

УТВЕРЖДЕН

ДАРЦ.70027-08 34 01-1-ЛУ

TECON - TECHNICS ON!®

ЦЕЛЕВАЯ ЗАДАЧА ISaGRAF ДЛЯ МНОГОФУНКЦИОНАЛЬНЫХ КОНТРОЛЛЕРОВ

Руководство оператора

ДАРЦ.70027-08 34 01-1

Листов 59



© ЗАО ПК «Промконтроллер», 2009-2010

При перепечатке ссылка на ЗАО ПК «Промконтроллер» обязательна.

TECON – TECHNICS ON![®], ТЕКОНИК[®], TeNIX[®] – зарегистрированные товарные знаки ЗАО "Текон-Груп".

IBM, PC –зарегистрированные товарные знаки IBM Corp.

ISaGRAF – зарегистрированный товарный знак ICS Triplex Inc.

Все другие названия продукции и другие имена компаний использованы здесь лишь для идентификации и могут быть товарными знаками или зарегистрированными товарными знаками их соответствующих владельцев. ЗАО ПК «Промконтроллер» не претендует ни на какие права, затрагивающие эти знаки.

Изготовитель оставляет за собой право вносить изменения в программное обеспечение, улучшающие характеристики изделия.

Юридический адрес:

ул. Красноказарменная, д.12, стр.9,

Москва, 111250, Россия,

ЗАО ПК «Промконтроллер»

тел.: +7 (495) 7304112

факс: +7 (495) 7304113

e-mail: support@tecon.ru

[http:// www.tecon.ru](http://www.tecon.ru)

Почтовый адрес:

ул. Б.Семеновская, д.40, стр.18,

Москва, 107023, Россия,

ЗАО ПК «Промконтроллер»

тел.: +7 (495) 7304112

факс: +7 (495) 7304113

e-mail: support@tecon.ru

[http:// www.tecon.ru](http://www.tecon.ru)

v 1.2.4 /22.06.10

СОДЕРЖАНИЕ

1	СОЗДАНИЕ ПРИКЛАДНОГО ПРОЕКТА	5
2	ОПРЕДЕЛЕНИЕ КОНТРОЛЛЕРА В СРЕДЕ РАЗРАБОТКИ ПРОЕКТА.....	8
2.1	Определение контроллера с использованием файлов переноса	8
2.2	Определение контроллера с использованием TDB-файла	9
3	ИСПОЛЬЗОВАНИЕ СПЕЦИАЛЬНЫХ ТИПОВ ДАННЫХ.....	11
4	ИСПОЛЬЗОВАНИЕ ДРАЙВЕРОВ МОДУЛЕЙ ВВОДА-ВЫВОДА.....	13
5	ИСПОЛЬЗОВАНИЕ ДРАЙВЕРОВ MODBUS.....	17
6	ИСПОЛЬЗОВАНИЕ СПЕЦИАЛЬНЫХ ФУНКЦИОНАЛЬНЫХ БЛОКОВ.....	21
6.1	Функциональный блок T_V04	22
6.2	Функциональный блок T_REBOOT	23
6.3	Функциональный блок T_WDOG	24
6.4	Функциональный блок T_DIAGSTAT.....	25
6.5	Функциональный блок T_DIAGMOD	26
6.6	Функциональный блок T_DIAGUSER.....	28
6.7	Функциональный блок T_INFO	29
6.8	Функциональный блок T_LOG	30
6.9	Функциональный блок T_CLOCK	31
7	ЗАГРУЗКА И ОТЛАДКА ПРИКЛАДНОГО ПРОЕКТА.....	33
8	ОСОБЕННОСТИ ПРИМЕНЕНИЯ ПРИКЛАДНОГО ПРОЕКТА В РЕЖИМАХ РЕЗЕРВИРОВАНИЯ КОНТРОЛЛЕРА	35
8.1	Ограничения на прикладные проекты	35
8.2	Создание прикладного проекта.....	36
8.3	Загрузка прикладного проекта	40
8.4	Отладка прикладного проекта.....	41
8.5	Специальный режим изменения прикладного проекта	42
9	СПИСОК ЛИТЕРАТУРЫ.....	44
	Приложение А (обязательное) Определение контроллеров МФК и ТКМ52.....	45
	Приложение Б (обязательное) Перечень драйверов модулей ввода-вывода МФК и ТКМ52	46
	Приложение В (обязательное) Перечень драйверов модулей ТЕКОНИК	47
	Приложение Г (обязательное) Определение контроллера МФК3000	48
	Приложение Д (обязательное) Перечень драйверов модулей ввода-вывода МФК3000	49
	Приложение Е (обязательное) Определение контроллера МФК1500.....	51
	Приложение Ж (обязательное) Перечень драйверов модулей ввода-вывода МФК1500.....	52
	Приложение И (обязательное) Драйвер ввода-вывода панели оператора V04M.....	54
	Приложение К (обязательное) Определение драйверов системы Modbus	56
	Приложение Л (обязательное) Драйверы системы Modbus.....	57

Целевая задача ISaGRAF для многофункциональных контроллеров разработана на базе ядра ISaGRAF v.5.01 и предназначена для исполнения в контроллерах МФК3000 [3], МФК1500 [4] с процессорным модулем P05 (как в одиночном, так и в резервированном применении) прикладной технологической программы, разработанной в среде ISaGRAF 5 Workbench.

Целевая задача ISaGRAF PRO для многофункциональных контроллеров разработана на базе ядра ISaGRAF PRO v.4.12 и предназначена для исполнения в контроллерах МФК [1, 2] (как в одиночном, так и в резервированном применении) и ТКМ52 [5] прикладной технологической программы, разработанной в среде ISaGRAF PRO Workbench или ISaGRAF 5 Workbench.

Далее в документе обе целевые задачи обозначаются как целевая задача ISaGRAF, среды разработки ISaGRAF 5 Workbench и ISaGRAF PRO Workbench как ISaGRAF Workbench.

Исходными данными для исполнения прикладного проекта служат входные переменные системы ввода/вывода и внутренние переменные (в терминах системы ISaGRAF), значения которых формируются алгоритмами прикладного проекта или приложением (приложениями) системы «верхнего» уровня АСУ ТП (СВУ). Результатом исполнения такой программы является формирование новых значений выходных переменных системы ввода/вывода и внутренних переменных прикладного проекта, доступных для приложения (приложений) СВУ.

Дополнительно целевая задача ISaGRAF, реализованная в рамках программного обеспечения (далее ПО) контроллера, выполняет следующие функции:

- хранение (в директории на Flash Disk) копии кода прикладного проекта и автоматический запуск на исполнение сохраненного кода прикладного проекта в случае последующего перезапуска контроллера и инициализации целевой задачи ISaGRAF;
- хранение в энергонезависимой памяти значений переменных прикладной программы с атрибутом *retain* (хранимые) и восстановления этих значений в случае последующего перезапуска контроллера и инициализации целевой задачи ISaGRAF;
- хранение (в директории на Flash Disk) файлов ресурсов (например, символьной таблицы прикладного проекта);
- обмен «технологическими» данными (значениями переменных прикладного проекта) с приложениями СВУ и другими контроллерами по сети Ethernet.

Разработка прикладного проекта для контроллера проводится с использованием среды ISaGRAF Workbench.

	<p>ИНФОРМАЦИЯ</p> <p>Везде, где вы увидите этот информационный знак, обратите внимание на важную, выделенную информацию.</p>
	<p>ВНИМАНИЕ</p> <p>Везде, где вы увидите этот предупреждающий знак, строго следуйте инструкциям во избежание повреждения оборудования.</p>

1 СОЗДАНИЕ ПРИКЛАДНОГО ПРОЕКТА

Процедура создания прикладного проекта в среде ISaGRAF Workbench предусматривает последовательное выполнение ряда действий, описанных далее по тексту.

1.1 Запустите на выполнение среду разработки прикладных проектов ISaGRAF Workbench. Выберите шаблон и создайте новый проект (см. рисунок 1.1).

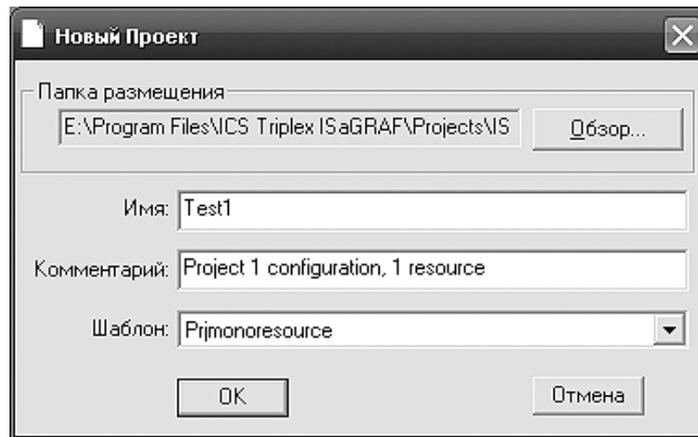


Рисунок 1.1 – Форма создания нового проекта

1.2 Если в среде ISaGRAF Workbench ранее не было произведено определение контроллера, выполните его, как указано в разделе 2.

1.3 В окне свойств ресурса определите в качестве используемой целевой системы **MFC**, укажите режим генерации платформо-независимого кода прикладного проекта (TIC кода). Указание встроенной таблицы идентификаторов (см. рисунок 1.2) является обязательным.

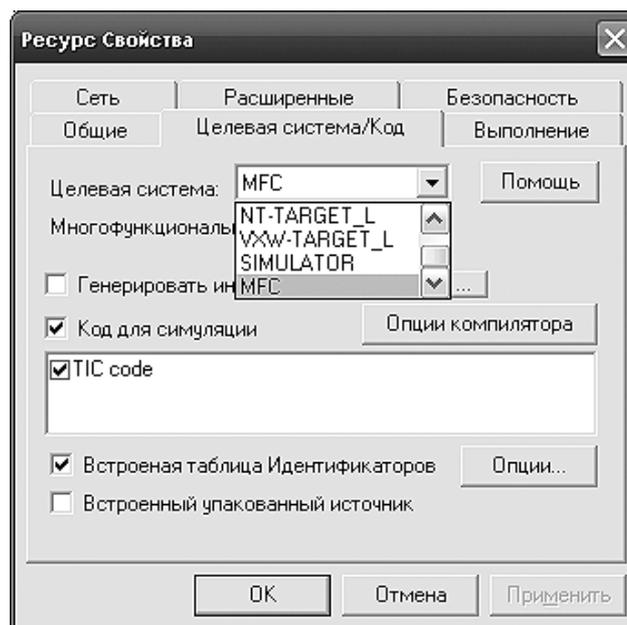


Рисунок 1.2 – Форма свойств ресурса

1.4 Нажав кнопку «Опции», укажите полную таблицу (см. рисунок 1.3).

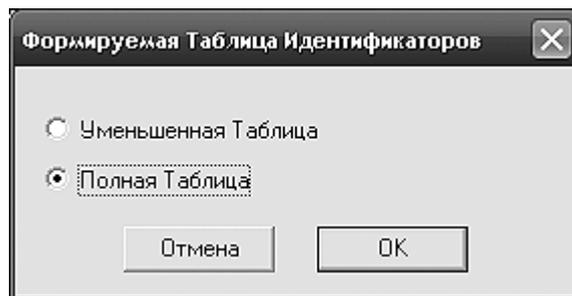


Рисунок 1.3 – Форма использования символьной таблицы ресурса

1.5 При использовании в прикладном проекте переменных с атрибутом *retain* (хранимая), в окне свойств ресурса укажите путь размещения в памяти контроллера области хранимых переменных (см. рисунок 1.4).

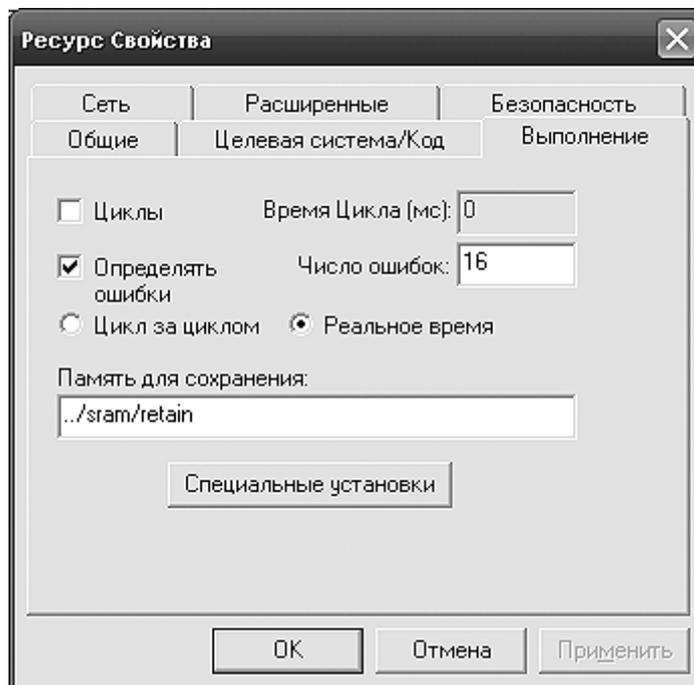


Рисунок 1.4 – Размещение области хранимых переменных

	<p>ВНИМАНИЕ</p> <p>Память для сохранения переменных с атрибутом <i>retain</i> (хранимая):</p> <p>..../sram/retain</p> <p>В многоресурсном проекте память для сохранения переменных с атрибутом <i>retain</i> (хранимая) должна быть разная!</p> <p>Например:</p> <p>..../sram/retainN</p> <p>где N – номер ресурса.</p>
---	--

1.6 Далее проводите разработку прикладного проекта с использованием функциональных возможностей, предоставляемых средой ISaGRAF Workbench. При разработке используйте следующие компоненты, включенные в среду разработки при определении контроллера:

- специальные типы данных (см. раздел 3);
- драйверы модулей УСО контроллера (см. раздел 4);
- специальные функциональные блоки (см. раздел 6).

2 ОПРЕДЕЛЕНИЕ КОНТРОЛЛЕРА В СРЕДЕ РАЗРАБОТКИ ПРОЕКТА

Перед началом разработки прикладных проектов необходимо произвести стандартную процедуру определения контроллера (определения конфигурации целевой задачи контроллера) в среде ISaGRAF Workbench.

В указанной конфигурации описываются следующие компоненты целевой задачи:

- ядро целевой задачи (секция Targets);
- типы данных (секция Data types);
- драйверы модулей УСО (секция I/O devices);
- функциональные блоки (секция «С» functions and function blocks);
- сетевые интерфейсы (секция Networks).

Исходные файлы конфигурации целевой задачи контроллеров МФК (ТКМ52), МФК3000 и МФК1500 поставляются Пользователю на компакт-диске [14] в каталоге «Целевая задача для многофункциональных контроллеров».

