цвете фона.

• Возвращаемое значение Нет

# DispHbar (real dt1, real dt2, real R Ldt, real RHdt, real xmin, real xmax)

#### • Описание Параметра

отображаемые данные 1
отображаемые данные 2
нижняя граница диапазона
верхняя граница диапазона
минимальная Х координата
максимальная Х координата

• Описание функциональных операций

Отображает данные дисплея в горизонтальной гистограмме от нижней границы диапазона до верхней границы диапазона.

Когда дисплей Data1 и низкое значение по диапазону, гистограмма правее. Когда дисплей Data2 и высокое значение по диапазону, гистограмма левее.

Гистограмма создаётся определяя данные, диапазон и граничные значения.

• Возвращаемое значение Нет

#### DispHslid (real Hdt, real RLdt, real RHdt, real xmin, real xmax)

#### • Описание Параметра

Отображаемые данные
нижний диапазон
верхний диапазон
минимальная Х координата
максимальная Х координата

• Описание функциональных операций

Отображает данные дисплея в слайдере с меткой скольжения от нижнего диапазона к верхнему.

• Возвращаемое значение Нет

### DispMofFrame (int type, int \*\_pAMON, int \*\_pMON, int BkCol)

• Описание Параметра

type	Тип тэга
_pAMON	Переменная AMON
pMON	Переменная MON
BkCol	Цвет фона

• Описание функциональных операций

Если сигнал пакетного сигнального монитора (AMOH) или частного сигнального монитора (MON) OFF, строки отображаются в синем индикаторе. Если оба сигнала ON, строки удаляются в цвете фона.

• Возвращаемое значение Нет

• Замечание Проверяйте переключение контроля вкл. или выкл., использовуя GetMonOff ().

### DispRealData (real \*\_pData, int decml)

• Описание Параметра	
_pData	Реальная переменная
decml	Десятичный разряд

• Описание функциональных операций

Отображает реальные данные с указанием десятичных разрядов.

• Возвращаемое значение Нет

• Замечание

Для получения переменной строки используйте ValueString ().

# DispVar (real dt1, real dt2, real R Ldt, real RHdt, real ymin, real ymax) DispVbarPFI (real dt1, real dt2, real R Ldt, real RHdt, real ymin, real ymax)

#### • Описание Параметра

	1	1
dt1		отображаемые данные 1
dt2		отображаемые данные 2
RLdt		нижняя граница диапазона
RHdt		верхняя граница диапазона
xmin		минимальная Х координата
xmax		максимальная Х координата
		-

• Описание функциональных операций

Отображает данные дисплея в вертикальной гистограмме от более низкого диапазона до более высокого диапазона.

Когда отображаемые данные 1 и нижняя граница диапазона та же самая, восходящая гистограмма кончается.

Когда отображаемые данные 2 и верхняя граница диапазона та же самая, нисходящая гистограмма кончается.

Определяя данные, диапазон разбиения, или PH-PL, можно создать гистограмму.

DispVbarPFI отображает вертикальную гистограмму измерителя коэффициента мощности.

• Возвращаемое значение Нет.

#### DispVslid (real Vdt, real RLdt, real RHdt, real ymin, real ymax) DispVslidPFI (real Vdt, real RLdt, real RHdt, real ymin, real ymax)

•	Описание	Параметра
---	----------	-----------

Vdt	отображаемые данные
RLdt	нижняя граница диапазона
RHdt	верхняя граница диапазона
xmin	минимальная Х координата
xmax	максимальная Х координата

• Описание функциональных операций

Отображает данные дисплея в вертикальном слайдере с меткой скольжения от нижнего диапазона к верхнему.

DispVslidPFI отображает измеритель коэффициента мощности в вертикальном слайдере с меткой скольжения.

• Возвращаемое значение Нет. DoGraphPoint (char \*tagno, int num)

• Описание Параметра		
tagno	Номер тэга	
num	Номер экрана	

• Описание функциональных операций

Разворачивает графический экран, обозначенный пит.

Если num - 0, или указанный графический экран уже существует, этот экран разворачивается к экрану группы, если tagno зарегистрирован в группе. Если никакая группа не зарегистрирована, этот экран разворачивает к существующему экрану.

Как правило, num - значение Tag.s Атома SCNO (экранный номер).

• Возвращаемое значение Нет.

• Замечание Gismo специальная функция.

#### GetAlmCol (int almgrd, int Blinkct)

• Описание Параметра	
almgrd	Значение уровня тревоги (сигнализации)
Blinkct	мигающий индикатор

• Описание функциональных операций

Regurns сигнальный цвет, который соответствует almgrd. Для неподтвержденного состояния, вкл\выкл мигающий индикатор.

• Возвращаемое значение Значение кода цвета.

• Замечание Используется с GetAlmGrd ().

### GetAlmGrd (int \*\_pSTIT, int \*\_pFLIT, int \*\_pAGRD, int bitno)

• Описание Параметра	
_pSTIT	STIT переменная
pFLIT	FLIT (МЕЛЬКАТЬ) переменная
pAGRD	AGRD переменная
bitno	Разрядный номер (от 0 до 31)

• Описание функциональных операций

Если тревога происходит в позиции, указанной bitno в сигнальном состоянии возникновения (STIT) и

неподтвержденном состоянии (FLIT), функциональных возвращает уровень (степень) сигнализации.

Возвращаемое значение
 Значение уровня сигнализации.

#### GetAreaName (int area)

• Описание Параметра Категория сигнала

• Описание функциональных операций Получает строку категории сигнала, указанную областью.

• Возвращаемое значение

Название {имя} категории сигнала. Если значение категории сигнала является другим, чем от 0 до 63, строка становится????.

#### GetBlinkCol (int col1, int col2, int Cflg, int Bflg, int BlkCt)

• Описание Параметра

coll	цвет 1
col2	цвет 2
Cflg	выбор цвета
Bflg	выбор мигания
BlkCt	спецификация мигания

• Описание функциональных операций

Отображает цвет дисплея 1 если выбор цвета = 1, и цвет дисплея 2 для выбор цвета = 0. Если выбор мигания = 1 и спецификация мигания = 0, отображение в черном (мигающий дисплей).

