

Инструментальные  
средства  
технологического  
программирования.  
ISaGRAF  
(стандарт IEC-61131)

# Структура ISaGRAF



MS Windows



ISaGRAF  
Workbench

- Набор инструментов отладки приложения (пользователя).
- Устанавливается на компьютере с ОС

*Аппаратная/ОС  
специфика*

- Интерфейс с платформой
- Управление памятью
- Системные часы

- Windows NT
- Windows CE
- Embedded NT
- NT w/ RTX

- OS 9/9000
- MS DOS
- LynxOS
- QNX
- VxWorks
- И т.д. ....

- ThreadX
- Custom OS

- И т.д. ....

- Devicenet
- Profibus
- Ethernet
- Modbus
- Interbus
- и т.д. ....

функции

Системные функции

Runtime  
Target

Пользовательские  
функции

ISaGRAF Target

система исполнения, с помощью которой выполняются приложения пользователя.  
Устанавливается на контроллере (ядро и исполнительная часть)

# Структура ISaGRAF



## Компоненты ISaGRAF

- **ISaGRAF Workbench**

Набор инструментов для разработки и отладки приложений (отдельных проектов ISaGRAF и групп проектов).

Устанавливается на персональном компьютере с ОС Windows.

- **ISaGRAF Target**

Система исполнения (ядро ISaGRAF), с помощью которой выполняются приложения пользователя (проекты ISaGRAF).

Устанавливается на контроллере, как часть его встроенного базового программного обеспечения.

Дополняется библиотекой подпрограмм, которые вызываются из приложения ISaGRAF и исполняются непосредственно на процессоре контроллера.

# Поддерживаемая аппаратура и операционные системы

---



**ISaGRAF Target функционирует на процессорах**

- Intel x86
- Motorola

**На таких платформах, как:**

- PC, PCI, VME и специализированные аппаратные средства

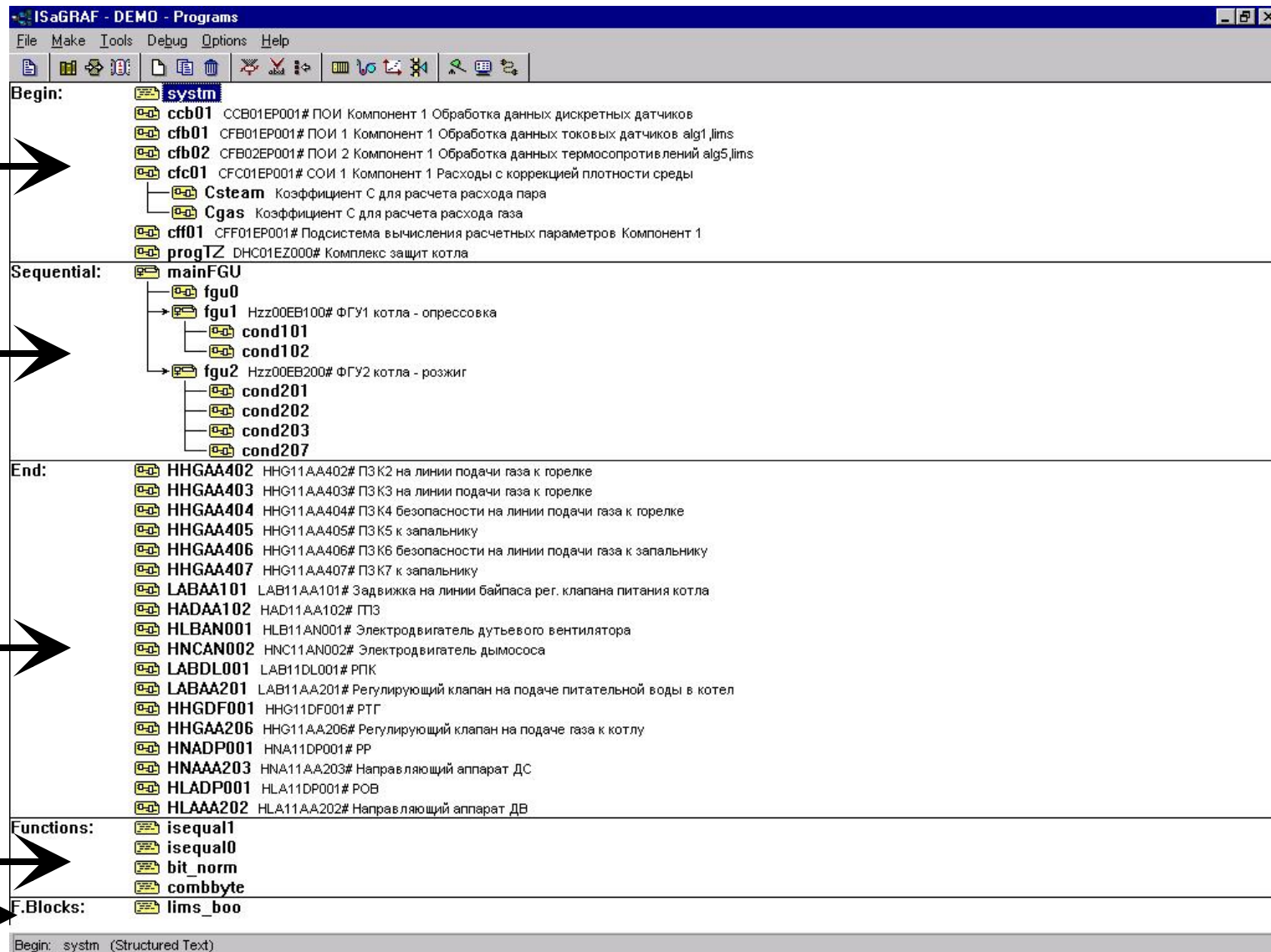
**ISaGRAF Target функционирует в следующих ОС:**