	<p>ИНФОРМАЦИЯ</p> <p>Для контроллеров ТКМ52 используются файлы определения и драйверы ввода-вывода контроллера МФК.</p>
--	--

Определение целевой задачи контроллера в среде ISaGRAF Workbench может быть реализовано одним из двух способов (пп. 2.1, 2.2).

При поставке целевой задачи ISaGRAF с библиотекой алгоритмов TIL PRO Std необходимо провести дополнительное определение контроллера (обновление), как указано в [12].

При поставке целевой задачи ISaGRAF с библиотекой алгоритмов TIL PRO Com необходимо провести дополнительное определение контроллера (обновление), как указано выше, используя файл определения **tilcom.txt** [13] с поставляемого компакт-диска.

2.1 ОПРЕДЕЛЕНИЕ КОНТРОЛЛЕРА С ИСПОЛЬЗОВАНИЕМ ФАЙЛОВ ПЕРЕНОСА

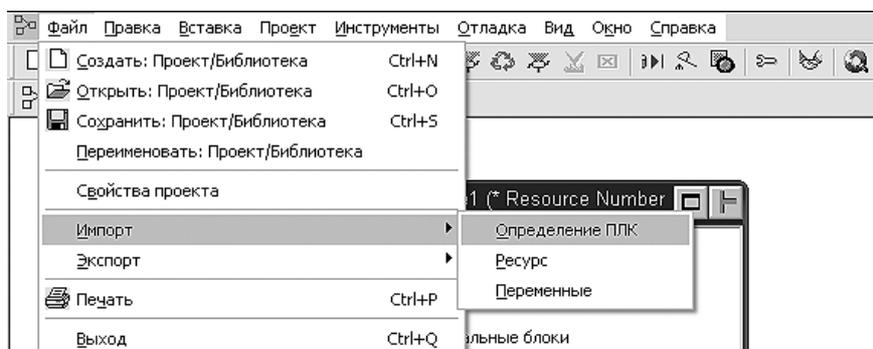


Рисунок 2.1 – Импортное определение контроллера

2.1.1 В окне открытого проекта среды ISaGRAF Workbench из пункта меню *Файл→Импорт→Определение ПЛК* запустить функцию импорта на выполнение (см. рисунок 2.1).

2.1.2 В появившемся окне выбора файлов выбрать исходный файл переноса определения контроллера (**mfc_target.txt**), расположенный в дисковом пространстве инструментальной станции (при необходимости вставить поставочный диск в дисковод).

2.1.3 Дождаться сообщения об окончании импорта определения контроллера (см. рисунок 2.2).



Рисунок 2.2 –Сообщение об окончании импорта определения ПЛК

2.1.4 Произвести обновление используемой целевой системы *MFC* необходимыми драйверами устройств ввода-вывода (выполните пп. 2.1.1-2.1.3 для соответствующего файла определения):

- драйвер V04M в исполнении с каналами ввода-вывода (каталог CD [14] *V04M/v04m_driver.txt*);
- драйверы модулей ТЕКОНИК (*ТЕКОНИК/teconic_drivers.txt*);
- драйверы модулей базовой системы ввода-вывода контроллера МФК и ТКМ52 (*МФК/mfc_drivers.txt*);
- драйверы модулей ввода-вывода контроллера МФК3000 (*МФК3000/mfc3000_drivers.txt*);
- драйверы модулей ввода-вывода контроллера МФК1500 (*МФК1500/mfc1500_drivers.txt*);
- драйверы системы Modbus (*Modbus/Modbus_drivers.txt*) (описание см. приложение Л).

	<p>ВНИМАНИЕ</p> <p>Не применяйте драйверы модулей ТЕКОНИК на контроллерах МФК3000 и МФК1500!</p>
--	---

2.2 ОПРЕДЕЛЕНИЕ КОНТРОЛЛЕРА С ИСПОЛЬЗОВАНИЕМ TDB-ФАЙЛА

2.2.1 Запустить программу *TDBuild.exe*, расположенную в директории *Bin* каталога установки среды ISaGRAF Workbench.

2.2.2 Из пункта меню *File/Open* открыть исходный файл определения контроллера:

- **mfc3000.tdb** для контроллера МФК3000;
- **mfc1500.tdb** для контроллера МФК1500;

- **mfc.tdb** для контроллеров МФК или ТКМ52.

TDB-файлы располагаются на поставочном диске [14].

2.2.3 Выделить все доступные компоненты целевой задачи ISaGRAF.

2.2.4 Из пункта меню *Build/Send to file* запустить на выполнение функцию создания файла переноса определения контроллера. Следовать указаниям на экране монитора. По ходу выполнения указать полное переопределение целевой задачи ISaGRAF.

2.2.5 Дождаться сообщения о создании файла переноса определения контроллера. Выйти из программы *TDBuild.exe*. Провести все операции, описанные в первом способе.

Определение целевой задачи контроллера в среде ISaGRAF Workbench, реализованное вторым способом, позволяет:

- выбрать необходимые для переноса в среду ISaGRAF Workbench компоненты целевой задачи;
- перенести справочную информацию о каждом из выбранных компонентов конфигурации целевой задачи.

3 ИСПОЛЬЗОВАНИЕ СПЕЦИАЛЬНЫХ ТИПОВ ДАННЫХ

При определении контроллера (см. раздел 2) производится установка в среде ISaGRAF Workbench специальных типов данных. Перечень специальных типов, необходимых для назначения переменных прикладного проекта с направлениями Input и Output, приводится в приложениях А, Г и Е.

При проведении монтажа (в терминах ISaGRAF) каналов ряда драйверов (см. приложения Б, В, Д, Ж) к переменным прикладного проекта, следует использовать переменные следующих типов:

VBOOL – структура, включающая поля:

- Value (тип BOOL);
- Valid (тип DINT);

VDINT – структура, включающая поля:

- Value (тип DINT);
- Valid (тип DINT);

VREAL – структура, включающая поля:

- Value (тип REAL);
- Valid (тип DINT);

где поля *Value* – значение сигнала указанной размерности,
Valid – оценка достоверности сигнала.

Поле *Valid* может иметь следующие значения:

- 0 – сигнал в канале модуля ввода-вывода достоверен (нормальная работа);
- 1 – отказ канала модуля ввода-вывода (при общем отказе модуля аналогичное значение будет установлено для всех каналов модуля);
- 2 – превышение верхней границы диапазона измерения модуля (в относительных единицах);
- 3 – превышение нижней границы диапазона измерения модуля (в относительных единицах);
- 4 – недостоверность сигнала, связанная с неготовностью канала модуля (всего модуля ввода-вывода) к проведению измерений (рестарт встроенного ПО, неисправен канал компенсации холодного спая при измерении сигнала термопар в модулях L16, L16i);
- 5 – в канал модуля вывода не было произведено ни одной записи с момента запуска (перезапуска) контроллера;
- 6 – обрыв линии связи канала модуля ввода (L16i, F24, T3501, все модули МФК3000) с датчиком;
- 7 – замыкание линии канала (DO16, DO32 только МФК3000);
- 8 – значение канала выходит за границы аварийной уставки (LIG16, LIG8, LIG4, LI16, AI16, AIG16, AIG8, ADO24);

- 9 – значение канала выходит за границы предупредительной уставки (LIG16, LIG8, LIG4, LI16, AI16, AIG16, AIG8, ADO24);
- 10 – недостоверное значение (ошибка связи с АЦП) (LIG16, LIG8, LIG4, LI16, AI16, AIG16, AIG8, ADO24);
- 11 – неисправен канал компенсации холодного спая (LIG16, LIG8, LIG4, LI16);
- 12 – неисправность узла питания канала (LI16);
- 13 – неготовность данных на канале (AI16, LIG16, LIG8, LIG4, LI16, FP6, AIG16, AIG8, ADO24 и т.д.);
- 255 – другая не определенная выше ошибка измерения.

Пример использования специальных типов данных приводится в разделе 4.

4 ИСПОЛЬЗОВАНИЕ ДРАЙВЕРОВ МОДУЛЕЙ ВВОДА-ВЫВОДА

При определении контроллера в среде ISaGRAF Workbench (см. раздел 2) производится установка драйверов модулей (устройств) ввода/вывода. Перечень драйверов модулей УСО контроллера МФК (ТКМ52) и системы ТЕКОНИК приводится в приложении Б и В, контроллера МФК3000 – в приложении Г, контроллера МФК1500 – в приложении Ж.

При разработке прикладного проекта в окне выбора устройств для каждого используемого в контроллере модуля УСО следует указать имя драйвера (см. рисунок 4.1), и параметр *HwId* (см. рисунок 4.2), определяющий адрес модуля ввода-вывода:

- для модулей УСО контроллера - адрес модуля в контроллере;
- для модулей системы ТЕКОНИК – сетевой адрес модуля.

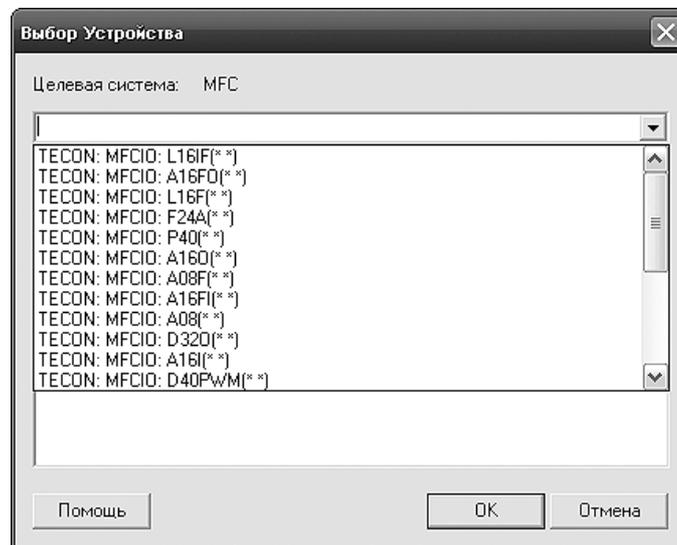


Рисунок 4.1 – Форма выбора драйвера ввода-вывода

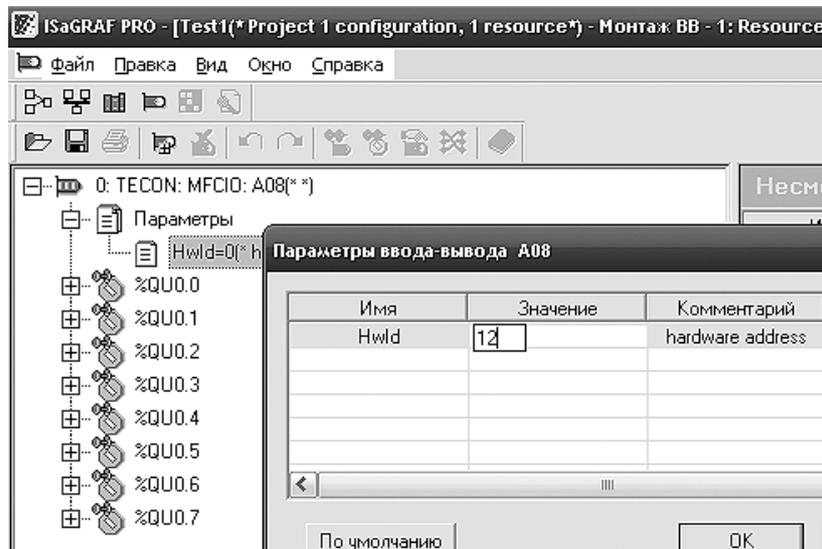


Рисунок 4.2 – Установка адреса драйвера ввода-вывода

	<p>ИНФОРМАЦИЯ</p> <p>Для модулей ввода-вывода контроллера МФК3000 или МФК1500, сконфигурированных для работы в составе резервированной группы, указывается ОДИН драйвер на группу, в параметр <i>Hwld</i> драйвера записывается значение адреса модуля, назначенного средствами СПО для работы в режиме MASTER по умолчанию.</p>
---	---

При привязке каналов драйверов модулей УСО к переменным проекта необходимо учитывать то, что переменная каждого из каналов драйвера имеет тип «структура» (см. раздел 3).

Пример:

При создании в словаре среды ISaGRAF Workbench входной переменной необходимо (см. рисунок 4.3):

- указать имя переменной (*ch1*), которая будет привязана к каналу (0) модуля УСО (*A16/0*), который имеет системный адрес (посадочное место в крейте контроллера – 0);
- присвоить переменной (*ch1*) тип VDINT;
- назначить переменной (*ch1*) направление Input и атрибут Read.

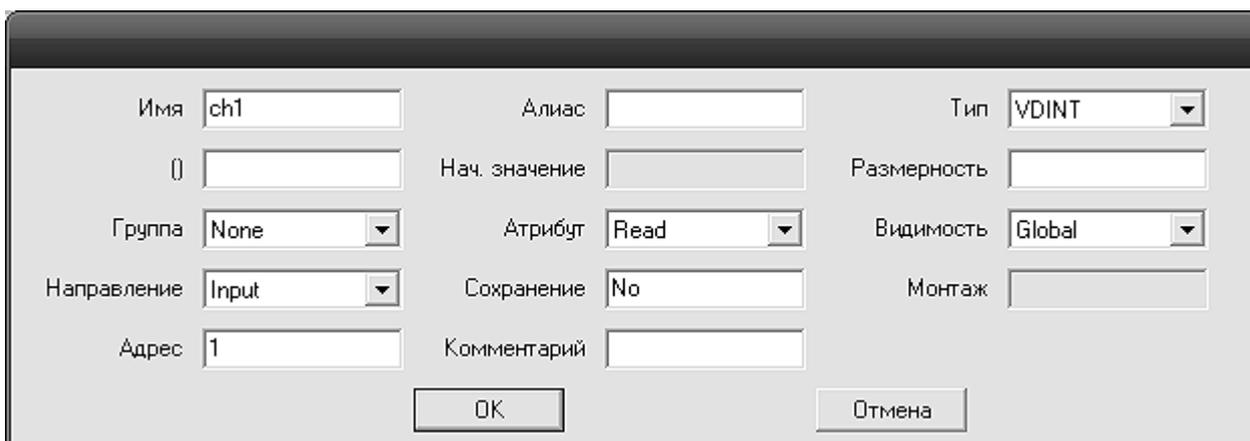


Рисунок 4.3 – Создание переменной типа Input

В словаре переменная (*ch1*) будет представляться структурой, состоящей из двух переменных (*ch1.Value*) типа DINT и (*ch1.Valid*) типа DINT (см. рисунок 4.4).

