Программируемая панель отображения.

Руководство по программированию.

(версия V1.27)

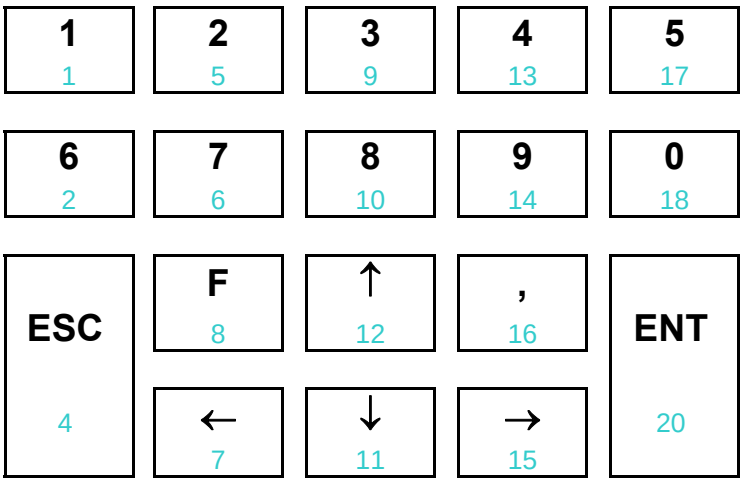
г. Зеленоград

СПЕЦИАЛИЗИРОВАННОЕ КОНСТРУКТОРСКОЕ БЮРО
ПРОМАВТОМАТИКА

www.skbpa

Считывание клавиатуры

Клавиатура панели отображения представляет собой матрицу со следующими кодами кнопок.



При нажатии на какую-либо кнопку её код кладётся в буфер. Буфер хранит 16 кодов.
При переполнении вновь нажимаемые кнопки не запоминаются до считывания кода хотя бы одного кода клавиши управляющим контроллером.

Программа контроллера должна периодически считывать статус буфера клавиатуры для определения наличия кодов нажатых кнопок.

Пример программы на языке ST, выполняющий эти функции для панелей с версией ПО 1.20 и выше.

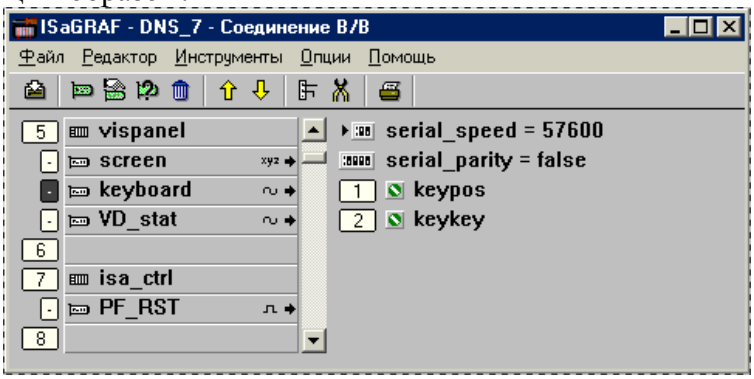
```
IO_rez:=OPERATE(keypos,1,5); (* читаем код клавиши *)
IF IO_rez<>64 THEN      (* если обмен состоялся *)
        IF keypos=-1 THEN       (* не сбросился указатель *)
                KeyKey:=-1; (* повторная инициализация *)
        END_IF;
        IF KeyKey<0 THEN  (* была рассинхронизация указателей позиции *)
                keypos:=-1;                     (* сбросить указатель позиции *)
                IO_rez:=OPERATE(keypos,1,5);
                IF IO_rez<>64 THEN (* сброс успешно завершён *)
                        (* скорее всего, произошёл рестарт панели, здесь можно *)
                        (* установить флаг для обработки этой ситуации *)
                        keypos:=0;
                END_IF;
        ELSE (* рассинхронизации не произошло *)
                key:=keykey;      (* читаем код клавиши *)
                IF key>0 THEN  (* есть коды клавиш в буфере *)
                        keypos:=keypos+1;         (* увеличиваем указатель позиции *)
                END_IF;
        END_IF;
END_IF;
```

```
IF key<>0 THEN (* есть необработанные кнопки *)
      CASE key OF (* обработка кнопок по коду *)
      1: ….. (* здесь пишется реакция для каждой кнопки*)
      9: …..
```

```
        END_CASE;
key:=0; (* клавиша обработана - очищаем *)
END_IF;
```

Пояснения по переменным:
IO_rez – статус обмена контроллера с панелью. Подробности смотрите в описании ПЛК.
Key – код необработанной клавиши. После обработки клавиши эта переменная должна быть обнулена.
Keykey – код клавиши. Описывается как ***выходная переменная***, но на самом деле после процедуры обмена в ней оказывается код клавиши или код ошибки.
Keypos – номер обрабатываемой клавиши. ***Выходная переменная***.