Windows NT, MS-DOS,

OS9, QNX, VxWorks, PSOS,

Pharlap

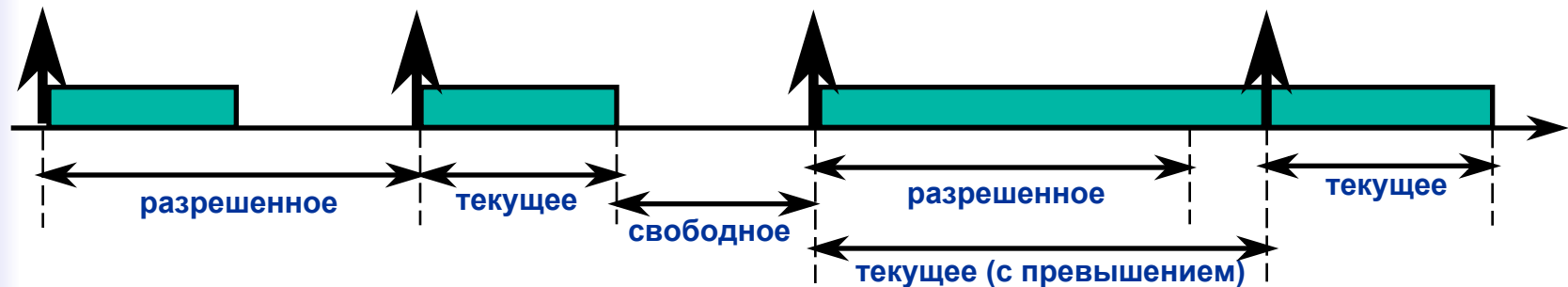
# Пример программы на ISaGRAF



# Правила выполнения

**ISaGRAF – синхронная система.**

- Программы приложения (проекта ISaGRAF) исполняются по шагам.
- Длительность исполнения очередного шага называется текущим временем цикла.
- Каждый шаг должен полностью завершиться. Только после этого может начаться выполнение следующего шага: по таймеру или сразу же после предыдущего шага.



## *Система обеспечивает*

Входная переменная сохраняет одно и то же значение в пределах цикла.  
Значения в каналах устройств вывода изменяются только один раз за цикл.  
Безопасную работу с одними и теми же переменными из разных программ.  
Оценку времени отклика для приложения и управление этим временем.

# Синхронизация



# Языки программирования ISaGRAF



---

**ISaGRAF поддерживает все 5 языков стандарта  
IEC 61131-3**

## **Графические**

- SFC**     **Sequential Function Chart** (ядро) – язык  
последовательных функциональных диаграмм
- FBD**     **Function Block Diagram** - язык функциональных  
блоковых диаграмм
- LD**      **Ladder Diagram** - язык релейной логики

## **Текстовые**

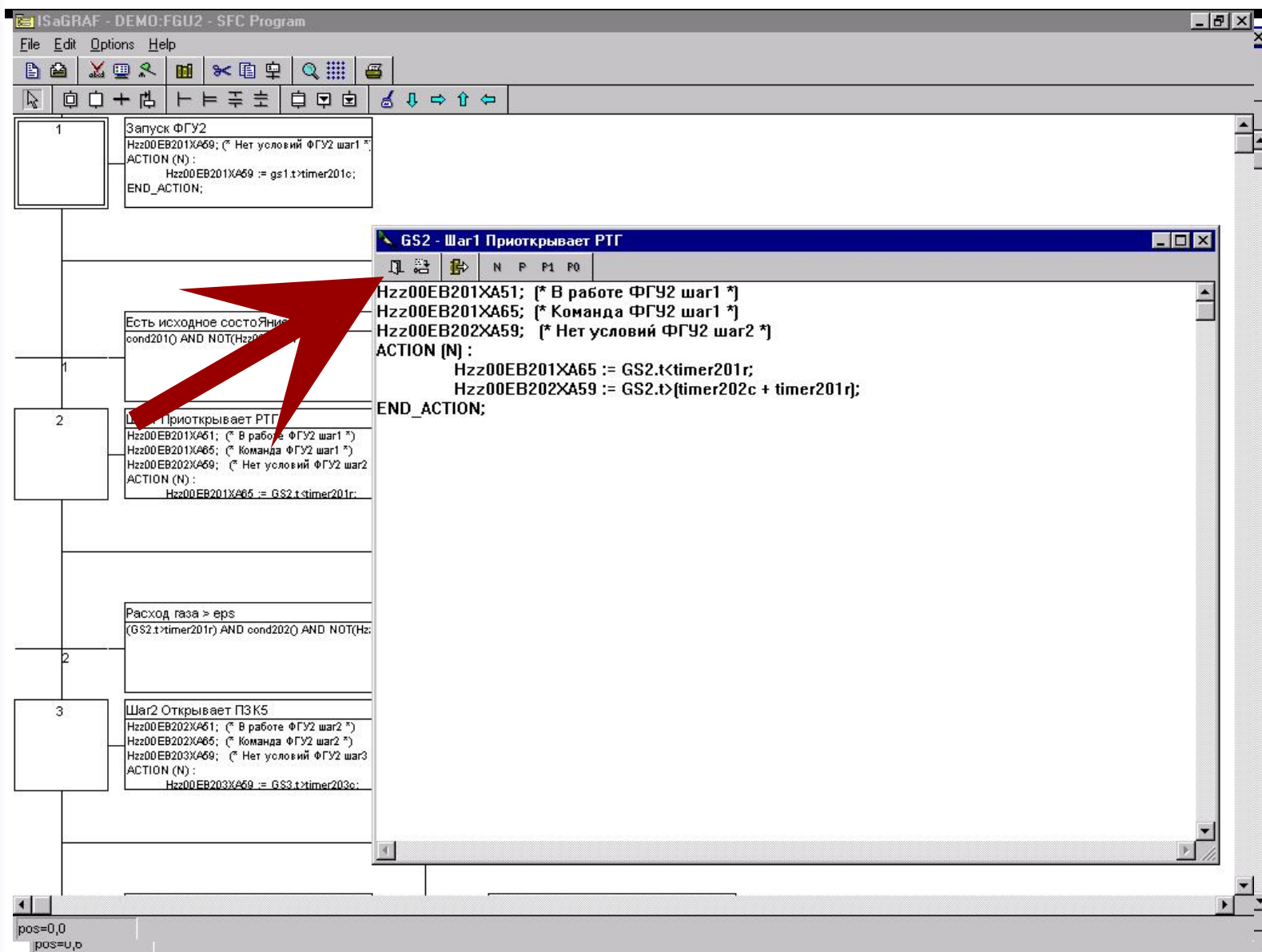
- ST**      **Structured Text** – язык структурированного текста
- IL**      **Instruction List** – язык инструкций