Глобальные Переменные										
Имя	Алиас	Тип	Атрибут	Видимо...	Направ...	Сохранение	Монтаж	Адрес	Комментар...	
[-] ch1		VDINT	Read	Global	Input	No	%U1.0.0	1		
[-] ch1.Value		DINT	Read	Global	Input	No	%U1.0.0	1	DINT value	
[-] ch1.Valid		DINT	Read	Global	Input	No	%U1.0.0	1	value's validity	
...										

Рисунок 4.4 – Представление переменной типа Input в словаре

При использовании в редакторах программ ISaGRAF Workbench переменная (*ch1*) раскладывается на *.Value (DINT) и *.Valid (DINT) (см. рисунок 4.5).

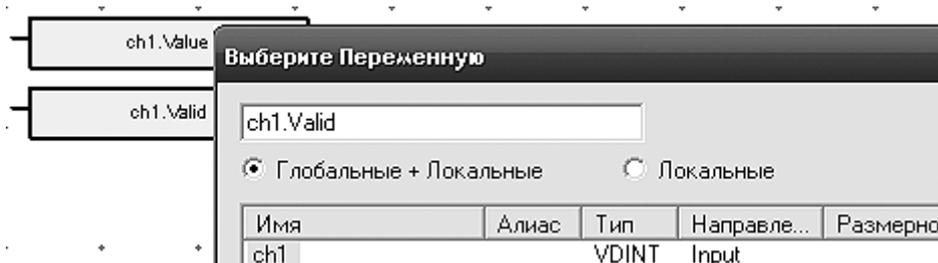


Рисунок 4.5 – Представление переменной типа Input в редакторе программ

Применение некоторых модулей ввода-вывода имеют ряд особенностей (все настройки каналов модулей производятся в программе TUNER):

4.1 В модулях дискретного вывода контроллеров МФК1500 и МФК3000 часть каналов могут быть переведены в режим «канальных пар» для реверсивного управления объектами.

Драйверы DOXX используются для записи дискретных сигналов.

Для управления канальными парами используются драйверы DOXXPWM. При записи в канальную пару длительности импульса отрицательного значения (в миллисекундах) задействуется нечетный канал пары, положительного – четный канал. При этом соответствующий канал выдает импульс заданной длительности.

Допускается одновременное использование драйвера DOXX и DOXXPWM для одного физического модуля. Дискретные каналы в драйверах DOXXPWM будут заблокированы и наоборот, и в драйверах DOXX будут заблокированы каналы пар.

4.2 В модулях дискретного ввода контроллеров МФК1500 и МФК3000 первые 16 каналов ввода могут быть настроены на чтение значений 16-разрядных или 32-разрядных счетчиков.

Для чтения дискретных значений используются драйверы DIXX.

Для чтения значений 16-разрядных счетчиков используются драйверы DIXXCNT. Драйвер имеет 32 канала, из которых первые 16 каналов – значения счетчиков импульсов, каналы, начиная с 16-го – значения времени, за которое подсчитаны импульсы канала. Соответствие каналов: в 0-м канале значение счетчиков, в 16-ом – значение таймера в миллисекундах; 1-17...15-31.

Для чтения значений 32-разрядных счетчиков используются драйверы DIXXCNT2. Драйвер имеет 16 каналов.

Допускается одновременное использование драйверов DIXX, DIXXCNT, DIXXCNT2. При этом драйвер DIXX получает дискретные значения всех каналов, а драйверы счетчиков получают значения только заказанных в конфигурации каналов. Допустимое количество каналов со счетчиками рассчитываются согласно [3].

4.3 В модулях T3601, T3602 и D40 часть каналов могут быть переведены в режим формирования импульсов, заданной длительности (ШИМ).

Драйверы T360X и D40 используются для записи дискретных сигналов.

Драйверы T360XPWM и D40PWM используются для записи в каналы, сконфигурированные на работу в режиме ШИМ. В канал ШИМ D40PWM записываются значения импульса в миллисекундах, в канал ШИМ T360XPWM записываются значения импульса в десятках миллисекунд.

Допускается одновременное использование основного драйвера и PWM драйвера для одного физического модуля. Дискретные каналы в драйверах PWM будут заблокированы и наоборот, и в основных драйверах каналы ШИМ будут также заблокированы.

4.4 В модулях **T3702** и **F24** часть каналов могут быть переведены в режим счетчиков импульсов.

Для чтения дискретных значений каналов используются драйверы T3702 и F24D.

Для получения информации счетчиков импульсов используются соответственно T3702CNT и F24A.

Допустимо одновременное использование основных драйверов и драйверов счетчиков импульсов. Драйверы T3702 и F24D получают дискретные значения всех каналов, а T3702CNT и F24A значения счетчиков импульсов только сконфигурированных каналов.

5 ИСПОЛЬЗОВАНИЕ ДРАЙВЕРОВ MODBUS

При определении контроллера в среде ISaGRAF Workbench (см. раздел 2, приложение К) производится установка драйверов системы Modbus. Перечень драйверов приводится в приложении Л.

При разработке прикладного проекта в окне выбора устройств для каждого используемого в контроллере драйвера Modbus следует указать:

- имя драйвера (см. рисунок 4.1);
- требуемое количество каналов (см. рисунок 5.1);
- параметры *ClientId*, *SlaveId*, *BlockId* для комплексных (сложных) драйверов MB_MASTER_* (см. рисунок 5.2), определяющие соответственно:

ClientId (STRING(16)) – IP адрес (с подчеркиванием вместо точек) в режиме TCP или порт (COM1-COMn) в режиме RTU/ASCII;

SlaveId (DINT) – адрес устройства;

BlockId (DINT) – номер блока в устройстве (начальный адрес диапазона Modbus регистров).

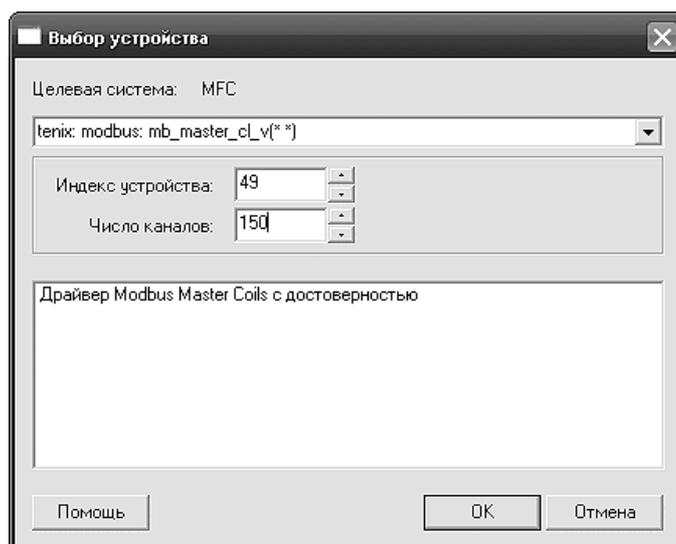


Рисунок 5.1 – Определение количества каналов драйвера Modbus

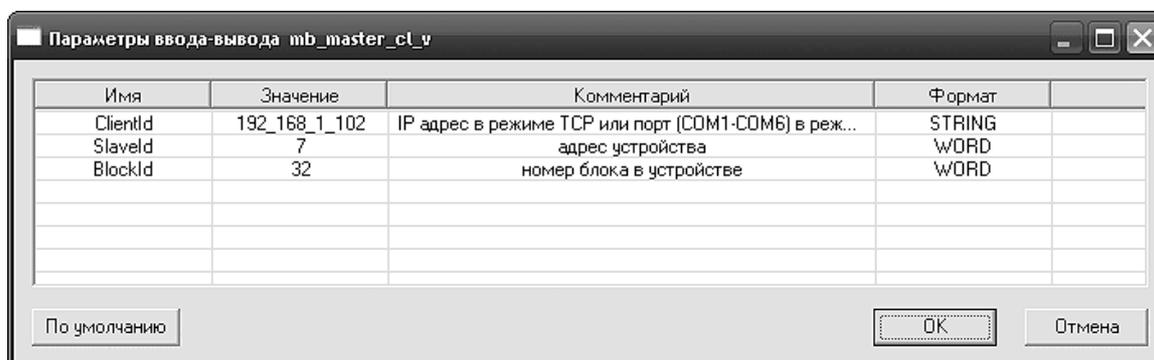


Рисунок 5.2 – Определение параметров комплексных (сложных) драйверов MB_MASTER_*

Для случая, когда контроллер выступает в качестве Modbus RTU/ASCII Slave (TCP Server) необходимо подключить драйвер MB_SLAVE_* соответствующего типа и привязать к каналам необходимые переменные, либо создать промежуточные переменные, через которые осуществлять обмен необходимыми данными.

Для случая, когда контроллер выступает в качестве Modbus RTU/ASCII Master (TCP Client) необходимо подключить драйвер MB_MASTER_* соответствующего типа и привязать к каналам необходимые переменные, либо создать промежуточные переменные, через которые осуществлять обмен необходимыми данными.

Описание драйверов системы Modbus

Комплексный драйвер MB_SLAVE_CL - для ввода-вывода дискретных данных. Данные, прочитанные или записанные в компоненты этого комплексного драйвера, доступны для опрашивающих контроллер устройств как регистры Modbus типа Coils (чтение и запись дискретных данных). Каждому каналу устройств соответствует один регистр (один бит). Состоит из следующих простых драйверов:

- MB_SLAVE_CL_IN (IN, BOOL) - чтение Coils-регистров, до 32000 каналов;
- MB_SLAVE_CL_OUT (OUT, BOOL) - запись Coils-регистров, до 32000 каналов.

Простой драйвер MB_SLAVE_DI (OUT, BOOL, до 32000 каналов) - для вывода дискретных данных. Данные, записываемые в этот драйвер, будут выглядеть как битовые регистры Modbus Discrete Inputs для опрашивающих контроллер устройств.

Комплексный драйвер MB_SLAVE_HR_INT - для ввода-вывода целых чисел размером 16 бит. Данные, прочитанные или записанные в компоненты этого комплексного драйвера, доступны для опрашивающих контроллер устройств как регистры Modbus типа Holding Registers (чтение и запись целых 16-битных данных). Каждому каналу устройств соответствует один регистр (16 бит). Несмотря на то, что размерность элементов DINT (32 бита), для ввода-вывода используются только младшие 16 бит каждого канала. Драйвер состоит из следующих простых драйверов:

- MB_SLAVE_HR_INT_IN (IN, DINT) - чтение Holding Register, до 10000 каналов;
- MB_SLAVE_HR_INT_OUT (OUT, DINT) - запись Holding Register, до 10000 каналов.

Простой драйвер MB_SLAVE_IR_INT (OUT, DINT, до 10000 каналов) - для вывода целых чисел размером 16 бит. Данные, записанные в этот драйвер, доступны для опрашивающих контроллер устройств как регистры Modbus Input Registers (только чтение), причем каждому каналу устройства будет соответствовать один регистр Modbus. Несмотря на то, что тип канала DINT, будут использоваться только младшие 16 бит записанных данных.

Комплексный драйвер MB_SLAVE_HR_DINT - для ввода-вывода целых чисел размером 32 бита. Аналогичен драйверу MB_SLAVE_HR_INT, отличие лишь в том, что каждый канал драйвера отображается в два регистра (первый регистр содержит старшие 16 бит, второй - младшие 16 бит). Драйвер состоит из следующих простых драйверов:

- MB_SLAVE_HR_DINT_IN (IN, DINT) - чтение Holding Register, до 5000 каналов;
- MB_SLAVE_HR_DINT_OUT (OUT, DINT) - запись Holding Register, до 5000 каналов.

Простой драйвер MB_SLAVE_IR_DINT (OUT, DINT, до 5000 каналов) - для вывода целых чисел размером 32 бита. Аналогичен драйверу MB_SLAVE_IR_INT, только каждому каналу соответствует два регистра Modbus.

Комплексный драйвер MB_SLAVE_HR_REAL - для ввода-вывода чисел с плавающей точкой размером 32 бита. Аналогичен MB_SLAVE_HR_DINT, только передает не целые числа, а числа с плавающей точкой в формате IEEE 754. Также, одному каналу устройств соответствует два регистра Modbus Holding Registers, первый содержит старшие 16 бит, второй - младшие 16 бит. Драйвер состоит из следующих простых драйверов:

- MB_SLAVE_HR_REAL_IN (IN, REAL, до 5000 каналов) - чтение Holding Registers;
- MB_SLAVE_HR_REAL_OUT (OUT, REAL, до 5000 каналов) - запись Holding Registers.

Простой драйвер MB_SLAVE_IR_REAL (OUT, REAL, до 5000 каналов) - для вывода чисел с плавающей точкой размером 32 бита в формате IEEE 754. Аналогичен MB_SLAVE_IR_DINT.

Комплексный драйвер MB_MASTER_CL - для чтения и записи регистров Modbus Coils удаленных опрашиваемых устройств, в соответствии с картой памяти устройства. Драйвер состоит из следующих простых драйверов:

- MB_MASTER_CL_IN (IN, BOOL) - чтение Coils, до 65535 каналов;
- MB_MASTER_CL_OUT (OUT, BOOL) - запись Coils, до 65535 каналов.

Простой драйвер MB_MASTER_DI (IN, BOOL) - для чтения регистров Modbus Discrete Inputs удаленных устройств, сконфигурированных в карте памяти устройства.

Комплексный драйвер MB_MASTER_HR_INT - для чтения и записи регистров Modbus Holding Register, представленных в виде целых чисел. В соответствии с картой памяти, каждому каналу может сопоставляться один или два регистра Modbus. Драйвер состоит из следующих простых драйверов:

- MB_MASTER_HR_INT_IN (IN, DINT) - чтение Holding Registers, до 125 каналов;
- MB_MASTER_HR_INT_OUT (OUT, DINT) - запись Holding Registers, до 125 каналов.

Простой драйвер MB_MASTER_IR (IN, DINT) - для чтения регистров Modbus Input Registers удаленных устройств. В зависимости от конфигурации карты памяти, каждому каналу может сопоставляться один или два регистра Modbus.

Комплексный драйвер MB_MASTER_HR_REAL - для чтения и записи регистров Modbus Holding Registers, представленных в карте памяти удаленного устройства в виде чисел с плавающей точкой:

- MB_MASTER_HR_REAL_IN (IN, REAL) - чтение чисел с плавающей точкой;
- MB_MASTER_HR_REAL_OUT (OUT, REAL) - запись чисел с плавающей точкой.

Простой драйвер MB_MASTER_IR_REAL - для чтения регистров Modbus Input Registers удаленных устройств, представленных в карте памяти удаленного устройства в виде чисел с плавающей точкой.

ИНФОРМАЦИЯ

В случае использования комплексных драйверов MB_MASTER_* через выходные каналы производится запись регистров, а через входные каналы, чтение тех же самых регистров. При этом запись в выходной канал осуществляется по изменению, а во входном канале находится последнее считанное с устройства значение.