Символ (0 бит), строка (1 бит), или заполнение (2 бит) определяется спецификацией дисплея.

• Возвращаемое значение Значение кода цвета.

GetBlinkColSP (int col1, int col2, int \*\_pStit, int \*\_pFlit, int B itNo1, int B itNo2, int B itNo3, int B itNo4, int B itNo5, int B itNo6, int B itNo7, int B itNo8, int B lkCt)

• Описание Параметра

col1	цвет 1
col1	цвет 2
Stit	значение Атома STIT (массив)
Flit	значение Атома FLIT (массив)
BitNo1 ~ BitNo8	разрядный Номер (от 0 до 7)
BlkCt	спецификация мигания

• Описание функциональных операций

BitNo1 до 8 определяют разрядный номер, который используется как условие. Для спецификации используйте номер бита начинающийся с наименьшего номера, BitNo1.

Выбор – OR (ИЛИ) между STIT и FLIT из указанного BitNo. Отображается цвет дисплея 1, если выбор цвета =1, и цвет дисплея 2, если выбор цвета =0.

Выбор мигания – OR (ИЛИ) между FLIT.s указанного BitNo. Если выбор мигания = 1 и спецификация мигания = 0, это отображается в черном (мигающий дисплей).

• Возвращаемое значение Код цвета.

### GetDIBitValue (int \*\_pFI)

• Описание Параметра

pFI

Переменная типа массива разрядов (двоичный код)

• Описание функциональных операций

Конвертирует {преобразут} разрядные данные массива, указанные в \_pFI к числовому значению. Например, если \_pFI [0] = 1, \_pFI [1] = 0, и \_pFI [2] = 1, это преобразуется в 5.

• Возвращаемое значение

Возвращается числовое значение, к которому преобразованы {конвертированы} разрядные данные.

#### GetMaxInt (int dat1, int dat2) GetMinInt (int dat1, int dat2)

• Описание Параметра	
dat1	Целочисленные данные 1
dat2	Целочисленные данные 2

• Описание функциональных операций

GetMaxInt сравнивает dat1 и dat2 и возвращает большее значение. GetMinInt сравнивает dat1 и dat2 и возвращает меньшее значение.

• Возвращаемое значение

GetMaxInt сравнивает dat1 и dat2 и возвращает большее значение. GetMinInt сравнивает dat1 и dat2 и возвращает меньшее значение.

#### GetMonOff (int тип, int \_pAMON, int \*\_pMON)

• Описание Параметра

type	Тип тэга
_pAMON	Переменная AMON
_pMON	Переменная MON

• Описание функциональных операций

Проверяет установленное состояние сигнального монитора, основанное на типе тэга, сигнальной спецификация общедоступного монитора (АМОН), и контроле индивидуальной спецификации тревог (MON).

Установка сигнального монитора становится OFF, если сигнальный монитор совместно использовал спецификацию OFF или не менее одной из сигнальных спецификаций индивидуального монитора OFF.

• Возвращаемое значение

1, если установка сигнального монитора ВЫКЛЮЧЕНА. Иначе 0.

InitSprintf () SetSprintfInt (char \*fmt, int dat) SetSprintfStr (char \*fmt, int dat) SetSprintfDouble (char \*fmt, double dat)

### GetSprintf ()

• Описание Параметра

fmt формата преобразования строки dat данные

• Описание функциональных операций

InitSprintf () инициализирует процесс преобразования строки данных. SetSprintfInt (), SetSprintfStr (), и SetSprintfDouble () конвертируют {преобразовывают} строку dat в формате, указанном в fmt и сохраняет это.

GetSprintf () получает сохраненную строку.

• Возвращаемое значение

Преобразованная {конвертированная} строка полученная, используя GetSprintf ().

• Замечание

Строка, преобразованная {конвертированная} SetSprintfInt (), SetSprintfStr (), и SetSprintfDouble () становится после преобразованная {конвертированная} последней строкой.

 Например:
 \_pVar1
 11 (целое число)

 \_pVar2
 10.5 (реальное)

 \_pStr
 Детальные данные (Строка)

 \_pVar1
 = \*

 InitSprintf ()
 SetSprintfInt (.Integer Значение: %d.,\_pVar1)

 SetSprintfStr (. %s.,\_pStr)
 SetSprintfDouble (. %5.2f.,\_pVar2)

 stext GetSprintf (). %s.
 Результат: Детальные данные для целочисленного значения: 11 - 10.50.

# GetTagAlarmCol (int тип, int \*\_pSTIT, int \*\_pFLIT, int \*\_pAGRD, int defCol, int BlkCt)

 • Описание Параметра

 type
 Т

 \_pSTIT
 S

 \_pFLIT
 F

 \_pAGRD
 A

 delCol
 Ш

 BlkCt
 C

Тип тэга STIT переменная FLIT переменная AGRD переменная Цвет по умолчанию Спецификация мигания

• Описание функциональных операций

Возвращает значение кода цвета тревоги с самым высоким приоритетом, основанным на типе тэга, состоянии возникновения тревоги (STIT), неподтвержденном состоянии (FLIT), и степень (AGRD).

Если \_pSTIT, \_pFLIT, или \_pAGRD не читается или не может читаться должным образом при отображении модели, возвращается цвет, определенный defCol.

• Возвращаемое значение

Цветное значение кода тревоги.