## **SFC** (Sequential Function Chart)

- Предназначен для последовательных действий
- Включает механизмы синхронизации
- Использует простые динамические правила
- Может представлять параллельные процессы
- Может быть использован как язык спецификаций

# Программа на языке SFC



# Языки программирования ISaGRAF



## **FBD (Function Block Diagram)**

- Для смешанных типов уравнений
- Большая библиотека блоков

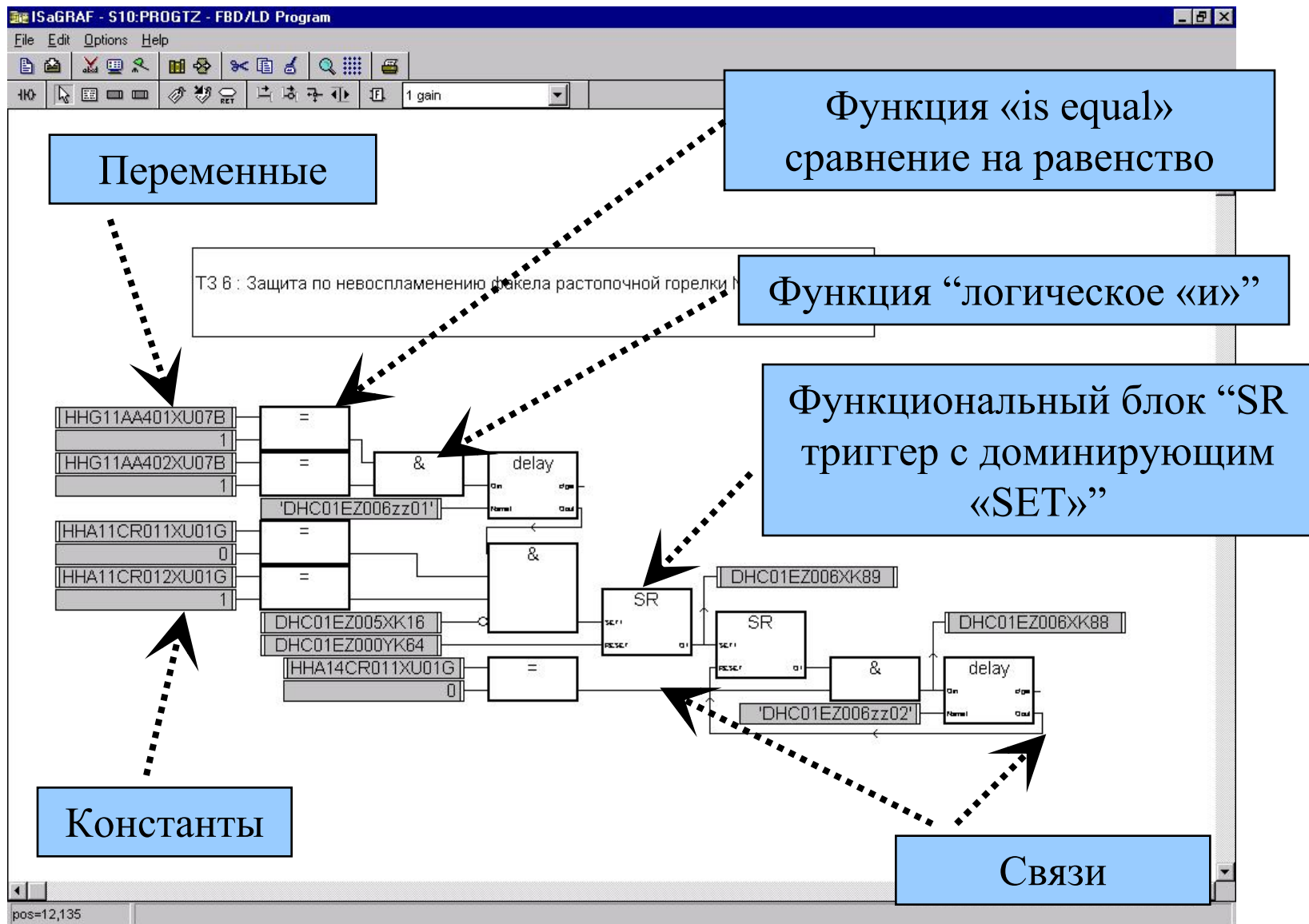
## **LD (Ladder Diagram)**

- Только для булевых уравнений
- Простые правила