В случае ошибки записи в устройство значение переменной, привязанной к выходному каналу, не сбрасывается автоматически в предыдущее состояние. Таким образом, на последующих циклах программы ввиду отсутствия изменения, не производится повторной попытки записи в устройство.

Один из вариантов избежать этого – замкнуть входной канал на выходной по истечению некоторого таймаута (величина таймаута зависит от устройства и от особенностей объекта управления).



В случае использования комплексных драйверов MB_SLAVE_* чтение команд на запись, отправленных мастером сети, производится во входных каналах драйвера. При этом трансляция полученных значений в выходные каналы драйвера автоматически не производится. Для обеспечения трансляции полученных значений в выходные каналы драйвера необходимо в прикладной программе ISaGRAF обеспечить присвоение значений входных каналов драйвера выходным.

Эти действия, так же необходимо выполнить для проверки прохождения команд на запись с использованием программы TUNER [1-5].

Так же на уровне пользовательской программы, разрешая или запрещая трансляцию значений входных каналов драйвера в выходные, можно обеспечить блокировку полученных команд мастера сети, например, в случае недостоверности или некорректности полученных значений.

6 ИСПОЛЬЗОВАНИЕ СПЕЦИАЛЬНЫХ ФУНКЦИОНАЛЬНЫХ БЛОКОВ

Для работы программного обеспечения контроллера в составе целевой задачи ISaGRAF разработаны специальные функциональные блоки, которые можно использовать в редакторах языков IEC 61131-3 при разработке прикладного проекта. Установка специальных функциональных блоков в среде разработки ISaGRAF Workbench производится при определении контроллера (см. раздел 2).

Специальные функциональные блоки, перечень которых приводится в нижеследующей таблице, служат дополнением к существующим (стандартным) функциям и функциональным блокам, интегрированным в систему ISaGRAF.

Таблица 6.1 - Перечень специальных функциональных блоков

Наименование	Обозначение в ISaGRAF
Блок работы с панелью оператора V04, функционирующей в текстовом режиме	T_V04
Блок программного перезапуска контроллера	T_REBOOT
Блок управления сторожевым таймером Watchdog	T_WDOG
Блок индикации текущего состояния программы диагностики контроллера	T_DIAGSTAT
Блок индикации текущего состояния конкретного объекта диагностики	T_DIAGMOD
Блок для реализации алгоритмической (пользовательской) диагностики	T_DIAGUSER
Блок индикации текущего состояния контроллера	T_INFO
Блок формирования пользовательских сообщений	T_LOG
Блок получения системного времени	T_CLOCK

Применение указанных функциональных блоков позволяет:

- обеспечивать функциональное наращивание целевой задачи в целях наиболее интенсивного использования операционной платформы;
- более удобно и быстро разрабатывать пользовательские приложения;
- отображать текущее состояние контроллера (T_INFO);
- программно перезапускать контроллер (ЦП) (T_REBOOT);
- активировать аппаратный сторожевой таймер (T_WDOG);
- отображать текущую диагностическую информацию (T_DIAGSTAT, T_DIAGMODE);

- формирование/устранение программируемых (пользовательских) отказов и ошибок (T_DIAGUSER);
- формирование пользовательского сообщения (T_LOG);
- получать системное время с точностью до миллисекунд (T_CLOCK).

В специальных функциональных блоках T_WDOG, T_DIAGSTAT, T_DIAGMOD, T_DIAGUSER, T_INFO, T_LOG предусмотрены входы, позволяющие использовать их в прикладных проектах на ЦП контроллеров в режимах резервирования:

MAIN (тип BOOL) - TRUE означает основной контроллер, FALSE - резервный.

В режиме одиночного контроллера значение MAIN игнорируется. При работе в резервированной паре блок запускается только, если значение этого параметра совпадает с текущим статусом контроллера (центрального процессора) и значение PLC равно 0. Используется для запуска блока на резервном контроллере (центральном процессоре) путем ввода значений на основном;

PLC (тип DINT) - Номер контроллера (модуля центрального процессора) в системе, который назначается при проведении процедуры конфигурирования контроллера.

Возможные значения лежат в интервале от 1 до 255 включительно. В режиме одиночного контроллера параметр игнорируется. При работе в резервированной паре блок запускается только, если значение этого параметра совпадает с номером контроллера в системе. Используется для запуска блока на резервном контроллере путем ввода значений на основном. Значение 0 означает игнорировать этот параметр. В случае если значение этого параметра отлично от 0, параметр MAIN игнорируется.

6.1 ФУНКЦИОНАЛЬНЫЙ БЛОК T_V04

Функциональный блок обеспечивает ввод в пользовательскую программу ASCII кода клавиши, нажатой на клавиатуре блока **V03** или **V04 (без индекса «М»)** [10], вывод текстового сообщения на индикатор и выдачу звукового сигнала. Алгоритм может быть использован в решении задачи индикации и управления переменными прикладного проекта непосредственно по месту расположения контроллеров.

Рисунок 6.1 показывает обозначение функционального блока T_V04 в редакторе прикладных программ среды разработки приложений ISaGRAF.

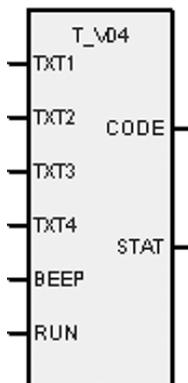


Рисунок 6.1 – Представление функционального блока T_V04 в редакторе языка FBD

Функциональный блок T_V04 имеет следующие входные параметры:

- TXT1* (тип STRING(16)) - первая строка на дисплее;
- TXT2* (тип STRING(16)) - вторая строка на дисплее (только V04);
- TXT3* (тип STRING(16)) - третья строка на дисплее (только V04);
- TXT4* (тип STRING(16)) - четвертая строка на дисплее (только V04);
- BEEP* (тип BOOL) - TRUE означает, что звуковой сигнал устройства включен, FALSE – выключен;
- RUN* (тип BOOL) - TRUE означает запуск блока на исполнение, FALSE – останов блока.

Блок имеет следующие выходные параметры:

CODE (тип SINT) - ASCII код символа, введенного с панели.

Соответствие клавиш и возвращаемых символов приведено в [10,15].
Возможные значения лежат в диапазоне от 0 до 255. Возвращает 0, если нет необработанных введенных символов;

STAT (тип BOOL) - Результат выполнения. Возможные значения:

- TRUE - нормальное выполнение;
- FALSE - внутренняя ошибка при опросе панели оператора.

Если RUN = TRUE, то алгоритм обрабатывает значения остальных входов и формирует значение выхода CODE, выходному сигналу STAT присваивается значение TRUE. В этом случае:

- для V04 на соответствующие строки индикатора будут выводиться сообщения TXT1, TXT2, TXT3, TXT4, каждая длиной до 16 символов (набор отображаемых символов – ASCII);
- для V03 на индикатор будет выводиться сообщение TXT1 длиной до 16 символов (набор отображаемых символов – ASCII);
- при установке входа BEEP в состояние TRUE, будет выдаваться звуковой сигнал;
- величина выходного сигнала CODE будет равняться ASCII-коду нажатой клавиши на клавиатуре блока (или «0», если клавиша не нажата).

Необходимо учесть, что панель оператора выдаёт код клавиши один раз, по факту нажатия, и не учитывает длительность её удержания.

При RUN = FALSE строки индикатора очищаются, и выходному сигналу STAT присваивается значение FALSE.



ВНИМАНИЕ

Запрещается использовать в прикладном проекте более одного блока T_V04!

6.2 ФУНКЦИОНАЛЬНЫЙ БЛОК T_REBOOT

Функциональный блок T_REBOOT в прикладном проекте обеспечивает перезапуск контроллера по инициативе оператора или по событию в прикладной программе.

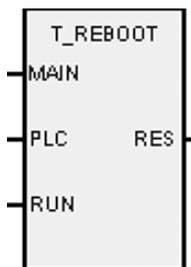


Рисунок 6.2 – Представление функционального блока T_REBOOT в редакторе языка FBD

Функциональный блок T_REBOOT имеет следующие входные параметры (см. рисунок 6.2):

MAIN, PLC – описание смотри раздел 6;

RUN (тип BOOL) - TRUE означает запуск блока, FALSE - останов.

Блок имеет следующие выходные параметры:

RES (тип DINT) - Результат выполнения. Возможные значения:

0 - нормальное выполнение;

1 - недопустимые входные параметры;

2 - внутренняя ошибка.

Количество блоков в прикладном проекте может быть более одного.



ВНИМАНИЕ

Использование блока T_REBOOT в прикладном проекте резервированного контроллера (ЦП) ЗАПРЕЩЕНО!

6.3 ФУНКЦИОНАЛЬНЫЙ БЛОК T_WDOG

Применение в рамках пользовательского приложения алгоритма T_WDOG позволяет активизировать аппаратный сторожевой таймер контроллера (Watchdog). Алгоритм может быть использован в решении задачи автономного перезапуска контроллера в случае непреднамеренного зависания целевой задачи ISaGRAF.

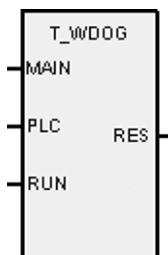


Рисунок 6.3 – Представление функционального блока T_WDOG в редакторе языка FBD

Функциональный блок T_WDOG (см. рисунок 6.3) имеет следующие входные параметры:

MAIN, PLC – описание смотри раздел 6;

RUN (тип BOOL) - TRUE означает запуск блока, FALSE -- останов.

Блок имеет следующие выходные параметры:

RES (тип DINT) - Результат выполнения. Возможные значения:

- 0 - нормальное выполнение;
- 1 - недопустимые входные параметры;
- 2 - внутренняя ошибка.

Количество блоков в прикладном проекте может быть более одного.

Время срабатывания сторожевого таймера ~ 1,2 – 2,0 секунды.

	<p>ВНИМАНИЕ</p> <p>Одновременное использование блока T_WDOG и системного демона watchdogd должно быть исключено! При необходимости использования T_WDOG запретите запуск watchdogd из программы TUNER.</p>
---	---

6.4 ФУНКЦИОНАЛЬНЫЙ БЛОК T_DIAGSTAT

Функциональный блок T_DIAGSTAT предназначен для взаимодействия с программой самодиагностики контроллера и позволяет получить общую информацию о текущем состоянии диагностики. Алгоритм может быть использован в решении задачи мониторинга системы диагностики.

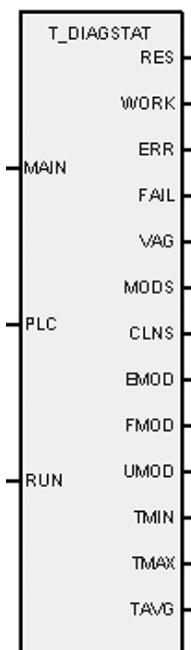


Рисунок 6.4 – Представление функционального блока T_DIAGSTAT в редакторе языка FBD

Функциональный блок T_DIAGSTAT имеет следующие входные параметры (см. рисунок 6.4):

MAIN, *PLC* – описание смотри раздел 6;

RUN (тип BOOL) - TRUE означает запуск блока, FALSE - останов.

Блок имеет следующие выходные параметры:

RES (тип DINT) - Результат выполнения. Возможные значения:

- 0 - нормальное выполнение;
- 1 - недопустимые входные параметры;
- 2 - внутренняя ошибка;

WORK (тип BOOL) - TRUE означает, что диагностирование производится,
FALSE – не производится;

ERR (тип BOOL) - TRUE означает, что обнаружена ошибка,
FALSE - ошибок нет;

FAIL (тип BOOL) - TRUE означает, что обнаружен отказ, FALSE - отказов нет;

VAG (тип BOOL) - TRUE означает наличие неопределенного состояния одного
из модулей диагностики, FALSE – неопределенности нет;

MODS (тип DINT) - Общее число работающих модулей диагностики;

CLNS (тип DINT) - Общее число подключенных клиентов диагностики;

EMOD (тип DINT) - Число модулей диагностики с ошибкой;

FMOD (тип DINT) - Число модулей диагностики с отказом;

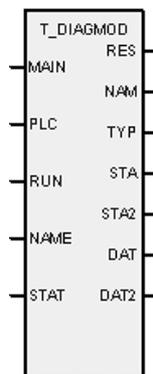
UMOD (тип DINT) - Число модулей диагностики в неизвестном состоянии;

TMIN (тип DINT), *TMAX* (тип DINT), *TAVG* (тип DINT) - Минимальное,
максимальное и среднее зарегистрированное время цикла диагностики,
измеряемое в миллисекундах.

Количество блоков в прикладном проекте может быть более одного.

6.5 ФУНКЦИОНАЛЬНЫЙ БЛОК T_DIAGMOD

Применение в рамках пользовательского приложения алгоритма T_DIAGMOD позволяет отслеживать состояние конкретного модуля (узла) диагностики. Алгоритм может быть использован в решении задачи мониторинга конкретного модуля диагностики.



*Рисунок 6.5 – Представление функционального блока T_DIAGMOD
в редакторе языка FBD*

Функциональный блок T_DIAGMOD имеет следующие входные параметры
(см. рисунок 6.5):

MAIN, *PLC* – описание смотри раздел 6;

RUN (тип BOOL) - TRUE означает запуск блока, FALSE – останов;

NAME (тип STRING(16)) - Символьное имя модуля диагностики.

Позволяет выбрать необходимый модуль по имени. В случае наличия более одного модуля с указанным именем будет выведена информация о первом найденном. Пустое значение означает игнорировать этот параметр при поиске модуля. Имена модулей диагностики доступны на вкладке *TUNER Конфигурирование→Диагностика*;

STAT (тип DINT) - Текущее состояние модуля диагностики.

Позволяет выбрать необходимый модуль по текущему состоянию.
Возможные значения:

0 – нормальное функционирование;

1 – ошибка;

2 – отказ;

3 – неизвестное состояние;

-1 (минус 1) – игнорировать этот параметр при поиске модуля.

В случае наличия более одного модуля с указанным состоянием будет выведена информация о первом найденном.