GetTagAlarmColPlus (int type, int \*\_pSTIT, int \*\_pFLIT, int \*\_pAGRD, int \_\_pSCN,

• Описание Параметра	
напечатайте Тип Тэга	
_pSTIT	STIT переменная
_pFLIT	FLIT переменная
_pAGRD	AGRD переменная
_pSCN	SCN переменная
_pAMON	АМОН переменная
_pMON	МОМ переменная
SCol	Просмотр, OFF цвет
MCol	Сигнальный Монитор, OFF цвет
defcol	Цвет по умолчанию
BlkCt	Спецификация мигания

Возвращает значение кода цвета, основанное на типе тэга, возникновении сигнального состояния (STIT), неподтвержденном состоянии (FLIT), степени (AGRD), установке просмотра (SCN), спецификации совместно используемого монитора тревог (AMOH), и спецификации контроля индивидуальных тревог (MON), в порядке, показанном ниже: Неподтвержденная тревога> последовательный просмотр OFF > подтвержденная тревога>последовательный сигнальный монитор OFF > Норма Используемые цвета следующие. Scan Setting OFF: Scan Setting OFF цвет используемый. Alarm Monitor Setting OFF: Alarm Monitor Setting OFF цвет используемый. Other (другой): Цвет аварийной ситуации. Eсли \_pSTIT, \_pFLIT, \_pAGRD, или \_pMON не читается, или не может читаться должным образом, при отображеннии модели, возвращается цвет, определенный defCol.

• Возвращаемое значение Значение кода цвета тревоги.

#### GetTagModel (int tagtype, int modelno)

• Описание Параметра	
tagtype	Тип тэга
modelno	Номер модели

• Описание функциональных операций

Создает название {имя} модели образца дисплея тэга, которое определено основанным на типе тэга и номере модели.

• Возвращаемое значение Название {Имя} модели образца дисплея тэга. При ошибке типа тэга возвращается TagError..

#### GetTagTYPLET (int tagtype)

• Описание Параметра tagtype Тип тэга

• Описание функциональных операций Создает сокращённую строку типа тэга.

• Возвращаемое значение

Возвращает сокращённую строку типа тэга. Возвращает пробел, если тип тэга = 0. При ошибке типа тэга возвращается Err.

# InitCntrIDO (char \*tag, int \*\_pMOD, int \*\_pMODS, int \*\_pFO, int \*\_pPB, char \*\*\_pSNAM)

• Описание Параметра

tag	Номер тэга
_pMOD	Указатель на Tag.s MOD данные
_pMODS	Указатель на Tag.s MODS данные
_pFO	Указатель на Tag.s FO данные
_pPB	Указатель на Tag.s PB данные
_pSNAM	Указатель на Tag.s SNAM данные

• Описание функциональных операций

Регистраторы DO из команды кнопки, отмечают как адресат операции вывода. Эта функция эффективна для тэгов, зарегистрированных в SetAlamConf (). А. "." нужно добавить в конце номер тэга.

Переменные \_pMOD, \_pMODS, \_pFO, \_pPB и \_pSNAM должны иметь доступ тэга, указанный DefVar ().

Эта функция должна быть выполнена при инициализации (\_INIT=5 или \_INIT=1).

• Возвращаемое значение Нет.

#### IsTagAlarm (int type, int \*\_pSTIT, int \*\_pFLIT, int \*\_pAGRD)

• Описание Параметра

type	Тип тэга
_pSTIT	STIT переменная
_pFLIT	FLIT переменная
_pAGRD	AGRD переменная

• Описание функциональных операций

Проверка любой тревоги в тэге, основанной на сигнальном статусе возникновения (STIT), неподтвержденном состоянии (FLIT), и степени (AGRD).

• Возвращаемое значение

0, если никакая тревога не происходит, и 1, если тревога происходит.

#### NaNX (real data, int decml)

• Описание Параметра	
data	Реальная переменная
decml	Десятичный разряд

• Описание функциональных операций Отображает реальные данные с указанным десятичным разрядом.

• Возвращаемое значение Ни один

• Замечание Чтобы получить конверсионную строку, используйте ValueString ().

#### NewMenuModel (char \*menu, int no)

• Описание Параметра menu Клавиша экрана меню, или клавиша экрана точки. No Номер

• Описание функциональных операций

Разворачивает экран, основанный на экране меню или экране точки, обозначенном по (номером)меню и номером страницы.

Если не существует модель соответствующая зарегистрированному меню и номеру меню, экран отображает пробел.

• Возвращаемое значение Нет

• Замечание

Gismo специальная функция.

Экранная клавиша стандартного экрана - показывает последовательно.

Когда тип - менк	), и номер страницы0., определена предыдущая страница.	
MENU	экран меню [NO = страница номер.]	

ALM	системная тревога/отдельная тревога/руководство
	[NO = 1: дисплей состояния системы, 2: дисплей отдельной тревоги, 3:
	руководство]
GRAM	графическое меню [NO = Страница Номер]
GRAP	графический символ [NO = Экранный Номер]
EXEM	программное меню [NO = Страница Номер]
EXEI	экран меню программ [NO = Экранный Номер]
FWDM	дисплей меню с 4 экранами [NO = Страница Номер]
FWDI	4х экранное отображение меню [NO = Страница Номер]
GRPM	экран обзора группы [NO = Страница Номер]
GRPI	кран группы [NO = Группа Номер]
TRDM	меню историческое тенденции [NO = Страница Номер]
HTRD	экран исторический тенденции [NO = Группа Номер]
RTRM	меню тенденции в реальном масштабе времени [NO = Страница Номер]
RTRD	экран тенденции в реальном масштабе времени [NO = Группа Номер]
PARM	дисплей параметра
SNSM	экран датчика корректируют
SCPM	VSCP утилита

#### NewModel (int no)

• Описание Параметра

по Номер

• Описание функциональных операций

Разворачивает экран регистрации нового меню и номера меню, основанного на номере, обозначенном

по и текущей странице меню. Если не существует никакая модель, соответствующая зарегистрирован-ному меню и номеру меню, отображается пустой экран.

• Возвращаемое значение Нет.

• Замечание

Gismo специальная функция.