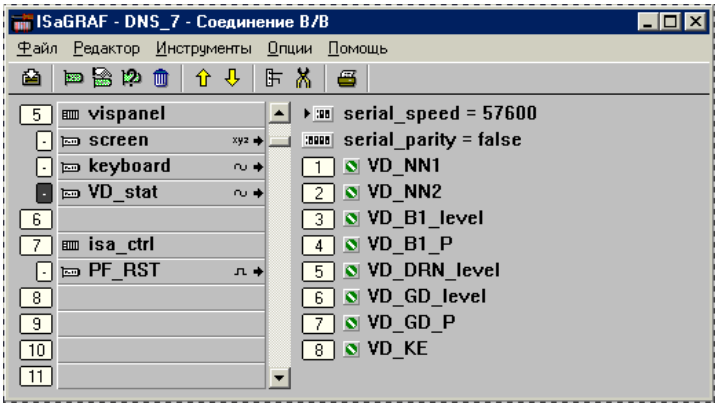
В карте соединений ввода/вывода переменные обработки клавиатуры необходимо назначить следующим образом:



Внимание!!! При переходе с версии ПО 1.1х на 1.2х необходимо переустановить библиотеку панели и заменить устройство vispanel в описателе соединений ввода/вывода.

Управление светодиодами

Управление светодиодами осуществляется путем записи в специализированное устройство вывода. К каждому светодиоду привязывается переменная целого типа.



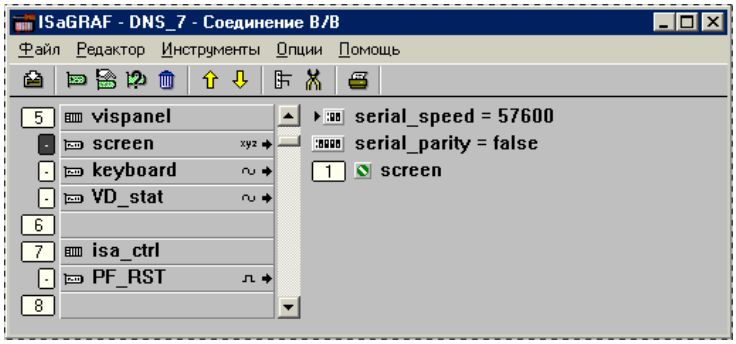
Зависимость состояния светодиода от содержимого переменной следующая:
0 – светодиод не горит
1 – светодиод горит зелёным цветом
2 – светодиод горит красным цветом

Организация экрана.

Экран панели организован в виде двух независимых слоёв – текстового и графического. Слои накладываются по принципу логического сложения. Размерность текстового слоя 53x30 символов. Размерность графического слоя 320x240 пикселей. Координаты отсчитываются от верхнего левого угла. В текстовом слое индикатор запоминает позицию последнего выведенного символа и при выдаче следующей текстовой строки, без привязки к координатной сетке, поместит её как продолжение предыдущей. После вызова команд работы с графикой, для корректного отображения текстовой строки, ей должна предшествовать команда позиционирования курсора.

Вывод текста

За вывод текста в устройстве ***vispanel*** отвечает переменная ***screen*** из раздела ***screen***.



Панель обрабатывает текстовую информацию из строковой переменной раздела ***screen*** следующим образом:

В текущую позицию курсора текстового слоя индикатора выводится вся текстовая информация, записываемая в переменную ***screen***. Пробелы игнорируются, для отображения пробела используется символ «_».

Если встречается символ «^», то следующий за ним символ интерпретируется как начало специальной команды. Пробелы в параметрах команды не допускаются. Необходимо соблюдать весь синтаксис команды, иначе она может выполниться не так, как ожидается. Обратите внимание на завершающую точку в ряде команд с численными параметрами.

Очистка экрана

«^C» - очистка всего текстового слоя экрана.
«^CP**n**.» - очистка с текущей позиции курсора до позиции, указанной в параметре «**n**».
«^CL» - очистка текущей строки с текущей позиции курсора до конца и переход на следующую строку.
«^CG» - очистка всего графического слоя экрана.

Позиционирование курсора

«^P**x**,**y**.» - установка курсора в позицию: столбец **x**, строка **y**. Левое верхнее знакоместо имеет координаты 1, 1. Нижнее правое – 53, 30.