Блок имеет следующие выходные параметры:

RES (тип DINT) - Результат выполнения. Возможные значения:

0 – нормальное выполнение;

1 – недопустимые входные параметры;

2 – внутренняя ошибка;

10 – модуль не найден;

NAM (тип STRING(16)) - Символьное имя найденного модуля;

TYP (тип STRING(16)) - Тип найденного модуля. Возможные значения:

EXT – внешний модуль (в TeNIX® TUNER – “REMOTE”);

PROC – процесс (в TeNIX® TUNER – “PROCESS”);

ETH – порт Ethernet (в TeNIX® TUNER – “ETHERNET”);

IO – ввод-вывод (в TeNIX® TUNER – “MFCIO”);

FS – файловая система (в TeNIX® TUNER – “FILESYSTEM”);

MFC3000 – шины питания и micro CAN контроллера МФК3000 (1500);

GROUP – группа модулей диагностики;

STA (тип STRING(16)) – Текущее состояние найденного модуля. Возможные значения:

OK – нормальное функционирование;

ERR – ошибка;

FAIL – отказ;

UNK – неизвестное состояние;

STA2 (тип DINT) – Необязательное дополнительное состояние найденного модуля. Возможные значения зависят от типа модуля (см. таблицу);

DAT (тип DINT) – Необязательные дополнительные данные найденного модуля. Возможные значения зависят от типа модуля (см. таблицу);

DAT2 (тип REAL) – Необязательные дополнительные данные найденного модуля в формате REAL.

Описание значений полей STA2 и DAT для конкретных типов модулей диагностики смотри в таблице:

Тип модуля	Нештатная ситуация	STA2	DAT (DAT2)
FS	Файловая система не смонтирована	1	-
MFC3000	Ошибка или отказ шины питания	-	Напряжение на шине в мВ
	Отказ линии CAN		Показатель качества линии (< 7)
IO	Отказ канала модуля	-	Номер канала
	Отказ более одного канала	2	Номер последнего отказавшего канала
	Отказ модуля	1	-

Блок допускает наличие не более двух экземпляров в ресурсе прикладного проекта, в противном случае поведение блоков не определено.

6.6 ФУНКЦИОНАЛЬНЫЙ БЛОК T_DIAGUSER

Функциональный блок T_DIAGUSER предназначен для взаимодействия с диагностикой контроллера и позволяет реализовать пользовательскую диагностику в прикладном проекте. При использовании данного функционального блока в прикладном проекте создается дополнительный модуль диагностики контроллера. Применение данного функционального блока позволяет по событию сформировать/устранить программируемые отказы и ошибки.

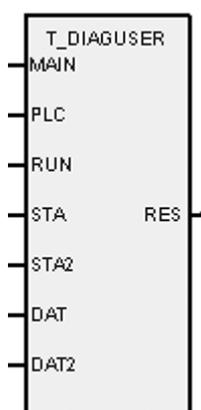


Рисунок 6.6 – Представление функционального блока T_DIAGUSER в редакторе языка FBD

Функциональный блок T_DIAGUSER имеет следующие входные параметры (см. рисунок 6.6):

MAIN, PLC – описание смотри раздел 6;

RUN (тип BOOL) - TRUE означает запуск блока, FALSE - останов. При запуске блока регистрируется дополнительный модуль диагностики с символьным именем «userN», где N - порядковый номер экземпляра функционального блока, в ресурсе прикладного проекта начиная от нуля. При останове блока модуль диагностики удаляется;

STA (тип DINT) - Текущее состояние модуля. Возможные значения:

- 0 - нормальное функционирование;
- 1 – ошибка;
- 2 – отказ;
- 3 - неизвестное состояние;

STA2 (тип DINT) - Необязательное дополнительное состояние модуля. Возможные значения определяются разработчиком прикладного проекта;

DAT (тип DINT) - Необязательное дополнительное состояние модуля. Возможные значения определяются разработчиком прикладного проекта;

DAT2 (тип REAL) - Необязательные данные модуля в формате REAL. В случае если параметр *DAT* равен нулю, в качестве дополнительных данных используется данный параметр.

Блок имеет следующие выходные параметры:

RES (тип DINT) - Результат выполнения. Возможные значения:

- 0 - нормальное выполнение;
- 1 - недопустимые входные параметры;
- 2 - внутренняя ошибка.

Количество блоков в прикладном проекте может быть более одного.

6.7 ФУНКЦИОНАЛЬНЫЙ БЛОК T_INFO

Функциональный блок T_INFO предназначен для отображения текущего состояния контроллера.

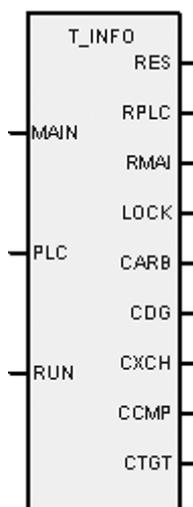


Рисунок 6.7 – Представление функционального блока T_INFO в редакторе языка FBD

Функциональный блок T_INFO имеет следующие входные параметры (см. рисунок 6.7):

MAIN, PLC – описание смотри раздел 6;

RUN (тип BOOL) - TRUE означает запуск блока, FALSE - останов.

Блок имеет следующие выходные параметры:

RES (тип DINT) - Результат выполнения. Возможные значения:

0 - нормальное выполнение;

1 - недопустимые входные параметры;

2 - внутренняя ошибка.

RPLC (тип DINT) - Номер контроллера;

RMAI (тип BOOL) - TRUE означает основной контроллер, FALSE - резервный.
Для локального контроллера этот параметр всегда равен TRUE;

LOCK (тип BOOL) - TRUE означает, что выходы модулей УСО заблокированы,
FALSE – выходы не заблокированы;

CARB (тип DINT) - Текущее время цикла процесса АРБИТР (определяет текущее состояние контроллера), измеряемое в мс. Значение «0» означает, что данный процесс не работает;

CDG (тип DINT) - Текущее время цикла процесса ДИАГНОСТИКА (реализует самодиагностику контроллера), измеряемое в мс;

CXCH (тип DINT) - Текущее время цикла процесса ОБМЕН (реализует синхронизацию состояний прикладной программы в основном и резервном контроллерах), измеряемое в мс. Значение «0» означает, что данный процесс не работает. Для локального контроллера этот параметр всегда равен нулю;

CCMP (тип DINT) – Текущее время цикла процесса СПАВНЕНИЕ (реализует сравнение значений входных сигналов основного и резервного контроллеров), измеряемое в мс. Значение «0» означает, что данный процесс не работает. Для локального контроллера этот параметр всегда равен нулю. (Актуально только для МФК);

CTGT (тип DINT) - Текущее время цикла целевой задачи ISaGRAF (исполняет прикладную программу), измеряемое в мс.

Количество блоков в прикладном проекте может быть более одного.

6.8 ФУНКЦИОНАЛЬНЫЙ БЛОК T_LOG

Применение в прикладном проекте функционального блока T_LOG позволяет генерировать по событию пользовательские сообщения. Сообщения генерируются посредством обращения к службе сообщений, входящей в состав системного программного обеспечения TeNIX® контроллера.

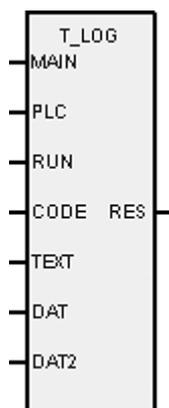


Рисунок 6.8 – Представление функционального блока T_LOG в редакторе языка FBD

Функциональный блок T_LOG имеет следующие входные параметры (см. рисунок 6.8):

MAIN, PLC – описание смотри раздел 6;

RUN (тип BOOL) - TRUE означает запуск блока, FALSE - останов. Новое сообщение генерируется только по положительному фронту значения этого параметра;

CODE (тип DINT) - Код сообщения. Возможные значения лежат в интервале от 20 до 255 включительно и выбираются разработчиком прикладного проекта;

TEXT (тип STRING(16)) - Необязательная дополнительная символьная строка сообщения. Возможные значения выбираются разработчиком прикладного проекта;

DAT (тип DINT) - Необязательные дополнительные данные сообщения. Возможные значения выбираются разработчиком прикладного проекта;

DAT2 (тип REAL) - Необязательные дополнительные данные сообщения в формате REAL. В случае если параметр DAT равен нулю, в качестве дополнительных данных можно использовать данный параметр.

Блок имеет следующие выходные параметры:

RES (тип DINT) - Результат выполнения. Возможные значения:

0 - нормальное выполнение;

1 - недопустимые входные параметры;

2 - внутренняя ошибка.

Количество блоков в прикладном проекте может быть более одного.

6.9 ФУНКЦИОНАЛЬНЫЙ БЛОК T_CLOCK

Функциональный блок T_CLOCK предназначен для получения в технологической задаче в цифровом виде текущего системного времени. Время исчисляется по Гринвичу (UTC).

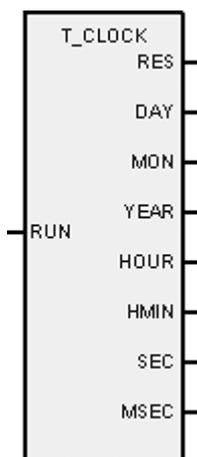


Рисунок 6.9 – Представление функционального блока T_CLOCK в редакторе языка FBD

Функциональный блок T_CLOCK имеет следующие входные параметры:

RUN (тип BOOL) - TRUE означает запуск блока, FALSE - останов.

Блок имеет следующие выходные параметры (см. рисунок 6.9):

RES (тип DINT) - Результат выполнения. Возможные значения:

0 - нормальное выполнение;

1,2,3 - внутренняя ошибка;

DAY (тип DINT) – день месяца (1-31);

MON (тип DINT) – месяц года (1-12);

YEAR (тип DINT) – год;

HOUR (тип DINT) – часы (0-23);

HMIN (тип DINT) – минуты (0-59);

SEC (тип DINT) – секунды (0-59);

MSEC (тип DINT) – миллисекунды (0-999).

Количество блоков в прикладном проекте может быть более одного.

7 ЗАГРУЗКА И ОТЛАДКА ПРИКЛАДНОГО ПРОЕКТА

При установлении связи между отладчиком среды разработки ISaGRAF Workbench и контроллером возможны контроль и управление выполнением прикладного проекта. Для этого при разработке прикладного проекта необходимо в среде ISaGRAF Workbench провести разработку, сборку (с указанием IP адреса контроллера), загрузку и отладку ресурса прикладного проекта.

Загрузка (обновление) ресурса прикладного проекта производится при выполнении команды инструментального меню (*Debug/Download*) среды ISaGRAF Workbench (см. рисунок 7.1).

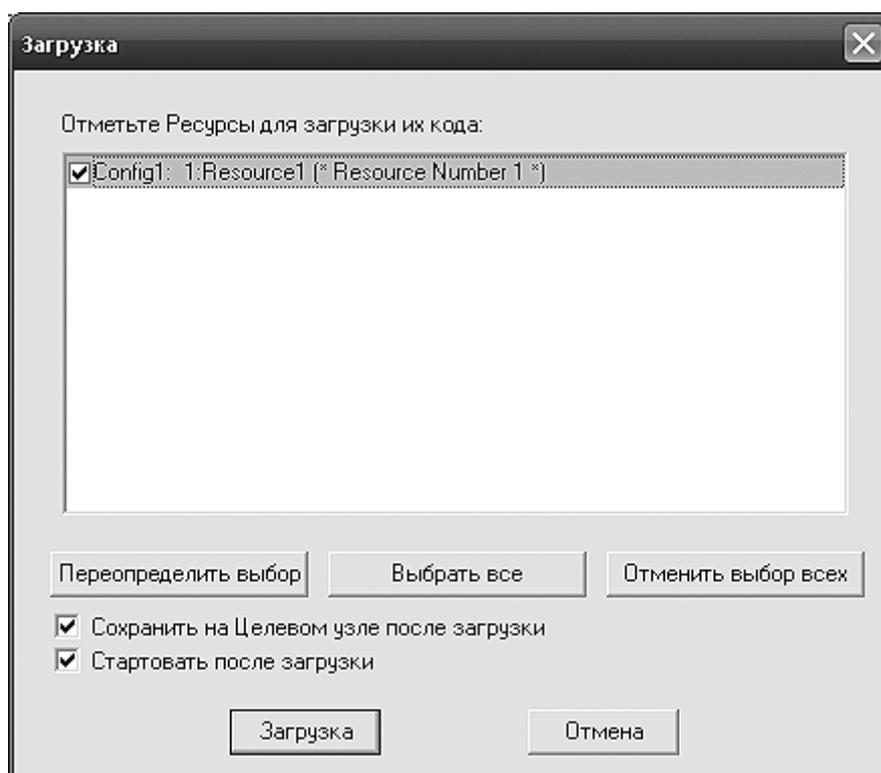


Рисунок 7.1 – Форма загрузки прикладного проекта на контроллер

Загрузка ресурса прикладного проекта в контроллер должна производиться в следующих случаях:

- реализуется первая загрузка или отладка прикладного кода, а также загрузка после удаления ранее загруженного прикладного кода;
- производится обновление отлаженного прикладного кода.

По завершению процедуры загрузки (обновления) прикладного проекта целевая задача ISaGRAF автоматически исполняет загруженный код прикладного проекта (если установлен флаг «Стартовать после загрузки»), начиная с первого цикла. При этом:

- в качестве исходных данных используются значения «по умолчанию» переменных «нового» прикладного проекта;

- переменные с атрибутом retain (хранимые) не восстанавливаются (совместно с обновлением кода прикладного проекта обновляется область хранимых переменных).

	<p>ВНИМАНИЕ</p> <p>При изменении имени проекта (ресурса) удалите на контроллере старый проект (<i>TeNIX TUNER</i>→<i>Конфигурирование</i>→<i>Технологическая задача</i>→<i>Удаление задачи</i>)!</p>
---	---

Отладка прикладного проекта производится при использовании отладчика среды разработки ISaGRAF Workbench и может проводиться в следующих режимах: On-line (непрерывный), Cycle to cycle (по-цикловый).

	<p>ИНФОРМАЦИЯ</p> <p>TeNIX® TUNER на странице <i>Конфигурирование</i>→<i>Технологическая задача</i> позволяет сохранять (делать резервную копию) загруженный проект или восстанавливать ранее сохраненный проект.</p>
---	--

	<p>ВНИМАНИЕ</p> <p>С целью обеспечения безударности загрузки (обновление) ресурса прикладного проекта необходимо проводить либо без подключения контроллера к объекту автоматизации, либо во время плановых остановов указанного объекта.</p>
--	--

	<p>ВНИМАНИЕ</p> <p>Отладку прикладного проекта в режиме Cycle to cycle, а также использование команд Stop/Start (останов/запуск прикладного кода), следует проводить либо без подключения контроллера к объекту автоматизации, либо во время плановых остановов указанного объекта.</p>
---	--

	<p>ВНИМАНИЕ</p> <p>Применение свободного цикла выполнения ресурса не рекомендуется!</p> <p>Для оптимальной работы контроллера рекомендуется выставлять цикл выполнения ресурса как максимально зафиксированное в отладчике время выполнения ресурса, умноженное на 1,5.</p> <p>Для установки цикла необходимо на вкладке <i>Выполнение</i> свойств ресурса (см. рисунок 1.4) установить галочку <i>Циклы</i> и, в открывшемся поле <i>Время Цикла (мс)</i>, ввести необходимое значение в миллисекундах.</p>
---	---

8 ОСОБЕННОСТИ ПРИМЕНЕНИЯ ПРИКЛАДНОГО ПРОЕКТА В РЕЖИМАХ РЕЗЕРВИРОВАНИЯ КОНТРОЛЛЕРА

8.1 ОГРАНИЧЕНИЯ НА ПРИКЛАДНЫЕ ПРОЕКТЫ

Разработка, загрузка и отладка локального прикладного ресурса для контроллера в режимах резервирования, с точки зрения использования всех функций, предоставляемых средой ISaGRAF Workbench, не отличается от работы с контроллером в одиночном режиме. Но при разработке, загрузке и отладке прикладного проекта комплекса в целом, необходимо учитывать ряд особенностей, отличающих проведение указанных процедур от работы с прикладным ресурсом нерезервированного контроллера:

- контроллер ТКМ52 работает только в одиночном режиме;
- прикладные проекты в резервированной паре контроллеров (ЦП) должны быть идентичны (за исключением параметров, описанных в п. 8.2);
- система «зеркализации» данных целевой задачи накладывает ограничения на «размер данных» ресурса.