#### RuntimeInstance (char \*model)

• Описание Параметра model

Модель

• Описание функциональных операций

Присоединяет модель, указанную в model к объекту на который назначен запрос функции обратного вызова

(в качестве примера). При присоединении модели, используемый размер изменяется на размер объекта, к которому модель присоединяется. Приложенная (присоединённая) модель становится подмоделью модели, которая содержит объект. Если не существует никакой модели, ничто не отображается на экране. Эта функции обратного вызова должна быть выполнена при инициализации с INIT=5 или INIT=1.

• Возвращаемое значение Нет.

#### SendWinMessage (int cmd, int wParm, int IParm)

• Описание Параметра	
cmd	Команда окна сообщения
wParm	Параметр 1 окна сообщения
IParm	Параметр 2 окна сообщения

• Описание функциональных операций

Посылает сообщение, обозначенное cmd/wParm/lParm в окно, отображающее модель. Это обычно используется, чтобы создать кнопку Exit для отображения в диалоге модели. Чтобы создать кнопку Exit, установите в cmd 273, в wParm 2, в lParm 0.

• Возвращаемое значение Нет.

• Замечание

Gismo специальная функция.

# SetAlarmConf (char \*tag, int tagtype, int \*\_parea, int \*\_plock, int \*\_patyp, int \*\_palit, int \*\_pstit, int \*\_pflit)

#### • Описание Параметра

tag	Номер тэга
_tagtype	Тип тэга
_parea	Указатель на данные категории сигнала Tag.s
_plock	Указатель на данные операции отключения Tag.s
_patyp	Указатель на данные метода тревоги Tag.s
_palit	Указатель на данные аварийной ситуации Tag.s
_pstit	Указатель на данные отображения состояния тревоги Tag.s
_pflit	Указатель на неподтвержденные данные состояния тревоги Tag.s

• Описание функциональных операций

Регистрирует тэг с номером тэга, обозначенным адресатом операции подтверждения и восстановле-ния дисплея. А. "." нужно добавить в конце строки номера тэга. Переменные parea, plock, patyp, palit, pstit и pflit должен иметь доступ тэга, указанный DefVar ().

lkz квалификации как адресат операций подтверждения и восстановления дисплея, тэг должен содержать следующие условия:

\* Правильный доступ тэга выполнен для переменных parea, plock, patyp, palit, pstit и pflit.

\* Категория сигнала указывает, что parea - в пределах сигнальной группы монитора.

Если выполнены вышеизложенные условия, тэг для адресата операции подтверждения и восстановления, определяется следующими условиями.

\* Операция подтверждения

Если сигнальный метод ( patyp) - B2, и pstit и pflit не должны быть 0.

Если сигнальный метод ( patyp) является другим, чем B2, pflit не должен быть 0.

\* Операция восстановления дисплея

Если сигнальный метод ( patyp) - A, и palit и pflit не должны быть 0. и pstit не должен быть 0. Если сигнальный метод ( patyp) является другим, чем B2, palit и pstit не должен быть 0. и pflit не должен быть0.

Эта функции обратного вызова должна быть выполнена при инициализации с INIT=5 или INIT=1.

• Возвращаемое значение Нет.

SetCnfColDefV (void \*\_pType, char \*type, void \*\_pAlit, char \*alit,

void \*\_pStit, char \*stit, void \*\_pFlit, char \*flit, void \*\_pArea, char \*area, void \*\_pAtyp, char \*atyp, void \*\_pLock,

char \*lock, void \*\_pVar, char \*pv, void \*\_pDcml, char \*dcml, void \* pAgrd, char \*agrd,

void \* pScn, char \*scn, void \* pAmon, char \*amon, void \* pMon, char \*mon)

• Описание Параметра

_ptype	Указатель на переменную, где введен результат доступа тэга ТҮРЕ.
Туре	Название {имя} атома тэга (ТҮРЕ)
_pAlit	Указатель на переменную, где введен результат доступа тэга ALIT.
alit	Название {имя} атома тэга (ALIT)
_pStit	Указатель на переменную, где введен результат доступа тэга STIT.
stit	Название {имя } атома тэга (АТІТ)
_pFlit	Указатель на переменную, где введен результат доступа тэга FLIT.
flit	Название {имя} атома тэга (FLIT)
_pArea	Указатель на переменную, где введен результат доступа тэга AREA.
area	Название {имя} атома тэга (AREA)
_Atype	Указатель на переменную, где введен результат доступа тэга АТҮР.
atype	Название {имя} атома тэга (АТҮР)
_pLock	Указатель на переменную, где введен результат доступа тэга LOCK.
lock	Название {имя} атома тэга (LOCK)
_pVar	Указатель на переменную, где введен результат доступа тэга PV
pv	Название {имя} атома тэга (PV)
_pDcml	Указатель на переменную, где введен результат доступа тэга DCML.
dcml	Название {имя } атома тэга (DCML)
_pAgrd	Указатель на переменную, где введен результат доступа тэга AGRD.
agrd	Название {имя } атома тэга (AGRD)
_pScn	Указатель на переменную, где введен результат доступа тэга SCN.
scn	Название {имя } атома тэга (SCN)

_pAmon	Указатель на переменную, где введен результат доступа тэга АМОНА.
mon	Название {имя} атома тэга (АМОН)
_pMon	Указатель на переменную, где введен результат доступа тэга MON.
mon	Название {имя} атома тэга (MON)

Чтение и наборе данных атома тэга, ТҮРЕ, ALIT, STIT, FLIT, AREA, LOCK, ATYP, PV,

DCML, AGRD, SCN, AMOH, и MON. Если Вы не хотите исполнять никакого чтения, определите 0 для указателей на переменные и названия {имена} атома тэга.

Эта функция обратного вызова должна быть выполнена с \_INIT=1 с тех пор DefVar () выполняется с набором типа: 1 (собираемый периодически).

• Возвращаемое значение Нет.