Форма курсора

«^F**q**.» - задание формы курсора
q=0 – подчёркивающий курсор
q=1 – курсор в виде блока 5*7 точек

«^S**h**.» - делает курсор видимым при **h**=1 и невидимым при **h**=0.

Звуковые сигналы

«^B**f**,**t**.» – выдача звукового сигнала частотой **f** Гц и длительностью **t** миллисекунд.

Работа с макросами

Панель поддерживает загрузку файла макросов. Это сделано с целью уменьшения количества информации, передаваемой от контроллера за счёт использования ранее загруженной информации. Соответственно, уменьшается и размер управляющей программы ПЛК так как вместо выдачи длинных строк, можно вызывать макрос. Файл макросов создаётся специальной программой и загружается при помощи программы локального пульта управления (ЛПУ) через порт RS-232.

Правила применения макросов:

1. Описание макроса начинается с директивы «#DEF». Далее следует имя макроса. Имя должно начинаться с буквы, может содержать цифры. Запрещено использование символа «_»
2. Описание макроса заканчивается директивой «#END_DEF».
3. Если в макросе встречается выражение «%%X», где X – цифра, то это выражение заменяется на соответствующий по порядку параметр, описанный при вызове макроса.

Например:

Описание макроса:

```
#DEF levelB1
  ^UV180,28,23,50,%%1.
#END_DEF
```

Вызов макроса из программы

```
Pp:=24;  (* переменная соответствующая уровню *)
Screen:= '&levelB1('+MSG(pp)+')';
Io_rez:=OPERATE(Screen,1,0); (* вывод на панель *)
```

Вызов этого макроса отобразит вертикальную шкалу с координатами верхнего левого угла (180,23) , с шириной 23 и высотой 50 пикселей, заполненную на 24%.

Макросы могут быть вложенными, т.е. содержать в себе вызовы других макросов.

Графические команды

Рисование линий

«^Lx1,y1,x2,y2.» рисование сплошной линии из точки (x1,y1) в точку (x2,y2).

Рисование прямоугольников

«^Rx,y,w,h.» - рисование прямоугольника по верхнему левому углу (x,y) с шириной «w» и высотой «h».
«^Qx,y,w,h,mask.» - рисование прямоугольника по верхнему левому углу (x,y) с шириной «w» и высотой «h», заполняя шаблоном из параметра «mask».

Рисование шкал

«^UVx,y,w,h,proc.»
Рисование вертикальной шкалы в прямоугольнике (x1,y1,w,h). Параметр proc задаёт степень заполнения в процентах от всей шкалы. Заполнение происходит снизу вверх.

«^UHx,y,w,h,proc.» - то же самое, что и «^UV…», только отображается горизонтальная шкала, заполнение происходит слева направо.

Очистка области и отображение рисунка

«^Yx,y,offs.» – отображает спрайт в позиции (x,y) – левый верхний угол. Параметр «offs» определяет адрес бинарного образа (рассчитывается компилятором макросов).
«^YDx,y,offs.» - стирает область в позиции (x,y). Использует ссылку на спрайт для определения размеров области стирания.