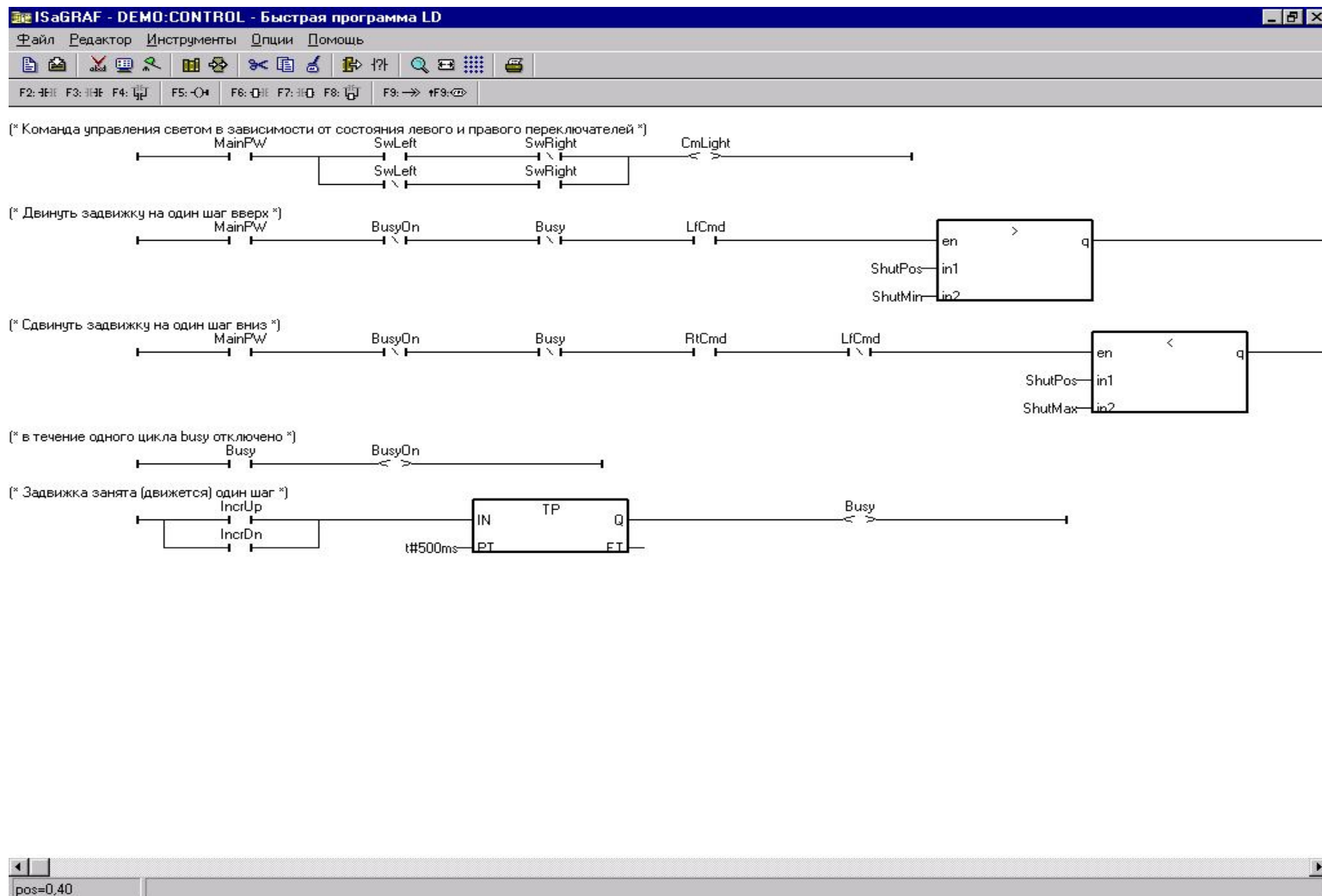


**Общий  
редактор**

# Программа на языке FBD



# Программа на языке LD



# Языки программирования ISaGRAF



## **ST** (Structured Text)

- Структурный язык
  - язык высокого уровня (подобен PASCAL)
- Высокая удобочитаемость исходного текста

## **IL** (Instruction List)

- Язык низкого уровня
  - ассемблер мнемонического процессора

## **C**

- Работает на predetermined интерфейсах (Функции или Блоки)
- Синхронный или асинхронный (доступ к ОС)
- Требуется не-ISaGRAF инструментальных средств