SetCnfColDefV10 (void \*\_pType, char \*type, void \*\_pAlit, char \*alit, void \*\_pStit, char \*stit, void \*\_pFlit, char \*flit, void \*\_pArea, char \*area, void \*\_pAtyp, char \*atyp, void \*\_pLock, char \*lock, void \*\_pVar, char \*pv, void \*\_pDcml, char \*dcml, void \*\_pAgrd, char \*agrd)

#### • Описание Параметра

_ptype	Указатель на переменную, где введен результат доступа тэга ТҮРЕ.
Туре	Название {имя } атома тэга (ТҮРЕ)
_pAlit	Указатель на переменную, где введен результат доступа тэга ALIT.
alit	Название {имя} атома тэга (ALIT)
_pStit	Указатель на переменную, где введен результат доступа тэга STIT.
stit	Название {имя } атома тэга (ATIT)
_pFlit	Указатель на переменную, где введен результат доступа тэга FLIT.
flit	Название {имя} атома тэга (FLIT)
_pArea	Указатель на переменную, где введен результат доступа тэга AREA.
area	Название {имя} атома тэга (AREA)
_Atype	Указатель на переменную, где введен результат доступа тэга АТҮР.
atype	Название {имя} атома тэга (АТҮР)
_pLock	Указатель на переменную, где введен результат доступа тэга LOCK.
lock	Название {имя} атома тэга (LOCK)
_pVar	Указатель на переменную, где введен результат доступа тэга PV
pv	Название {имя} атома тэга (PV)
_pDcml	Указатель на переменную, где введен результат доступа тэга DCML.
dcml	Название {имя} атома тэга (DCML)
_pAgrd	Указатель на переменную, где введен результат доступа тэга AGRD.
agrd	Название {имя } атома тэга (AGRD)

• Описание функциональных операций

Чтение и набор данных атома тэга, ТҮРЕ, ALIT, STIT, FLIT, AREA, ATYP, LOCK, PV, DCML, и AGRD. Если Вы не хотите исполнять ни какого чтения, определите 0 для указателей на переменные, и отметьте название {имя} атома.

Эта функция обратного вызова должна быть выполнена с \_INIT=1 с тех пор DefVar () выполняется с набором типа: 1 (собираемый периодически).

• Возвращаемое значение Нет.

### SetDirectDO (char \*tag, int num, int \_pChk)

#### • Описание Параметра

tag	Номер тэга
num	DO Номер
_pChk	DO Выводят Состояние

• Описание функциональных операций

Выводит тэг DO, с номер DO, обозначенным num, среди DO.s с обозначенным номером тэга.

А "." нужно добавить в конце строки номера тэга.

Регистрация тэга SetAlarmConf () и отдаленная операция регистрации InitCntrlDO () должна быть выполнена заранее.

DO выходы разрешены, когда состояние DO выходов, обозначенное \_pChk - 1.

\_pChk может быть установлен, используя функцию ChkDirect (), который проверяет четыре типа состояния вывода, DO.

Этот обратный вызов (call back) функционирует, чтобы вывести DO.s фактически, когда несколько DO.s выходов используют SetDirectStart () и SetDirectEnd () соответственно до и после.

• Возвращаемое значение Нет.

• Замечание Gismo специальная функция.

#### SetDirectStart () SetDirectEnd ()

• Описание Параметра Нет

• Описание функциональных операций

SetDirectStart () и SetDirectEnd () - функции, которые могут исполнить DO вывод, обычно выполнен-ный SetDirectDO () одной операцией.

SetDirectStart () устанавливает DO пакетный вывод, a SetDirectEnd () выполняет DO пакетный вывод.

Если SetDirectStart () вызов DO выхода не выполнен, SetDirectEnd () также не вызывают.

• Возвращаемое значение Нет.

• Замечание Gismo специальная функция.

#### SetPutDataDbl (char \*tagAtom, double data) SetPutDataInt (char \*tagAtom, int data)

• Описание Параметра tagatom Название {имя } атома тэга data Устанавливаемое значение

#### • Описание функциональных операций

Пишет значение установки, полученное из данных атома тэга, обозначенного tagAtom.

Когда данные написаны в нескольких атомах тэга одновременно, использовать StartPutData () и EndPutData () функции соответственно до и после обратного вызова (call back) функции, чтобы эффективно {фактически} исполнить запись данных.

• Возвращаемое значение Нет.

• Замечание Gismo специальная функция

#### SetRTVarInt (char \*name, int val) SetRTVarReal (char \*name, real val) SetRTVarStr (char \*name, char val)

• Описание Параметра		
name	Строка переменную	
val	Значение переменной	

• Описание функциональных операций

SetRTVarInt () / SetRTVarReal () / SetRTVarStr () устанавливают значение, обозначенное val, для переменной с названием {именем}, обозначенным name. SetRTVarInt () устанавливает целочисленное значение, SetRTVarReal () устанавливает реальное значение, a SetRTVarStr () устанавливает строку. Эта функция обратного вызова типично (обычно) используется, чтобы определить различные используемые переменные в модели, когда новая модель приводится в качестве примера RuntimeInstance ().

• Возвращаемое значение Нет.

#### SetTGATM (char \*tag, char \*atom)

• Описание Параметра

tag Номер тэга (...) atom Название {имя} атома

• Описание функциональных операций

Подключает номер тэга, обозначенный tag и названием {именем} атома, обозначенным atom и возвращает результат.

• Возвращаемое значение

Строка, которая является результатом подключения строки тэга и строки атома.

#### SetTitleText (char \*str)

• Описание Параметра str

Строка для области заголовка

• Описание функциональных операций

Отображает символ, обозначенный str в области заголовка.

• Возвращаемое значение Нет.

# SetVarChk (char \*TagAtom, int TagType, int \_pBtn, void \*\_pDlgBody)

• Описание Параметра

TagAtom	Отмечают название {имя} атома
ТадТуре	Тип тэга
_pBtn	Состояние кнопки дисплея диалога
_pDlgBody	Состояние дисплея диалога

Отображает диалог установки данных, указанного атома тэга.