	<p>ВНИМАНИЕ</p> <p>В резервированном контроллере (ЦП) использование более одного ресурса недопустимо.</p> <p>Для подпрограмм прикладного проекта, разработанных на языках FC и SFC система «зеркализации» работать не будет!</p>
---	---

		<u>Информация о ресурсе N2</u>																																																	
Всего объектов	: 489																																																		
Размер данных	: 6 кВ (7078 byte)	1024																																																	
Переменных	: 83 (1063 byte)																																																		
Экземпляров ФБД	: 53 (2897 byte)																																																		
Использование ФБД																																																			
<table border="1"> <thead> <tr> <th>Тип</th> <th>Экземпляров</th> <th>Размер</th> <th>Всего</th> </tr> </thead> <tbody> <tr> <td>ECHDL</td> <td>8</td> <td>192</td> <td>1536</td> </tr> <tr> <td>USCR16</td> <td>2</td> <td>144</td> <td>288</td> </tr> <tr> <td>ADC16</td> <td>2</td> <td>136</td> <td>272</td> </tr> <tr> <td>BLINK</td> <td>11</td> <td>24</td> <td>264</td> </tr> <tr> <td>CTU</td> <td>14</td> <td>16</td> <td>224</td> </tr> <tr> <td>ADJUSTTM</td> <td>8</td> <td>12</td> <td>96</td> </tr> <tr> <td>USCR8</td> <td>1</td> <td>72</td> <td>72</td> </tr> <tr> <td>ADC8</td> <td>1</td> <td>72</td> <td>72</td> </tr> <tr> <td>DAC16</td> <td>2</td> <td>24</td> <td>48</td> </tr> <tr> <td>DAC8</td> <td>1</td> <td>16</td> <td>16</td> </tr> <tr> <td>R_TRIG</td> <td>3</td> <td>3</td> <td>9</td> </tr> </tbody> </table>	Тип	Экземпляров	Размер	Всего	ECHDL	8	192	1536	USCR16	2	144	288	ADC16	2	136	272	BLINK	11	24	264	CTU	14	16	224	ADJUSTTM	8	12	96	USCR8	1	72	72	ADC8	1	72	72	DAC16	2	24	48	DAC8	1	16	16	R_TRIG	3	3	9			
Тип	Экземпляров	Размер	Всего																																																
ECHDL	8	192	1536																																																
USCR16	2	144	288																																																
ADC16	2	136	272																																																
BLINK	11	24	264																																																
CTU	14	16	224																																																
ADJUSTTM	8	12	96																																																
USCR8	1	72	72																																																
ADC8	1	72	72																																																
DAC16	2	24	48																																																
DAC8	1	16	16																																																
R_TRIG	3	3	9																																																

Рисунок 8.1 – Окно общей информации о загруженном в контроллер (ЦП) ресурсе

Параметр «размер данных» (далее VD) ресурса и его ограничение (далее VDmax) доступны на странице TeNIX® TUNER «Информация о ресурсе N» (см. рисунок 8.1). Доступ к странице осуществляется через ссылку *Общая→Прикладное программное обеспечение→Ресурс X→подробнее* (см. рисунок 8.2).

Прикладное программное обеспечение	
БПО	ISaGRAF
Библиотека (TIL PRO Std)	Активировано
Библиотека (TIL PRO Com)	Не активировано
Ресурс N2	символьная таблица есть >>> подробнее

Рисунок 8.2 – Информация о целевой задаче контроллера на «общей» странице TeNIX® TUNER

VDmax - максимальный объем данных ресурса, допустимый в режимах резервирования контроллера (ЦП). Отображается на странице справа от графического представления параметра VD.

Параметр VD показывается значением на красном фоне  при превышении ограничения VDmax. В этом случае необходимо провести оптимизацию прикладного проекта с целью уменьшения объема кода проекта.

8.2 СОЗДАНИЕ ПРИКЛАДНОГО ПРОЕКТА

Таблица 8.1 Исполнения среды разработки ISaGRAF Workbench

Среда ISaGRAF Workbench		Способ создания проекта
Исполнение	Функциональность	
ISaGRAF 5 WB Distributed unlimited	Создание распределенных приложений для системы контроллеров в сети Ethernet. На каждом ЦП возможно выполнение нескольких локальных ресурсов. Обмен данными между локальными и удаленными ресурсами. Неограниченное число точек ввода-вывода	1
ISaGRAF 5 WB multiscan unlimited	Создание приложений для автономных ЦП. На каждом ЦП возможно выполнение нескольких локальных ресурсов. Обмен данными между локальными ресурсами. Неограниченное число точек ввода-вывода	2
ISaGRAF 5 WB essential unlimited	Создание приложений для автономных ЦП. Неограниченное число точек ввода-вывода	
ISaGRAF 5 WB essential I/Os	Создание приложений для автономных ЦП. Количество точек ввода-вывода – 32-512	



ИНФОРМАЦИЯ

При разработке прикладного проекта комплекса необходимо учитывать, что число точек ввода-вывода, характерное для исполнения среды ISaGRAF Workbench, не должно быть меньше числа используемых каналов ввода-вывода в прикладном ресурсе одного ЦП.

В зависимости от имеющегося исполнения среды ISaGRAF Workbench (определяется при заказе по таблице 7.1), существует два способа разработки прикладного проекта резервированных контроллеров (ЦП), которые описаны далее.

Способ 1

Используя рекомендации, приведенные в документации на ISaGRAF Workbench, создайте новый прикладной проект с аппаратной архитектурой, содержащей два контроллера (ЦП) с указанием индивидуальных IP адресов внешних интерфейсов ЦП. В качестве наименования ЦП (должны быть различны) рекомендуется использовать обозначение контроллеров (ЦП) комплекса в системе автоматизации.

В рамках единого прикладного проекта создайте ресурс для одного из контроллеров (ЦП) комплекса, используя дополнительные рекомендации, приведенные в разделе 7. При разработке ресурса используйте следующие компоненты, включенные в среду разработки при определении контроллеров:

- драйверы модулей УСО контроллера;
- специальные типы данных;
- специальные функциональные блоки.

Проведите проверку синтаксиса программ и компоновку ресурса для выбранного ЦП. Если необходимо, проведите при использовании среды ISaGRAF Workbench компоновку и компиляцию «неполного» проекта, загрузку и отладку ресурса в выбранном ЦП комплекса.

Последовательно используя процедуры экспорта и импорта ресурса, создайте прикладной ресурс для второго ЦП комплекса, идентичный по составу с исходным ресурсом. При импорте укажите размещение ресурса во втором контроллере комплекса и номер ресурса, отличный от номера исходного ресурса (см. рисунок 8.3 и рисунок 8.4).

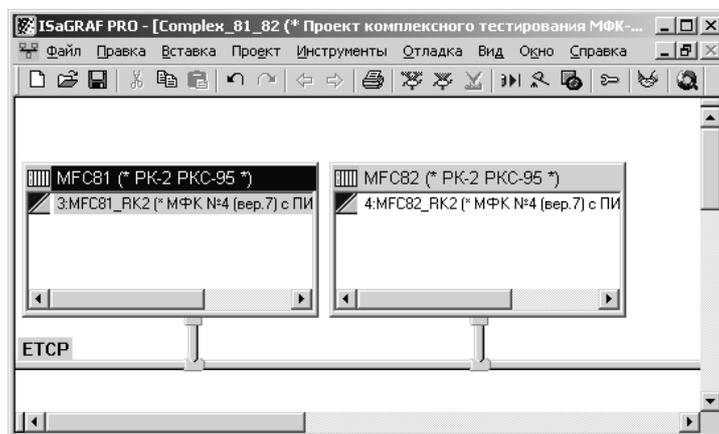


Рисунок 8.3 – Аппаратная архитектура прикладного проекта РК МФК

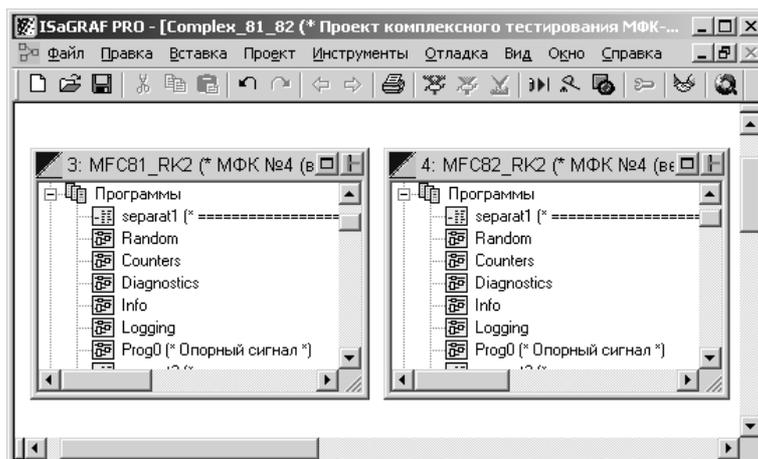
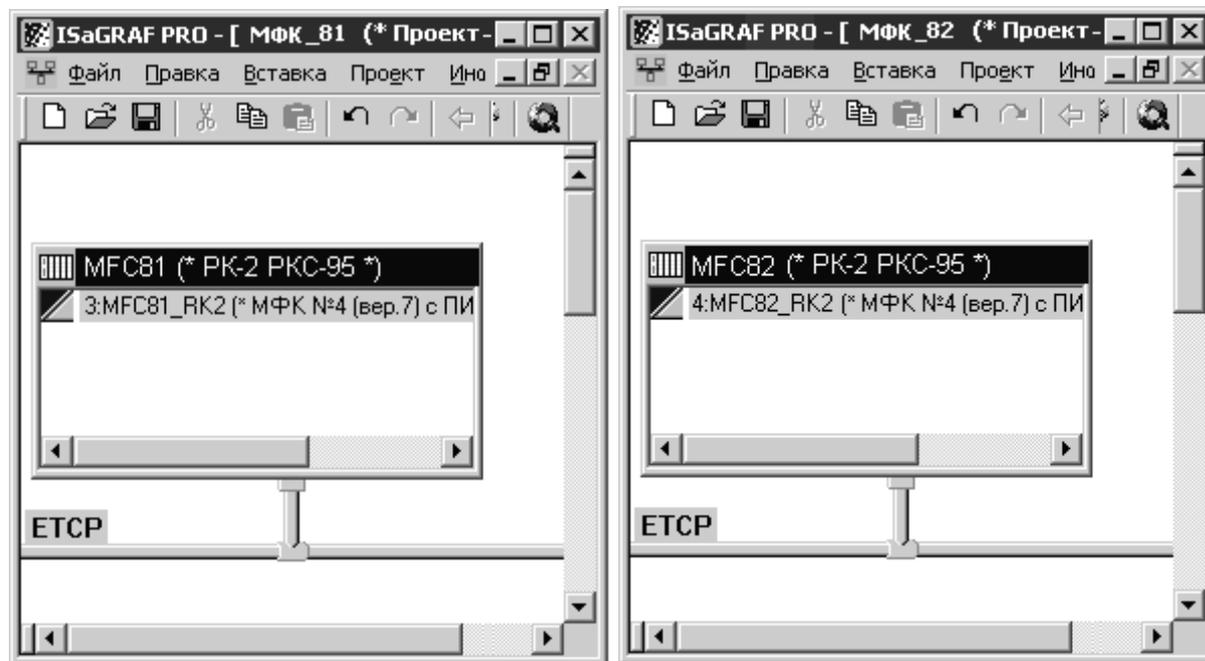


Рисунок 8.4 – Архитектура связей прикладного проекта комплекса

Проведите компоновку и компиляцию проекта комплекса в целом. Далее, при использовании среды ISaGRAF Workbench, установленной на инженерной (рабочей) станции, становится возможным проведение загрузки и отладки ресурсов единого прикладного проекта.

Способ 2

Используя рекомендации, приведенные в документации на ISaGRAF Workbench, создайте новый прикладной проект с аппаратной архитектурой, содержащей один контроллер (см. рисунок 8.5а) с указанием индивидуального IP адреса одного из ЦП комплекса.



а

б

Рисунок 8.5 - Аппаратная архитектура моноресурсных прикладных проектов для резервированной пары контроллеров (ЦП)

При разработке зафиксируйте создание директории *<Имя проекта>*, например MFC81 (см. рисунок 8.5а), которая содержит файлы прикладного проекта и располагается по адресу установки среды ISaGRAF Workbench в каталоге *Prj*.

В рамках прикладного проекта выбранного ЦП создайте ресурс, используя дополнительные рекомендации, приведенные в разделе 7. При разработке ресурса используйте следующие компоненты, включенные в среду разработки при определении контроллеров:

- драйверы модулей УСО контроллера;
- специальные типы данных;
- специальные функциональные блоки.

Проведите проверку синтаксиса программ и компоновку ресурса для выбранного ЦП. Если необходимо, проведите при использовании среды ISaGRAF Workbench компоновку и компиляцию проекта, загрузку и отладку ресурса в выбранном ЦП комплекса.

По адресу установки среды ISaGRAF Workbench в каталоге *Prj* создайте директорию, с именем отличным от имени ранее используемой директории *<Имя проекта>*, например MFC82 (см. рисунок 8.5б). Скопируйте файлы прикладного проекта во вновь созданную директорию. В наименовании моноресурсных проектов рекомендуется использовать обозначение ЦП комплекса в системе автоматизации.

Проведите компоновку и компиляцию прикладного проекта с указанием индивидуального IP адреса второго ЦП в комплексе, предварительно открыв проект из вновь созданной директории.