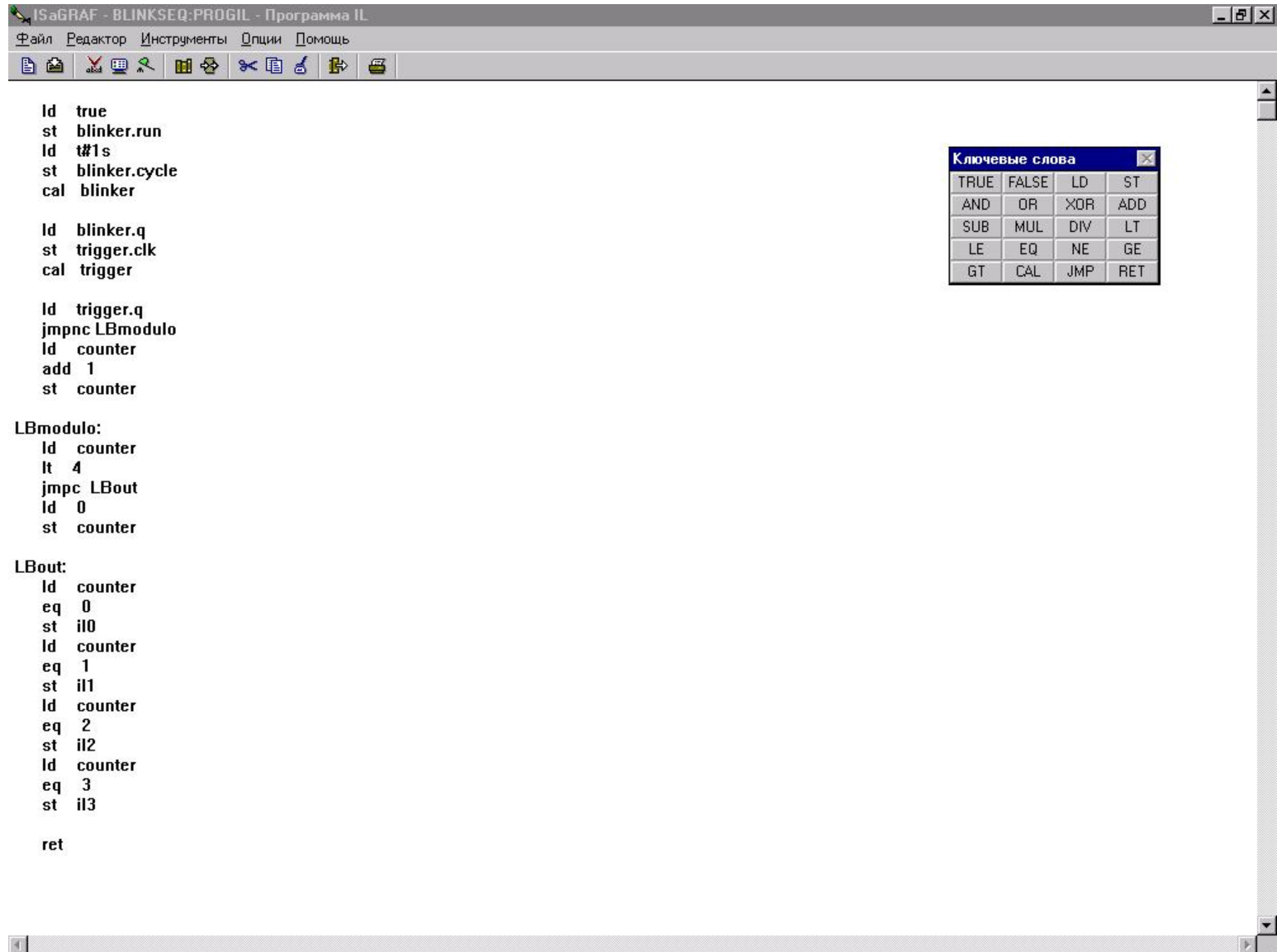
# Программа на языке ST

```
ISaGRAF - SMART_1:TEMPDEF - ST program
File Edit Options Help
[Icons]

TypeWireLocal := TypeWire;
END_IF;
Templnauth := 0;
CASE TypeWireLocal OF
    0: k1 := k1MF85; k2 := k2MF85; k3 := k3MF85;          (* провод МФ-85 *)
    1: k1 := k1MF100; k2 := k2MF100; k3 := k3MF100;      (* провод МФ-100 *)
    ELSE k1 := k1PKSA; k2 := k2PKSA; k3 := k3PKSA;        (* провод ПКСА-100/25 *)
END_CASE;
RealTemp := srcAirTemp;
IF (RealTemp > 60.0) OR (RealTemp < -60.0) THEN RealTemp := 0.0; END_IF;
curr := REAL(CurrentFA) * REAL(CurrentFA);               (* однопроводная схема *)
LocalTempOne := TemperatureOne + ClcPer * curr * (k1 + k2 * TemperatureOne) + k3 * (TemperatureOne - RealTemp);
curr := 0.3249 * curr;                                    (* двухпроводная схема *)
LocalTempTwo := TemperatureTwo + ClcPer * curr * (k1 + k2 * TemperatureTwo) + k3 * (TemperatureTwo - RealTemp);
IF DsbIClc2 OR (CurrentFA > I_min) THEN                    (* если расчет по 2-проводной схеме запрещен или то *)
    RealTemp := LocalTempOne;
    IntTemp := TmrstZ[-1, FANumber];
ELSE
    IF TmrstZ[0, FANumber] = 0 THEN                        (* если предистории еще не было *)
        RealTemp := LocalTempOne;
        IntTemp := TmrstZ[1, FANumber];                  (* запускаем ее *)
    ELSE
        IF TmrstZ[0, FANumber] > TimeHistory THEN        (* время предистории прошло *)
            IntTemp := TmrstZ[-1, FANumber];
            IF PositionFA THEN RealTemp := LocalTempTwo; ELSE RealTemp := LocalTempOne; END_IF; (* если автомат включен, выходная температура - по *)
        ELSE
            RealTemp := LocalTempOne;
        END_IF;
    END_IF;
END_IF;
IF RealTemp > T_MAX THEN RealTemp := T_MAX; END_IF;
ELSE
    Templnauth := 1;                                     (* сигнал тока недоуверен *)
    RealTemp := TemperatureOne;                          (* выставляем недоуверенность расчета температуры *)
END_IF;
IF EnableTZ THEN
    IF RealTemp <= REAL(TempLimit) THEN
        TempDefStateOut := 1;
    ELSIF TempDefState < 2 THEN
        IntTemp := FAOff(FANumber, TZCode);
        TempDefStateOut := 2;
        TempDefOff := ANA(RealTemp);
    END_IF;
END_IF;
```