Проверяет состояние кнопки дисплея диалога и состояние дисплея диалога и отображает диалог уста-новки данных, когда выполнены следующие условия.

\* Если состояние кнопки дисплея диалога допускается.

Состояние кнопки дисплея диалога находится в одном из следующих.

- 0: действующая кнопка
- Не 0: кнопка, не действующая

\* Если состояние дисплея диалога не отображено.

Состояние дисплея диалога находится в одном из следующих.

0: диалог не отображен.

Не 0: диалог отображается.

• Возвращаемое значение

0 для ошибки параметра или запроса от другого чем Gismo. Иначе 1.

• Замечание Gismo специальная функция.

SetVarAtom (char \*TagAtom, double Max, double Min, int type, int size, int dtype,

int dcml, \_pBtn, void \*\_pDlgBtn)

SetVarAtomSP (char \*TagAtom, double Max, double Min, int type, int size, int dtype,

int dcml, \_pBtn, void \*\_pDlgBtn, char \*atmName)

<ul> <li>Описание Параметра</li> </ul>	
TagAtom	Отмечают название {имя} атома
Max	Максимальное значение
Min	Минимальное значение
type	Тип Проверки Данных
size	Размер входных цифр
dtype	Тип данных
dcml	Десятичное место (разряд)
_pBtn	Состояние кнопки дисплея диалога
_pDlgBody	Состояние дисплея диалога
atmName	Название {имя} атома дисплея диалога

• Описание функциональных операций

Отображает данные, устанавливающие диалог указанного атома тэга, если состояние кнопки дисплея диалога «действующий» и состояние дисплея диалога «не отображен». Размер области ввода данных зависит от размера входных цифр.

Для входных данных исследование диапазона указанного максимального значения и минимального значения основано на типе проверки данных.

Тип проверки данных - один из следующих.

 $1: \leq Min \ X \leq Max, \ 2: \ Min \ <\!\! X \leq Max, \ 3: \leq Min \ X \leq Max, \ 4: \ Min \ <\!\! X \leq Max$ 

Тип данных - один из следующих.

1: Целочисленный тип, 2: Реальный тип, 3: Строковый тип

Десятичное место эффективно, когда тип данных - реальный тип.

Состояние кнопки дисплея диалога указывается любе 0: кнопка не действующая, не 0: кнопка действующая.

Состояние дисплея диалога указывается любое 0: диалог не отображен, не 0: диалог отображен.

SetVarAtomSP отображает указанное название {имя} атома дисплея диалога в диалоге руководства.

• Возвращаемое значение

0 для ошибки параметра или запроса от другого чем Gismo. Иначе 1.

• Замечание Gismo специальная функция.

#### SetVarInt (char \*TagAtom, int M ax, int M in, int type, int size, \_pBtn, void \*\_pDlgBtn) SetVarIntSP (char \*TagAtom, int M ax, int M in, int type, int size, \_pBtn, void \*\_pDlgBtn, char \*atmName)

• Описание Параметра	
TagAtom	Отмечают название {имя} атома
Max	Максимальное значение
Min	Минимальное значение
type	Тип Проверки Данных
size	Размер входных цифр
pBtn	Состояние кнопки дисплея диалога
pDlgBody	Состояние дисплея диалога
atmName	Название {имя} атома дисплея диалога

• Описание функциональных операций

Отображает диалог установки данных Целочисленного типа, указанного атома тэга если кнопка дисплея диалога в действующем состоянии, и состояние дисплея диалога «не отображено». Размер области ввода данных зависит от размера входных цифр.

Для входных данных исследование диапазона указанного максимального значения и

минимального значения основано на типе проверки данных.

Тип проверки данных - один из следующих.

 $1: \le Min X \le Max, 2: Min \le X \le Max, 3: \le Min X \le Max, 4: Min \le X \le Max$ 

Тип данных - один из следующих.

1: Целочисленный тип, 2: Реальный тип, 3: Строковый тип

Десятичное место эффективно, когда тип данных - реальный тип.

Состояние кнопки дисплея диалога указывается любе 0: кнопка не действующая, не 0: кнопка действующая.

Состояние дисплея диалога указывается любое 0: диалог не отображен, не 0: диалог отображен.

SetVarIntSP отображает указанное название {имя} атома дисплея диалога в диалоге руководства.

• Возвращаемое значение

0 для ошибки параметра или запроса от другого чем Gismo. Иначе 1.

• Замечание

Gismo специальная функция.

SetVarReal (char \*TagAtom, real Max, real Min, int type, int size, int dcml, \_pBtn, void \*\_pDlgBtn, char \*atmName) SetVarRealSP (char \*TagAtom, real Max, real Min, int type, int size, int dcml, \_pBtn, void \* pDlgBtn, char \*atmName)

• Описание Параметра

TagAtom	Отмечают название {имя} атома
Max	Максимальное значение
Min	Минимальное значение
type	Тип Проверки Данных
size	Размер входных цифр
dcml	Десятичное место (разряд)
_pBtn	Состояние кнопки дисплея диалога
_pDlgBody	Состояние дисплея диалога
atmName	Название {имя} атома дисплея диалога

• Описание функциональных операций

Отображает диалог установки данных реального типа, указанного атома тэга если состояние кнопки дисплея диалога является действующим, а состояние дисплея диалога «Не отображено». Размер области ввода данных зависит от размера входных цифр.

При отображении текущего значения, десятичная точка отображена согласно указанному десятично-му месту (разряду).

Для входных данных исследование диапазона указанного максимального значения и минимального значения основано на типе проверки данных.

Тип проверки данных - один из следующих.

1:  $\leq$  Min X  $\leq$  Max, 2: Min <X < Max, 3:  $\leq$  Min X < Max, 4: Min <X  $\leq$  Max Тип данных - один из следующих.

1: Целочисленный тип, 2: Реальный тип, 3: Строковый тип

Десятичное место эффективно, когда тип данных - реальный тип.