Далее, при использовании среды ISaGRAF Workbench, установленной на инженерной (рабочей) станции, становится возможным проведение загрузки и отладки прикладных проектов, последовательно (поочередно) в каждом из ЦП комплекса.

Для реализации в прикладной задаче резервированных контроллеров (ЦП) любой из функций специальных функциональных блоков необходимо:

- создать в программе два экземпляра специального функционального блока;
- при необходимости реализации указанных функций в каждом из контроллеров, для каждого экземпляра блока указать на входе PLC число, соответствующее индивидуальному номеру процессора (контроллера) заданному в TUNER'e. При этом переменная на входе MAIN может иметь любое значение;
- при необходимости реализации указанных функций в контроллере с определенным статусом (MASTER или SLAVE), для каждого экземпляра блока указать на входе PLC число, равное 0, и установить на входе MAIN соответствующее значение (TRUE или FALSE).

Такая реализация функций обеспечивает идентичность прикладных ресурсов комплекса и гарантирует исполнение в каждом из процессоров (контроллеров) только одного экземпляра специального функционального блока.

8.3 ЗАГРУЗКА ПРИКЛАДНОГО ПРОЕКТА

Прикладные ресурсы ЦП комплекса, разработанные в среде ISaGRAF Workbench, могут располагаться:

- в одном едином проекте при разработке по *Способу 1*;
- в двух локальных проектах при разработке по *Способу 2*.

Загрузка ресурсов прикладного проекта может производиться в работающие контроллеры (ЦП) комплекса, которые находятся в состоянии, отличном от состояния «Конфигурирование». Загрузка ресурсов прикладного проекта должна производиться в предварительно сконфигурированные ЦП. При выполнении процедуры загрузки ресурсов прикладного проекта комплекса реализуется автоматическое сохранение ресурсов прикладного проекта.

Для нормального функционирования комплекса соответствующие ресурсы прикладного проекта должны быть загружены в каждый из ЦП комплекса (независимо от статуса) в следующих случаях:

- первая загрузка или отладка ресурсов прикладного проекта комплекса;
- после удаления ранее загруженного прикладного ресурса;
- после замены (ремонта) одного из ЦП комплекса;
- обновление ресурсов отлаженного прикладного проекта.

Рекомендуется следующая последовательность действий при загрузке (обновлении) ресурсов прикладного проекта:

- используя средства аппаратной диагностики или контролируя состояние выхода MAIN функционального блока T_INFO, идентифицируйте ЦП со статусом SLAVE;
- установите переключатель «Режим» ЦП SLAVE в положение *Блок.* (для МФК) или в положение *Lock out* (МФК3000, МФК1500);
- произведите загрузку соответствующих ресурсов прикладного проекта в ЦП комплекса;
- по окончании загрузки ресурсов и в отсутствие отказа ЦП в статусе SLAVE, установите переключатель «Режим» указанного ЦП в положение *Вых. активн.* (МФК) или *Run* (МФК3000, МФК1500).

По окончании загрузки (обновления) возникновение неисправности обмена данными в комплексе (в ЦП статусом SLAVE возникает отказ, в ЦП со статусом MASTER – ошибка), необходимо интерпретировать как несовпадение прикладных ресурсов в контроллерах (ЦП) комплекса. Идентификация неисправностей производится при использовании программы TUNER. Причинами возникновения таких неисправностей могут являться:

- загрузка/обновление ресурса только в одном ЦП комплекса (загрузите/обновите ресурс во втором контроллере комплекса);
- некорректная разработка ресурсов прикладного проекта комплекса (проверьте ресурсы на соответствие требованиям, приведенным в п. 8.2, проведите дополнительную компоновку и компиляцию прикладного проекта, повторите загрузку);

- возможно некорректное завершение процедур загрузки и сохранение ресурса в ЦП (перезапустите ЦП со статусом SLAVE, при необходимости повторите загрузку ресурсов).

При успешном завершении процедуры загрузки (обновления) ресурсов прикладного проекта в каждом из ЦП комплекса целевая задача ISaGRAF автоматически выполняет загруженный ресурс прикладного проекта, начиная с первого цикла. При этом:

- в качестве исходных данных используются значения «по умолчанию» переменных «нового» прикладного проекта.
- переменные с атрибутом retain (хранимые) не восстанавливаются (совместно с обновлением кода прикладного проекта обновляется область хранимых переменных).

	<p>ВНИМАНИЕ</p> <p>С целью обеспечения безударности загрузки (обновление) ресурсов прикладного проекта комплекса необходимо проводить либо без подключения модулей вывода к объекту автоматизации, либо во время плановых остановов указанного объекта.</p>
---	--

8.4 ОТЛАДКА ПРИКЛАДНОГО ПРОЕКТА

Для проведения отладки связь между инженерной станцией и контроллерами (ЦП) комплекса с прикладными ресурсами, разработанными в среде ISaGRAF Workbench (см. п. 8.2), может быть установлена запуском отладчика:

- при открытии одного единого проекта (разработка по *Способу 1*);
- при поочередном открытии двух проектов для локальных контроллеров (разработка по *Способу 2*).

Отладка ресурсов прикладного проекта может производиться на запущенных контроллерах ЦП комплекса, которые находятся в состоянии, отличном от состояния «Конфигурирование». Соответствующие ресурсы прикладного проекта должны быть предварительно загружены в ЦП (см. п. 8.2).

При установлении связи между отладчиком среды разработки ISaGRAF Workbench и ЦП возможны:

- в контроллере (ЦП) статуса MASTER возможно проведение полноценной отладки прикладного ресурса (контроль и управление);
- в контроллере (ЦП) статуса SLAVE возможен контроль выполнения прикладного ресурса и процесса «зеркализации» его состояния.

Полноценная отладка прикладного проекта комплекса производится в ЦП со статусом MASTER и может проводиться в следующих режимах: On-line (непрерывный), Cycle to cycle (по-цикловый).

	<p>ВНИМАНИЕ</p> <p>Отладку прикладного проекта в режиме Cycle to cycle, а также использование команд Stop/Start (останов/запуск прикладного кода), следует проводить либо без подключения комплекса к объекту автоматизации, либо во время плановых остановов указанного объекта.</p>
---	--

При использовании в прикладном проекте специальных функциональных блоков (см. раздел 6) и функционировании в комплексе процесса межконтроллерного обмена, управление соответствующими переменными прикладного ресурса ЦП MASTER позволяет реализовать следующие функции в ЦП статуса SLAVE:

- индикация текущего состояния контроллера;
- программный перезапуск контроллера;
- активизация аппаратного сторожевого таймера;
- индикация текущей диагностической информации;
- формирование/устранение программируемой неисправности;
- формирование пользовательского сообщения.

8.5 СПЕЦИАЛЬНЫЙ РЕЖИМ ИЗМЕНЕНИЯ ПРИКЛАДНОГО ПРОЕКТА

При необходимости изменения проекта на резервированном контроллере (контроллерах) без останова управления объектом имеется возможность воспользоваться «Спец. режимом» [6-8].

	<p>ВНИМАНИЕ</p> <p>Режим используется для резервированной пары процессоров (контроллеров), является дополнительным (не штатным) и накладывает на прикладного программиста АСУ требования по аккуратности внесения изменений в технологическую задачу.</p> <p>Ответственность за управление объектом при изменении статусов ЦП в этом случае лежит на прикладном программисте. Разработчик контроллера гарантирует только синхронизацию данных из старого (на основном ЦП) в новый проект (на резервном) по совпадающим по имени, типу и области видимости (принадлежности программе или функциональному блоку) переменным, в том числе внутренним переменным алгоблоков.</p>
---	---

При применении этого (специального) режима накладывается ряд ограничений на правила изменения проекта:

- Нельзя делать «очистку и пересборку» проекта. При этом может нарушиться внутренняя нумерация блоков, и тогда работа режима не гарантируется.
- Нельзя менять одинаковые функциональные блоки местами в одной и той же программе.
- Нельзя добавлять и удалять из середины программы функциональные блоки для работы с диагностикой, а также блоки T_COM. Также нельзя добавлять или удалять из середины списка программ программу, содержащую эти блоки.

- Изменения типа переменной (в том числе изменения размера строковой переменной) считаются полноценными изменениями переменной. Такие переменные копироваться не будут.
- Особенно осторожно надо быть с массивами. Если старая переменная была массивом с границами [N..M], а новая - массив с границами [N+delta..M+delta], то копирование будет производиться следующим образом:

```
old[N] -> new[N+delta]
old[N+1] -> new[N+delta+1]
...
old[M] -> new[M+delta]
```

Порядок применения специального режима

- 8.5.1 Переведите резервный контроллер в режим блокировки выходов (см. [6-8]).
- 8.5.2 Загрузите в работающий РЕЗЕРВНЫЙ контроллер (ЦП) обновленный (с соблюдением вышеперечисленных ограничений) проект.
- 8.5.3 Переведите программное обеспечение резервного контроллера (ЦП) в специальный режим (см. [6-8]).

Измените статус ОСНОВНОГО контроллера (ЦП) в резервированной паре на РЕЗЕРВНЫЙ и загрузите обновленный проект в новый резервный контроллер.

9 СПИСОК ЛИТЕРАТУРЫ

- 1 **Многофункциональный контроллер МФК.** Руководство по эксплуатации. ДАРЦ.420002.001РЭ.
- 2 **Резервированный комплекс контроллеров МФК.** Руководство по эксплуатации. ДАРЦ.421457.003РЭ.
- 3 **Многофункциональный контроллер МФК3000.** Руководство по эксплуатации. ДАРЦ.420002.002РЭ.
- 4 **Многофункциональный контроллер МФК1500.** Руководство по эксплуатации. ДАРЦ.420002.003РЭ.
- 5 **Технологический контроллер моноблочный ТКМ52.** Руководство по эксплуатации. ДАРЦ.421243.000РЭ.
- 6 **Системное программное обеспечение TeNIX (МФК3000).** Руководство оператора. ДАРЦ.73022-4X 34 01-1.
- 7 **Системное программное обеспечение TeNIX (МФК1500).** Руководство оператора. ДАРЦ.73022-4X 34 01-3.
- 8 **Системное программное обеспечение TeNIX (МФК).** Руководство оператора. ДАРЦ.73022-4X 34 01-2.
- 9 **Системное программное обеспечение TeNIX (4823) (ТКМ52).** Руководство оператора. ДАРЦ.73029-4X 34 01-1.
- 10 **Панель оператора V04.** Руководство по эксплуатации. ДАРЦ.426476.501РЭ.
- 11 **VisiBuilder.** Руководство пользователя.
- 12 **Библиотека алгоритмов для среды ISaGRAF TIL PRO Std v.1.2.** Описание применения. ДАРЦ.70015-12 31 01-Х.
- 13 **Библиотека алгоритмов для среды ISaGRAF TIL PRO Com v.2.0.** Описание применения. ДАРЦ.70020-20 31 01-Х.
- 14 **Определения контроллеров ТЕКОН для среды ISaGRAF v.4 (PRO) и v.5 Workbench. Версия 0.X.X.** CD. ДАРЦ.70023-0X 99 02-Х.
- 15 **Блок клавиатуры и индикации V03.** Руководство по эксплуатации. ДАРЦ.426449.001РЭ.

Приложение А (обязательное) Определение контроллеров МФК и ТКМ52

<input checked="" type="checkbox"/> Targets	<input checked="" type="checkbox"/> MFC (*Контроллер МФК*)	<input checked="" type="checkbox"/> L16I
<input checked="" type="checkbox"/> Data types	<input checked="" type="checkbox"/> VBOOL (*Значение BOOL с достоверностью*)	<input checked="" type="checkbox"/> L16IF
	<input checked="" type="checkbox"/> VDINT (*Значение DINT с достоверностью*)	<input checked="" type="checkbox"/> P40
	<input checked="" type="checkbox"/> VREAL (*Значение REAL с достоверностью*)	<input checked="" type="checkbox"/> SENSOR
<input checked="" type="checkbox"/> "C" functions and function blocks	<input checked="" type="checkbox"/> T_CLOCK (*Системное время*)	<input checked="" type="checkbox"/> SENSORF
	<input checked="" type="checkbox"/> T_DIAGMOD (*Модуль диагностики*)	<input checked="" type="checkbox"/> T12
	<input checked="" type="checkbox"/> T_DIAGSTAT (*Состояние диагностики*)	<input checked="" type="checkbox"/> T12F
	<input checked="" type="checkbox"/> T_DIAGUSER (*Пользовательская диагностика*)	<input checked="" type="checkbox"/> T3101
	<input checked="" type="checkbox"/> T_INFO (*Информация о контроллере*)	<input checked="" type="checkbox"/> T3101F
	<input checked="" type="checkbox"/> T_LOG (*Пользовательские сообщения*)	<input checked="" type="checkbox"/> T3102
	<input checked="" type="checkbox"/> T_REBOOT (*Перезапуск контроллера*)	<input checked="" type="checkbox"/> T3102F
	<input checked="" type="checkbox"/> T_V04 (*Панель V04*)	<input checked="" type="checkbox"/> T3204
	<input checked="" type="checkbox"/> T_WDOG (*Аппаратный сторожевой таймер*)	<input checked="" type="checkbox"/> T3204F
<input checked="" type="checkbox"/> I/O devices		<input checked="" type="checkbox"/> T3205
	<input checked="" type="checkbox"/> A08	<input checked="" type="checkbox"/> T3205F
	<input checked="" type="checkbox"/> A08F	<input checked="" type="checkbox"/> T3501
	<input checked="" type="checkbox"/> A16	<input checked="" type="checkbox"/> T3501F
	<input checked="" type="checkbox"/> A16F	<input checked="" type="checkbox"/> T3601
	<input checked="" type="checkbox"/> A16FI	<input checked="" type="checkbox"/> T3601PWM
	<input checked="" type="checkbox"/> A16FO	<input checked="" type="checkbox"/> T3602
	<input checked="" type="checkbox"/> A16I	<input checked="" type="checkbox"/> T3602PWM
	<input checked="" type="checkbox"/> A16O	<input checked="" type="checkbox"/> T3603
	<input checked="" type="checkbox"/> D32	<input checked="" type="checkbox"/> T3604
	<input checked="" type="checkbox"/> D32I	<input checked="" type="checkbox"/> T3702
	<input checked="" type="checkbox"/> D32O	<input checked="" type="checkbox"/> T3702CNT
	<input checked="" type="checkbox"/> D40	<input checked="" type="checkbox"/> T3703
	<input checked="" type="checkbox"/> D40PWM	<input checked="" type="checkbox"/> V04M
	<input checked="" type="checkbox"/> D48	<input checked="" type="checkbox"/> V04MI
	<input checked="" type="checkbox"/> F24A	<input checked="" type="checkbox"/> V04MO
	<input checked="" type="checkbox"/> F24D	<input type="checkbox"/> Conversion functions
	<input checked="" type="checkbox"/> L16	<input checked="" type="checkbox"/> Networks
	<input checked="" type="checkbox"/> L16F	<input checked="" type="checkbox"/> ETCP (*protocol based on TCP/IP*)
		<input checked="" type="checkbox"/> HSD (*Host System Driver for local binding*)
		<input checked="" type="checkbox"/> ISARSI (*standard Remote Serial Interface link*)
		<input type="checkbox"/> Defined words