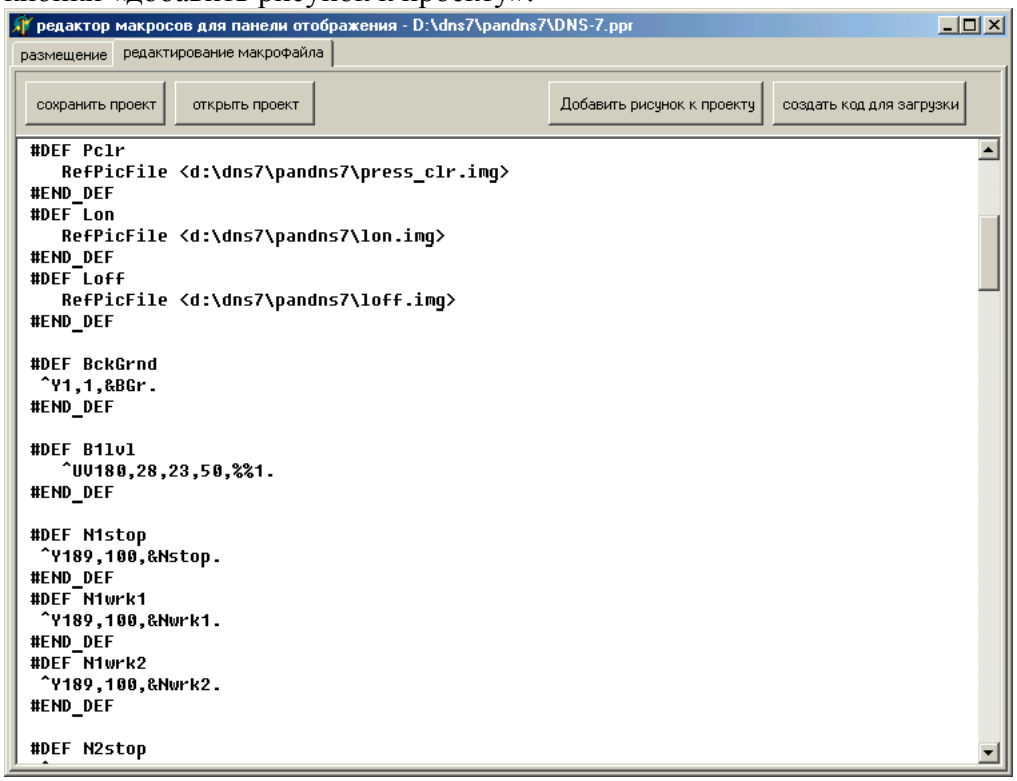
Компилятор макросов

Программа компилятора макросов обеспечивает:

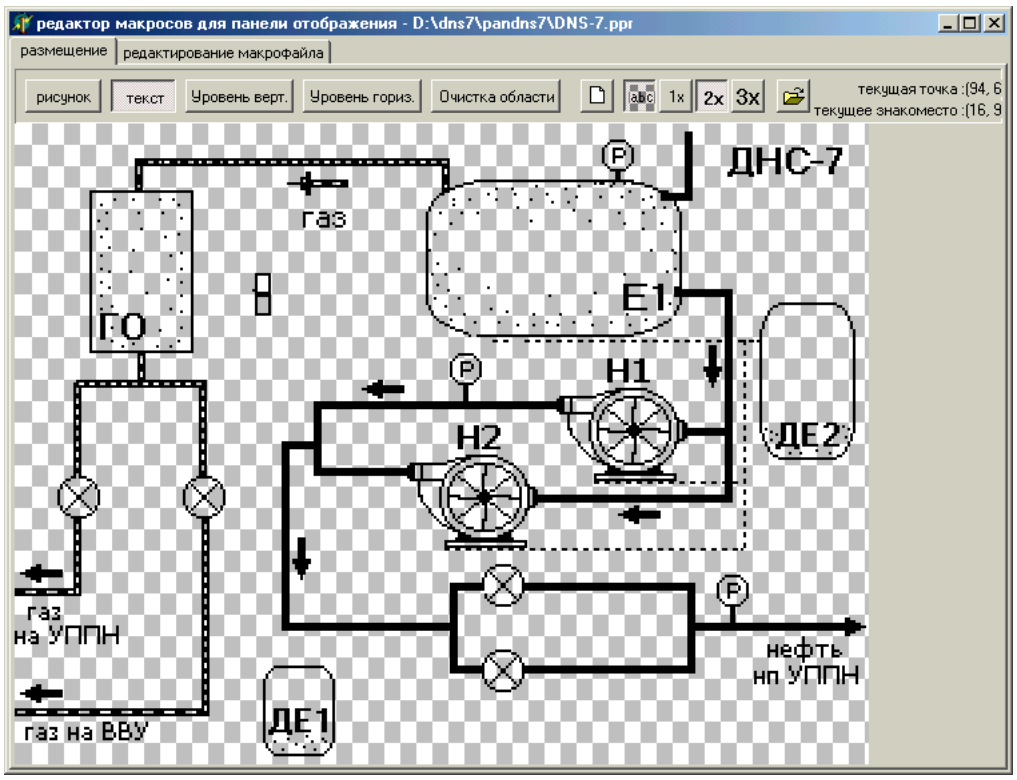
1. редактирование файла макросов;
2. удобство размещения спрайтов, шкал и текстовых полей на рабочем поле индикатора;
3. привязку спрайтов к их логическим именам;
4. создаёт код для загрузки в панель отображения;

Порядок создания проекта макросов.

1. Загрузить программу редактора макросов.
2. После загрузки программы выбрать закладку «редактирование макрофайла»
3. Если в проекте используются спрайты – добавить их ссылки к проекту при помощи кнопки «добавить рисунок к проекту».