Состояние кнопки дисплея диалога указывается любе 0: кнопка не действующая, не 0: кнопка действующая.

Состояние дисплея диалога указывается любое 0: диалог не отображен, не 0: диалог отображен.

SetVarAtomSP отображает указанное название {имя} атома дисплея диалога в диалоге руководства.

• Возвращаемое значение

0 для ошибки параметра или запроса от другого чем Gismo. Иначе 1.

• Замечание Gismo специальная функция.

SetVarStr (char \*TagAtom, int M ax, int M in, int type, \_pBtn, void \*\_pDlgBtn) SetVarStrSP (char \*TagAtom, int M ax, int M in, int type, \_pBtn, void \*\_pDlgBtn, char \*atmName)

• Описание Параметра

TagAtom	Отмечают название {имя} атома
Max	Максимальное значение
Min	Минимальное значение
type	Тип Проверки Данных
pBtn	Состояние кнопки дисплея диалога
pDlgBody	Состояние дисплея диалога
atmName	Название {имя} атома дисплея диалога

Отображает диалог установки данных строкового типа указанного атома тэга, если состояние кнопки дисплея диалога является действующим, а состояние дисплея диалога «Не отображено». Размер области ввода данных зависит от максимального значения.

Для входных данных исследование диапазона указанного максимального значения и минимального значения основано на типе проверки данных.

Тип проверки данных - один из следующих.

 $1: \le Min X \le Max, 2: Min \le X \le Max, 3: \le Min X \le Max, 4: Min \le X \le Max$ 

Тип данных - один из следующих.

1: Целочисленный тип, 2: Реальный тип, 3: Строковый тип

Десятичное место эффективно, когда тип данных - реальный тип.

Состояние кнопки дисплея диалога указывается любе 0: кнопка не действующая, не 0: кнопка действующая.

Состояние дисплея диалога указывается любое 0: диалог не отображен, не 0: диалог отображен.

SetVarAtomSP отображает указанное название {имя} атома дисплея диалога в диалоге руководства.

• Возвращаемое значение

0 для ошибки параметра или запроса от другого чем Gismo. Иначе 1.

• Замечание Gismo специальная функция.

#### StartPutData () EndPutData ()

• Описание Параметра Нет

• Описание функциональных операций

StartPutData () и EndPutData () - функции, которые могут исполнить запись данных атома тэга, обычно

выполненные одной операцией SetPutDataInt () или SetPutDataDbl (). StartPutData () устанавливает запись пакета данных, a EndPutData () выполняет пакет мастер (wrigin) данных атома тэга. Даже если StartPutData () вызван, DO выход, не выполнен, если EndPutData () также не вызван.

• Возвращаемое значение Нет.

• Замечание Gismo специальная функция.

#### TankLevelBar (real level, real lmin, real lmax)

• Описание Параметра level данные lmin нижняя граница lmax верхняя граница

• Описание функциональных операций

Отображает заполнение емкости на рисунке в соответствии с данными.

• Возвращаемое значение Нет.

### ValueString (real \*\_pData, int Type, int decml)

• Описание Параметра	
_pData	Реальная переменная
Туре	Тип данных (0 установленный)
decml	Десятичное место

• Описание функциональных операций Создает строку реальных данных с десятичным указанным местом.

Возвращаемое значение
 Возвращает числовую строку значения.

### MakeAIStsBlock (void \*block, void \*trigger, char \*tagno)

• Описание Параметра

block	Переменная блокировки доступа данных для значения числового дисплея
trigger	Переменную для уведомления изменения данных доступа тэга
tagno	Номер тэга

• Описание функциональных операций

Интегрирует доступ тэга, обозначенный tagno и возвращает значения аналогового тэга в переменную, обозначенную block, для числового дисплея. Переменная спускового механизма (trigger) изменяется, когда данные изменяются на противоположные. Функция MakeAIStsBlock используется с DispAIBlinkStdCol (). Когда происходит тревога, функция исполняет дисплей аналогового значения в сигнальном цвете, процесс останова

нежелательного мерцания, и процесс внутреннего восстановления дисплея. Дисплей аналогового значения ограничен значением PV.

• Возвращаемое значение Нет.

Замечание
 Использование ограничено следующим.
 \_INT
 =1
 call MakeAIStsBlock(& pBlock, & pTrigger, \$TAGNO\$)

#### DispAlBlinkStdCol (void \*block, int obj, int blink)

• Описание Параметра

 block
 Переменная блокировки доступа данных метка для отображения числового

 значения
 о

 obj
 Тип объекта

 blink
 Счетчик мигания

• Описание функциональных операций

Отображает изменение аналогового тэга, обозначенного obj, используя block переменную для отображения числового значения, возвращенную из функции MakeAIStsBlock (). DispAlBlinkStdCol () функция используется с MakeAIStsBlock (). Это выполняет отображение аналогового значения, сигнального цвета, когда происходит тревога, процесс останова нежелательного мерцания, и восстановление отображения внутренний процесс. Дисплей аналогового значения выполнен, когда тип объекта, обозначенный obj Text. Символ (0 бит), Строка (1 бит), или Заполнение (2 бит) определяется obj.

• Возвращаемое значение Нет.

• Замечание Использование ограничено следующим: \_INT =1 call MakeAIStsBlock(&\_pBlock, &\_pTrigger, \$TAGNO\$) \_pTrigger =\* call DispAIBlinkStdCol(&\_pBlock, 1, \_BLINKCT)

Но указанное значение различается типом объекта, это обращение obj.

# MakeDIStsBlock (void \*block, void \*trigger, char \*tagno, int bit1, int bit2, int bit3, int bit4, int bit5, int bit6, int bit7, int bit8)

#### • Описание Параметра

block Переменная блокировки доступа данных для значения числового дисплея trigger Переменную для уведомления изменения данных доступа тэга tagno Номер тэга bit1 ~ bit8 Разрядный Номер (от 0 до 7)

• Описание функциональных операций

Интегрирует доступ тэга, обозначенный tagno и возвращает значения аналогового тэга в переменную, обозначенную block, для числового дисплея. Переменная спускового механизма изменяется когда изменяются данные тэга, к которым обращаются. Определите разрядные числа {номера} bit1 - bit8, которые контролируют состояние цифрового тэга. Если используется разрядный номер 0, 3, или 5, определить: bit1=0, bit2=3, и bit3=5.