Keywords	
:=	TRUE
FALSE	AND
OR	XOR
RETURN;	IF
THEN	ELSE
ELSIF	END_IF;
CASE	END_CASE;

# Программа на языке IL



ISaGRAF - BLINKSEQ:PROGIL - Программа IL

Файл Редактор Инструменты Опции Помощь

ld true  
st blinker.run  
ld t#1s  
st blinker.cycle  
cal blinker

ld blinker.q  
st trigger.clk  
cal trigger

ld trigger.q  
jmpnc LBmodulo  
ld counter  
add 1  
st counter

LBmodulo:  
ld counter  
lt 4  
jmpc LBout  
ld 0  
st counter

LBout:  
ld counter  
eq 0  
st il0  
ld counter  
eq 1  
st il1  
ld counter  
eq 2  
st il2  
ld counter  
eq 3  
st il3

ret

Ключевые слова

TRUE	FALSE	LD	ST
AND	OR	XOR	ADD
SUB	MUL	DIV	LT
LE	EQ	NE	GE
GT	CAL	JMP	RET



# Словарь переменных ISaGRAF

ISaGRAF - DEMO - CSTEAM - FBD/LD Program

File Edit Options Help

1 gain

ISaGRAF - DEMO - Global integers/real

File Edit Search Options Help

Name	Attrib.	Adc	Comment
LAB11CF001XQ10	[input, integer]	000	Сигнал от драйвера Расход питательной воды на котел
LAB11CF001XQ29	[internal, integer]	000	Нормированный(0-1000) Расход питательной воды на котел
LAB11CF001XQ30	[output, real]	000	Расход питательной воды на котел
LAB11CF001ZU30	[output, integer]	000	Статус Расход питательной воды на котел
HAD11CF002XQ10	[input, integer]	000	Сигнал от драйвера Расход пара за котлом
HAD11CF002XQ29	[internal, integer]	000	Нормированный(0-1000) Расход пара за котлом
HAD11CF002XQ30	[output, real]	000	Расход пара за котлом
HAD11CF002ZU30	[output, integer]	000	Статус Расход пара за котлом
HHG11CF004XQ10	[input, integer]	000	Сигнал от драйвера Расход газа на котел
HHG11CF004XQ29	[internal, integer]	000	Нормированный(0-1000) Расход газа на котел
HHG11CF004XQ30	[output, real]	000	Расход газа на котел
HHG11CF004ZU30	[output, integer]	000	Статус Расход газа на котел
LAB11CG001XQ10	[input, integer]	000	Сигнал от драйвера УП клапана рег. Уровня в барабане котла
LAB11CG001XQ19	[internal, integer]	000	Нормированный(0-1000) УП клапана рег. Уровня в барабане котла
LAB11CG001XQ20	[output, real]	000	УП клапана рег. Уровня в барабане котла
LAB11CG001ZU20	[output, integer]		
HLA11CG002XQ10	[input, integer]		
HLA11CG002XQ19	[internal, integer]		
HLA11CG002XQ20	[output, real]		
HLA11CG002ZU20	[output, integer]		
HNA11CG003XQ10	[input, integer]		
HNA11CG003XQ19	[internal, integer]		
HNA11CG003XQ20	[output, real]		
HNA11CG003ZU20	[output, integer]		
HHF11CG005XQ10	[input, integer]		
HHF11CG005XQ19	[internal, integer]		
HHF11CG005XQ20	[output, real]		

LAB11CG001XQ20 (\* УП клапана рег. Уровня в барабане котла) @0000 [output, real] Unit=проц.

Integer/Real Variable

Name: LAB11CG001XQ20 Network Address:

Comment: УП клапана рег. Уровня в барабане котла

Unit: % Conversion: [none]

Attributes

- ☐ Internal
- ☐ Input
- ☒ Output
- ☐ Constant

Format

- ☐ Integer [standard]
- ☒ Real

Initial value: 0

☐ Retain

Store Cancel Next Previous Extended

pos=8,30

# Привязка переменных к устройствам ввода-вывода

ISaGRAF - DEMO - Programs

File Make Tools Debug Options Help

Begin:

- sysm
- ccb01 CCB01EP001# ПОИ Компонент 1 Обработка данных дискретных датчиков
- cfb01 CFI
- cfb02 CFI
- cfc01 CFC
- Cste
- Cga
- cff01 CFF
- progTZ

Sequential:

- mainFGL
- fgu0
- fgu1
- fgu2

End:

- HHGAA40
- HHGAA40
- HHGAA40
- HHGAA40
- HHGAA40
- HHGAA40
- LABAA10
- HADAA10
- HLBAN00
- HNCAN00
- LABDL00
- LABAA20
- HHGDF0
- HHGAA20
- HNADP00
- HNAAA20
- HLADP00
- HLAAA20

Functions:

- isequal1
- isequal0
- bit\_norm
- combbyte

F.Blocks:

- lims\_boo

ISaGRAF - DEMO - I/O connection

File Edit Options Help

0	PB_DO16	
	cmd	~ →
	boo	~ →
1	PB_DO16	
	cmd	~ →
	boo	~ →
2		
3		
4	PB_DIN3	~ →
5	PB_DIN3	~ →
6		
7		
8	PB_V35A	~ →
9	PB_V35A	~ →
10	PB_PT100	~ →
11	PB_PT100	~ →
12		
13	PB_V35A	~ →
14		
15	ana_out	~ →
16	imst_out	~ →
17	ana_out	~ →
18	boo_out	~ →
19		
20	system	~ →

cfgname = cfgPB\_PT100\_61

1	HAD11CT003XQ11 (* Линейризованный сигнал Температура металла барабана котла *)
2	HAD11CT004XQ11 (* Линейризованный сигнал Температура пара за ГПЗ *)
3	HAC11CT002XQ11 (* Линейризованный сигнал Температура воды за экономайзером *)
4	

Connect I/O channel #1

Channel:

Free: LAB11CT001XQ11

Close

Connect

Free

Next

Previous

Begin: Csteam (Function Block Diagram)

# Конверсионные таблицы ISaGRAF

Таблица преобразования

Графическое представление функции преобразования

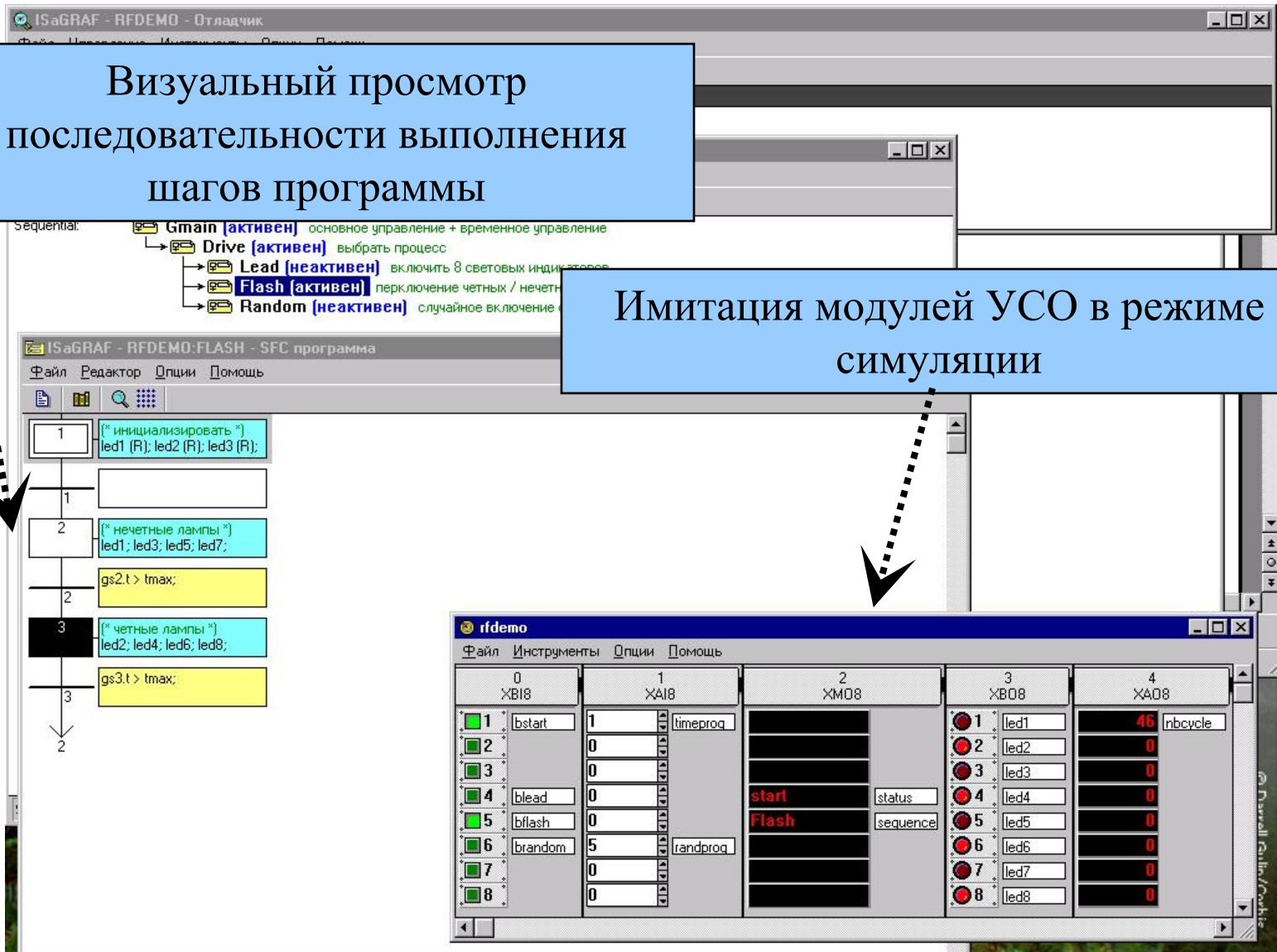
The screenshot shows the ISaGRAF - DEMO - Programs interface. The main window displays a project tree with a 'Begin:' section containing 'system', 'ccb01', 'cfb01', and 'cfb02'. A 'Sequential:' section contains a tree structure with 'progTZ', 'mainFGU', and several 'fgu' blocks. An 'End:' section contains a list of components like 'HHGAA402', 'LABAA101', etc. A 'Functions:' section lists 'isequal1', 'isequal0', 'bit\_norm', and 'combbyte'. The 'F.Blocks:' section contains 'lims\_boo'. A 'Conversion table 'TXK'' dialog is open, showing a list of points and a graph. The 'Points:' list contains two columns of values. The 'Electrical:' field is set to 16174 and the 'Physical:' field is set to 220. The 'Store' button is highlighted. A red line on the graph represents the conversion function.