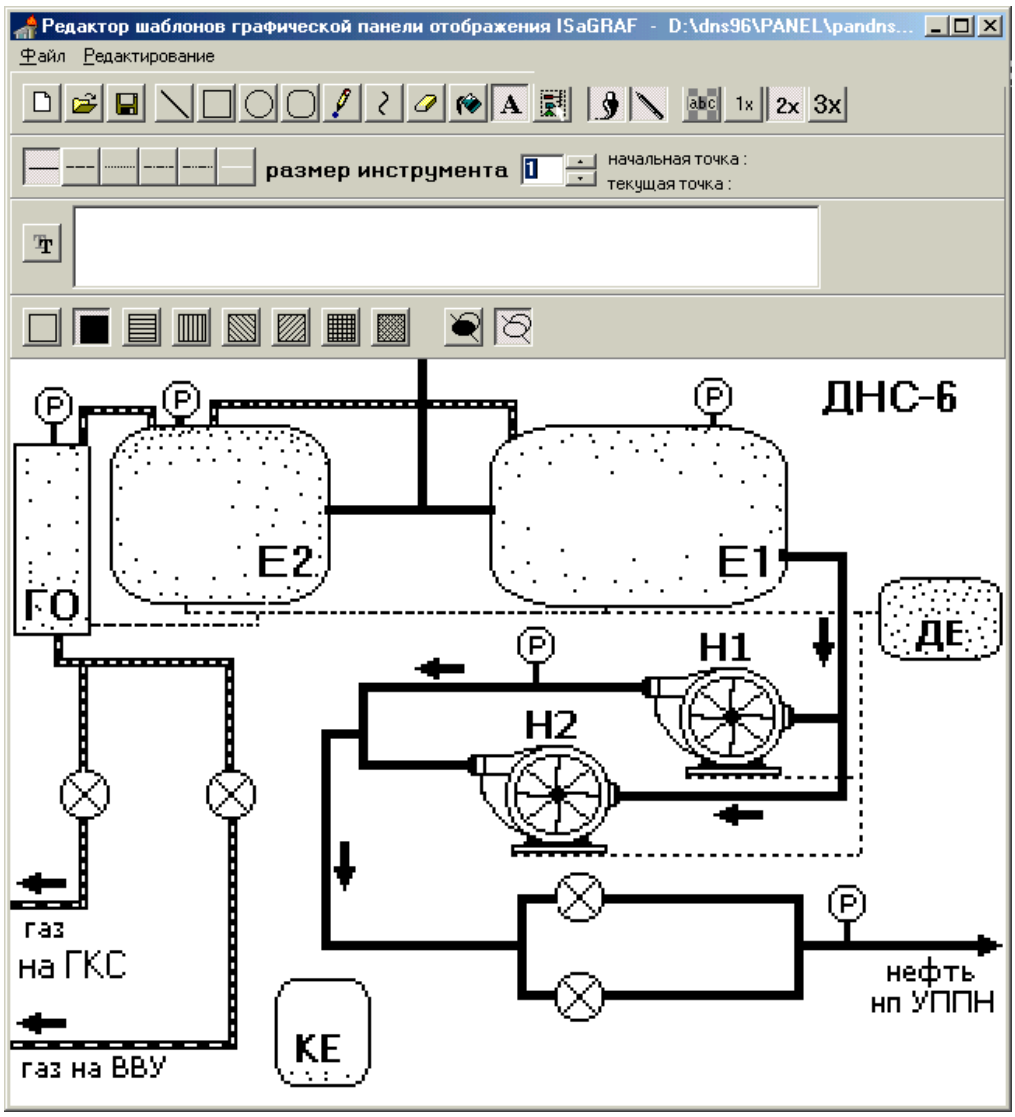
4. Далее можно перейти к размещению объектов, выбрав закладку «размещение». Для каждого размещаемого на рабочем поле объекта будет запрошено имя, через которое этот объект может быть вызван из ПЛК или из другого макроса.
5. В процессе размещения объектов, добавляются записи в файл макросов. Если произошло некорректное размещение, то можно вручную удалить или поправить последнюю запись.



6. По окончании работы над проектом, можно создать код загрузки для панели нажав кнопку «создать код для загрузки». При успешном завершении программа напишет, что файл «*.PDS» успешно создан. Он находится в той же директории, что и файл проекта. Для его загрузки пользуйтесь программой локального пульта версии 1.86 и выше.

Редактор изображений.

Редактор изображений позволяет создавать и редактировать изображения в формате, приемлемом для компилятора макросов. Поддерживается импорт изображений через буфер обмена.


Редактор шаблонов графической панели отображения ISaGRAF - D:\dns96\PANEL\pandns...
Файл Редактирование
размер инструмента
начальная точка :
текущая точка :
ДНС-6
Е2
Е1
ГО
ДЕ
Н1
Н2
газ
на ГКС
газ на ВВУ
КЕ
нефть
нп УППН