Функция MakeDlStsBlock ()используется с DispDlBlinkStdCol ()/DispDlBlinkUsrCol ()/ DispDlBlinkUsrCol2 (). Это включает сигнальный цветной дисплей, когда происходит тревога, процесс останова нежелательного мерцания, и внутренний процесс восстановления дисплея цифрового тэга.

Если никакая сигналлизация, для цифрового тэга, не зарегистрирована, никакой процесс, типа изменяющегося цвета, не может быть выполнен.

• Возвращаемое значение Нет.

Замечание Использование ограничено следующим: \_INT =1 cakk MakeDIStsBlock(&\_pBlock, &\_pTrigger,\$TAGNO\$) \_pTrigger =\* call DispDIBlinkStdCol(& pBlock, 1, BLINKCT)

#### DispDIBlinkStdCol (void \*block, int obj, int blink) DispDIBlinkStdCol1 (void \*block, int offcol, int obj, int blink) DispDIBlinkStdCol2 (void \*block, int oncol, int offcol, int obj, int blink)

• Описание Параметра

block Переменная блокировки доступа данных метка для отображения числового значения оbj Тип объекта

blink	Счетчик мигания
oncol	Сигнальный цвет возникновения
offcol	Сигнальный цвет восстановления

• Описание функциональных операций

Отображает изменение аналогового тэга, обозначенного obj, используя block переменную для отображения числового значения, возвращенную из функции MakeAIStsBlock ().

DispDlBlinkStdCol () функция используется с MakeDlStsBlock (). Это выполняет отображение цветной сигнализации, когда происходит тревога, процесс останова нежелательного мерцания, и процесс восстановления внутреннего дисплея цифрового тэга.

Символ (0 бит), Строка (1 бит), или Заполнение (2 бит) определяется obj.

Функция DispDlBlinkStdCol1 ()используется, чтобы позволить пользователю определять цвет для сигнализации восстановления.

Функция DispDlBlinkStdCol2 ()используется, чтобы позволить пользователю определять цвет для сигнализации возникновения и восстановления.

• Возвращаемое значение Нет.

Замечание Использование ограничено следующим. \_INT =1 call MakeDIStsBlock(&\_pBlock, &\_pTrigger, \$TAGNO\$) \_pTrigger call SispDIBlinkUsrCol(&\_pBlock, 1, \_BLINKCT)

Следующая функция используется для ненормативного цвета (зеленый) отображаемого при сигнализации восстановления.

DispDIBlinkUsrCol1(void\*block, int offcol, int obj, int blink)

offcol используется для указанного цвета при сигнализации восстановления.

Следующая функция используется для не-тревожного отображения цвета при сигнальном возникно-вении и восстановлении.

DispDIBlinkUsrCol2(void\*block, int offcol, int offcol, int obj, int blink)

oncol используется для указанного цвета при сигнальном возникновении.

offcol используется для указанного цвета при сигнальном восстановлении.

MakeDIStsNBlockStart (void \*block, void \*trigger, int num) MakeDIStsNBlockEnd (void \*block, void \*trigger) AddDIStsNBlock (void \*block, void \*trigger, char \*tag, int bit1, int bit2, int bit3,

int bit4, int bit5, int bit6, int bit7, int bit8)

#### DispDINBlinkStdCol (void\*body, int obj, int blink) DispDINBlinkUsrCol1 (void \*body, int offcol, int obj, int blink) DispDINBlinkUsrCol2 (void \*body, int oncol, int offcol, int obj, int blink)

#### • Описание Параметра

block	Переменная блокировки доступа данных для значения числового дисплея
trigger	Переменную для уведомления изменения данных доступа тэга
num	Указанный номер тэга
tagno	Номер тэга
bit1 ~ bit8	Разрядный номер (от 0 до 7)
blink	Тип объекта
blink	Счетчик мигания
oncol	Сигнальный Цвет Возникновения
offcol	Сигнальный Цвет Восстановления

• Описание функциональных операций

Эти функции - расширенные версии функций для отображения состояния цифрового тэга типа MakeDIStsBlock () и DispDIBlinkStdCol (), для обработки множественных тэгов. Тэг и бит определены

AddDIStsNBlock () между MakeDLStsNBlockStart () и MakeDLStsNBlockEnd (). Другие функции могут использоваться точно так же даже при том, что их названия {имена} различны.

• Возвращаемое значение Нет.

• Замечание Использование ограничено следующим.

INT =1 call MakeDIStsNBlockStart(& pBlock, & pTrigger, 4) call AddDIStsNBlock(& pBlock, & pTrigger, \$TAGNO1\$, bit1 1, bit1 2, bit1 3, bit1 4, bit1 5, bit1 6, bit1 7, bit1 8) call AddDIStsNBlock(& pBlock, & pTrigger, \$TAGNO2\$, bit2\_1, bit2\_2, bit2\_3, bit2\_4, bit2\_5,bit2\_6, bit2\_7, bit2\_8) call AddDIStsNBlock(& pBlock, & pTrogger, \$TAGNO3\$, bit3 1, bit3 2, bit3 3, bit3 4, bit3 5, bit3 6, bit3 7, bit3 8) call AddDIStsNBlock(& pBlock, & pTrigger, \$TAGNO4\$, bit4 1, bit4 2, bit4 3, bit4 4, bit4 5, bit4 6, bit4 7, bit4 8) call MakeDIStsNBlockEnd(& pBlock, & pTrigger) \_pTrigger

call DispDINBlinkStdCol(&\_pBolck, 6, \_BLINKCT)