Points:	
5441	80
5483	81
5595	95
5697	110
8342	120
9094	130
10237	145
11785	165
12964	180
14557	200
16174	220
17812	240
19470	260
21147	280
23691	310
26267	340
28868	370
31488	400
33241	420
50847	620
...	...

# Отладчик ISaGRAF

Визуальный просмотр  
последовательности выполнения  
шагов программы

Имитация модулей УСО в режиме  
симуляции





# Отладчик ISaGRAF

ISaGRAF - RFDEMO - Отладчик

Файл Управление Инструменты Опции Помощь

Работа

ISaGRAF - RFDEMO - Отладка программ

Файл Проект Инструменты Опции Помощь

Sequential:

- Gmain [активен] основное управление + временное управление
- Drive [активен] выбрать процесс
- Lead [неактивен] включить 8 световых индикаторов
- Flash [активен] переключение четных / нечетных
- Random [неактивен] случайное включение световых индикаторов

ISaGRAF - RFDEMO:FLASH - SFC программа

Файл Редактор Опции Помощь

ISaGRAF - RFDEMO - Глобальные булевские переменные

Файл Редактор Инструменты Опции Помощь

Булевские	Цели/Действительные	Таймеры	Сообщения	FB экземпляры	Макроопределения
Имя	Атриб.	Адр.	Значение		Комментарий
bflash	[вход]	0000	on		выбор анимации flash
blead	[вход]	0000	off		выбор анимации lead
brandom	[вход]	0000	off		выбор анимации random
bstart	[вход]	0000	running		команда start
led1	[выход]	B000	off		
led2	[выход]	B001	on		
led3	[выход]	B002	off		
led4	[выход]	B003	on		

brandom (\* выбор анимации random \*)  
@0000 [вход] (off,on)

Запись переменной булевская

переменная brandom

off on

Заблокир Разблокир Отказ

X008

led1 94 nbcycle

2 led2 0

3 led3 0

4 led4 0

5 led5 0

6 led6 0

7 led7 0

8 led8 0

Возможность интерактивной модификации значений переменных

# Отладчик ISaGRAF

The screenshot displays the ISaGRAF - RFDEMO - Отладчик (Debugger) interface. A dialog box titled 'Изменить временной цикл' (Change time cycle) is open, showing a field for 'Временной цикл (мсек):' (Time cycle (ms):) and buttons for 'Принять' (Accept) and 'Отказ' (Refuse). A dashed arrow points from the 'Отказ' button to a blue callout box.

The callout box contains the text: **Изменение цикла ISaGRAF в процессе отладки** (Changing the ISaGRAF cycle during debugging).

The background shows the 'Работа' (Work) area with a tree view containing 'Drive', 'Lead (неактивен)' (Lead (inactive)), 'Flash (активен)' (Flash (active)), and 'Random (неактивен)' (Random (inactive)). The 'Flash (активен)' item is selected, showing its description: 'перключение четных / нечетных' (switching even / odd).

The 'ISaGRAF - RFDEMO:FLASH - SFC программа' (ISaGRAF - RFDEMO:FLASH - SFC program) window is also visible, showing a sequence editor with steps 1, 2, and 3. Step 1 is '(\* инициализировать \*) led1 (R); led2 (R); led3 (R);'. Step 2 is '(\* нечетные лампы \*) led1; led3; led5; led7;' and 'gs2.t > tmax;'. Step 3 is '(\* четные лампы \*) led2; led4; led6; led8;' and 'gs3.t > tmax;'. The sequence is shown as a vertical flowchart with a loop back from step 3 to step 2.

The 'Watch' window at the bottom right shows the state of variables. The 'XBO8' register contains the value 211, and the 'XA08' register contains the value 0. The 'nbcycle' variable is also shown.

3	XBO8	4	XA08
1	led1	211	nbcycle
2	led2	0	
3	led3	0	
4	led4	0	
5	led5	0	
6	led6	0	
7	led7	0	
8	led8	0	

## **Итак, что же может и что поддерживает ISaGRAF:**

- стандартные языки программирования (стандарт IEC 61131-3);
- открытость как инструментальных средств, так и внутренних структур данных прикладной задачи;
- многофункциональный графический отладчик;
- стандартный человеко-машинный интерфейс (HMI);
- возможность связи с любой системой визуализации и управления данными (SCADA-системы).

Такой обширный набор инструментов позволяет программировать конфигурации оборудования любой сложности и получить качественный результат стандартными средствами с минимальными издержками.