Приложение Б
(обязательное)

Перечень драйверов модулей ввода-вывода МФК и ТКМ52

Обозначение в ISaGRAF	Тип модуля	Назначение	Атрибуты ISaGRAF			
			Тип	Кол. каналов	Тип переменных	Тип устройства
A16	A16	Значения входных каналов в условных единицах	Input	16	VDINT	Сложное
		Значения выходных каналов в условных единицах	Output	2	VDINT	
A16F		Значения входных каналов в float	Input	16	VREAL	Сложное
		Значения выходных каналов в float	Output	2	VREAL	
A16I		Input	Значения входных каналов в условных единицах	16	VDINT	Простое
A16FI			Значения входных каналов в float		VREAL	Простое
A16O		Output	Значения выходных каналов в условных единицах	2	VDINT	Простое
A16FO			Значения выходных каналов в float		VREAL	Простое
A08	A08	Значения выходных каналов в условных единицах	Output	8	VDINT	Простое
A08F		Значения выходных каналов в float			VREAL	Простое
D32	D32	Значения дискретных входов	Input	16	VBOOL	Сложное
		Значения дискретных выходов	Output	16	VBOOL	
D32I		Значения дискретных входов	Input	16	VBOOL	Простое
D32O		Значения дискретных выходов	Output	16	VBOOL	Простое
D40	D40	Значения дискретных выходов	Output	40	VBOOL	Простое
D40PWM		Значения длительности импульсов			VDINT	Простое
L16	L16	Значения входных каналов в условных единицах	Input	16	VDINT	Простое
L16F		Значения входных каналов в float			VREAL	Простое
L16I	L16i	Значения входных каналов в условных единицах	Input	17	VDINT	Простое
L16IF		Значения входных каналов в float			VREAL	Простое
P40	P40	Значения дискретных выходов	Output	40	VBOOL	Простое
F24D	F24	Значения дискретных входов	Input	24	VBOOL	Простое
F24A		Значения входных счетчиков/частоты			VDINT	Простое
D48	D48	Значения дискретных входов	Input	48	VBOOL	Простое
T12	T12	Значения входных каналов тензодатчиков в условных единицах	Input	12	VDINT	Простое
T12F		Значения входных каналов в float			VREAL	Простое

Приложение В
(обязательное)
Перечень драйверов модулей ТЕКОНИК

Обозначение в ISaGRAF	Тип модуля	Назначение	Атрибуты ISaGRAF			
			Тип	Кол. каналов	Тип переменных	Тип устройства
SENSOR	ТСТ11, Метран-100	Значения входных каналов в условных единицах	Input	1	VDINT	Простое
SENSORF		Значения входных каналов в float			VREAL	Простое
T3101	T3101	Значения входных каналов в условных единицах	Input	8	VDINT	Простое
T3101F		Значения входных каналов в float			VREAL	Простое
T3102	T3102	Значения входных каналов в условных единицах	Input	6	VDINT	Простое
T3102F		Значения входных каналов в float			VREAL	Простое
T3204	T3204	Значения входных каналов в условных единицах	Input	8	VDINT	Простое
T3204F		Значения входных каналов в float			VREAL	Простое
T3205	T3205	Значения входных каналов в условных единицах	Input	8	VDINT	Простое
T3205F		Значения входных каналов в float			VREAL	Простое
T3501	T3501	Значения выходных каналов в условных единицах	Output	4	VDINT	Простое
T3501F		Значения выходных каналов в float			VREAL	Простое
T3601	T3601	Значения дискретных выходов	Output	8	VBOOL	Простое
T3601PWM		Значения длительности импульсов			VDINT	Простое
T3602	T3602	Значения дискретных выходов	Output	8	VBOOL	Простое
T3602PWM		Значения длительности импульсов			VDINT	Простое
T3702	T3702	Значения дискретных входов	Input	16	VBOOL	Простое
T3702CNT		Значения счетчиков импульсов			VDINT	Простое
T3603	T3603	Значения дискретных выходов	Output	16	VBOOL	Простое
T3604	T3604	Значения дискретных выходов	Output	12	VBOOL	Простое
T3703	T3703	Значения дискретных входов	Input	8	VBOOL	Простое

Приложение Г (обязательное) Определение контроллера МФК3000

<input checked="" type="checkbox"/> Targets <input checked="" type="checkbox"/> MFC (*Контроллер МФК3000*) <input checked="" type="checkbox"/> Data types <input checked="" type="checkbox"/> VBOOL (*Значение BOOL с достоверностью*) <input checked="" type="checkbox"/> VDINT (*Значение DINT с достоверностью*) <input checked="" type="checkbox"/> VREAL (*Значение REAL с достоверностью*) <input checked="" type="checkbox"/> "C" functions and function blocks <input checked="" type="checkbox"/> T_CLOCK (*Системное время*) <input checked="" type="checkbox"/> T_DIAGMOD (*Модуль диагностики*) <input checked="" type="checkbox"/> T_DIAGSTAT (*Состояние диагностики*) <input checked="" type="checkbox"/> T_DIAGUSER (*Пользовательская диагностика*) <input checked="" type="checkbox"/> T_INFO (*Информация о контроллере*) <input checked="" type="checkbox"/> T_LOG (*Пользовательские сообщения*) <input checked="" type="checkbox"/> T_REBOOT (*Перезапуск контроллера*) <input checked="" type="checkbox"/> T_V04 (*Панель V04*) <input checked="" type="checkbox"/> T_WDOG (*Аппаратный сторожевой таймер*) <input type="checkbox"/> I/O devices <input type="checkbox"/> AI16 <input type="checkbox"/> AI16F <input type="checkbox"/> AI32 <input type="checkbox"/> AI32F <input type="checkbox"/> AOC8 <input type="checkbox"/> AOC8F <input type="checkbox"/> DI16 <input type="checkbox"/> DI16CNT <input type="checkbox"/> DI16CNT2 <input type="checkbox"/> DI32 <input type="checkbox"/> DI32CNT <input type="checkbox"/> DI32CNT2	<input type="checkbox"/> DI48 <input type="checkbox"/> DI48CNT <input type="checkbox"/> DI48CNT2 <input type="checkbox"/> DO16 <input type="checkbox"/> DO16PWM <input type="checkbox"/> DO24 <input type="checkbox"/> DO24PWM <input type="checkbox"/> DO32 <input type="checkbox"/> DO32PWM <input type="checkbox"/> FP6CNT2 <input type="checkbox"/> FP6DF <input type="checkbox"/> FP6F <input type="checkbox"/> FP6O <input type="checkbox"/> LI16 <input type="checkbox"/> LI16F <input type="checkbox"/> V04M <input type="checkbox"/> V04MI <input type="checkbox"/> V04MO <input type="checkbox"/> Conversion functions <input checked="" type="checkbox"/> Networks <input checked="" type="checkbox"/> ETCP (*protocol based on TCP/IP*) <input checked="" type="checkbox"/> HSD (*Host System Driver for local binding*) <input checked="" type="checkbox"/> ISARSI (*standard Remote Serial Interface link*)
--	--

Приложение Д
(обязательное)
Перечень драйверов модулей ввода-вывода МФК3000

Обозначение в ISaGRAF	Тип модуля	Назначение	Атрибуты ISaGRAF			
			Тип	Кол-во каналов	Тип переменных	Тип устройства
LI16	LI16	Значения каналов в условных единицах	Input	16	VDINT	Простое
LI16F		Значения каналов в float			VREAL	Простое
FP6F	FP6	Значения частоты в float	Input	6	VREAL	Простое
FP6DF		Значения дифференциала частоты в float				Простое
FP6CNT2		Значения 32p счетчиков			VDINT	Простое
FP6O		Значения дискретных выходов	Output	6	VBOOL	Простое
AI16	AI16	Значения входных каналов в условных единицах	Input	16	VDINT	Простое
AI16F		Значения входных каналов в float			VREAL	Простое
AI32	AI32	Значения входных каналов в условных единицах	Input	32	VDINT	Простое
AI32F		Значения входных каналов в float			VREAL	Простое
AOC8	AOC8	Значения выходных каналов в условных единицах	Output	8	VDINT	Простое
AOC8F		Значения выходных каналов в float			VREAL	Простое
DI16	DI16	Значения дискретных входов	Input	16	VBOOL	Простое
DI16CNT		Значения каналов 0-15 – значения 16 разрядных счетчиков. Значения каналов 16-31 – значения таймеров для 16 разрядных счетчиков. Значения канала 0 и 16 образуют пару - счетчик количества импульсов и таймер времени в мс, за которое эти импульсы пришли, для физического канала 0 модуля		32		
DI16CNT2		Значения 32p счетчиков		16	Простое	

Обозначение в ISaGRAF	Тип модуля	Назначение	Атрибуты ISaGRAF			
			Тип	Кол-во каналов	Тип переменных	Тип устройства
DI32	DI32	Значения дискретных входов	Input	32	VBOOL	Простое
DI32CNT		Значения каналов 0-15 – значения 16 разрядных счетчиков. Значения каналов 16-31 – значения таймеров для 16 разрядных счетчиков. Значения канала 0 и 16 образуют пару - счетчик количества импульсов и таймер времени в мс, за которое эти импульсы пришли, для физического канала 0 модуля		32	VDINT	Простое
DI32CNT2		Значения 32р счетчиков		16		Простое
DI48	DI48	Значения дискретных входов	Input	48	VBOOL	Простое
DI48CNT		Значения каналов 0-15 – значения 16 разрядных счетчиков. Значения каналов 16-31 – значения таймеров для 16 разрядных счетчиков. Значения канала 0 и 16 образуют пару - счетчик количества импульсов и таймер времени в мс, за которое эти импульсы пришли, для физического канала 0 модуля		32	VDINT	Простое
DI48CNT2		Значения 32р счетчиков		16		Простое
DO16	DO16	Значения дискретных выходов	Output	16	VBOOL	Простое
DO16PWM		Значение длительности импульса в мс на выходе канала ШИМ		8	VDINT	Простое
DO24	DO24	Значения дискретных выходов	Output	24	VBOOL	Простое
DO24PWM		Значение длительности импульса в мс на выходе канала ШИМ		12	VDINT	Простое
DO32	DO32	Значения дискретных выходов	Output	32	VBOOL	Простое
DO32PWM		Значение длительности импульса в мс на выходе канала ШИМ		16	VDINT	Простое

Приложение Е (обязательное) Определение контроллера МФК1500

<ul style="list-style-type: none"> <input checked="" type="checkbox"/> Целевые системы: <ul style="list-style-type: none"> <input checked="" type="checkbox"/> MFC (*Контроллер МФК1500*) <input checked="" type="checkbox"/> Типы данных <ul style="list-style-type: none"> <input checked="" type="checkbox"/> VBOOL (*Значение BOOL с достоверностью*) <input checked="" type="checkbox"/> VDINT (*Значение DINT с достоверностью*) <input checked="" type="checkbox"/> VREAL (*Значение REAL с достоверностью*) <input checked="" type="checkbox"/> Функции и функциональные блоки "С" <ul style="list-style-type: none"> <input checked="" type="checkbox"/> T_CLOCK (*Системное время*) <input checked="" type="checkbox"/> T_DIAGMOD (*Модуль диагностики*) <input checked="" type="checkbox"/> T_DIAGSTAT (*Состояние диагностики*) <input checked="" type="checkbox"/> T_DIAGUSER (*Пользовательская диагностик. <input checked="" type="checkbox"/> T_INFO (*Информация о контроллере*) <input checked="" type="checkbox"/> T_LOG (*Пользовательские сообщения*) <input checked="" type="checkbox"/> T_REBOOT (*Перезапуск контроллера*) <input checked="" type="checkbox"/> T_V04 (*Панель V04*) <input checked="" type="checkbox"/> T_WDOG (*Аппаратный сторожевой таймер*) <input checked="" type="checkbox"/> Устройства ввода-вывода <ul style="list-style-type: none"> <input checked="" type="checkbox"/> CAI4 <input checked="" type="checkbox"/> CAI4F <input checked="" type="checkbox"/> CAI8 <input checked="" type="checkbox"/> CAI8F <input checked="" type="checkbox"/> CAIG16 <input checked="" type="checkbox"/> CAIG16F <input checked="" type="checkbox"/> CAIG8 <input checked="" type="checkbox"/> CAIG8F <input checked="" type="checkbox"/> CAO2 <input checked="" type="checkbox"/> CAO2F <input checked="" type="checkbox"/> CAO4 <input checked="" type="checkbox"/> CAO4F 	<ul style="list-style-type: none"> <input checked="" type="checkbox"/> CDI16 <input checked="" type="checkbox"/> CDI16CNT <input checked="" type="checkbox"/> CDI16CNT2 <input checked="" type="checkbox"/> CDI32 <input checked="" type="checkbox"/> CDI32CNT <input checked="" type="checkbox"/> CDI32CNT2 <input checked="" type="checkbox"/> CDO16 <input checked="" type="checkbox"/> CDO16PWM <input checked="" type="checkbox"/> CDO32 <input checked="" type="checkbox"/> CDO32PWM <input checked="" type="checkbox"/> CLG04 <input checked="" type="checkbox"/> CLG04F <input checked="" type="checkbox"/> CLG08 <input checked="" type="checkbox"/> CLG08F <input checked="" type="checkbox"/> CLG16 <input checked="" type="checkbox"/> CLG16F <input checked="" type="checkbox"/> V04M <input checked="" type="checkbox"/> V04MI <input checked="" type="checkbox"/> V04MO <input type="checkbox"/> Функции преобразования <input checked="" type="checkbox"/> Сети <ul style="list-style-type: none"> <input checked="" type="checkbox"/> ETCP (*protocol based on TCP/IP*) <input checked="" type="checkbox"/> HSD (*Host System Driver for local binding*) <input checked="" type="checkbox"/> ISARSI (*standard Remote Serial Interface link*) <input type="checkbox"/> Слова-определители
---	---

Приложение Ж
(обязательное)
Перечень драйверов модулей ввода-вывода МФК1500

Обозначение в ISaGRAF	Тип модуля	Назначение	Атрибуты ISaGRAF			
			Тип	Кол-во каналов	Тип переменных	Тип устройства
CAI4	AI4	Значения входных каналов в условных единицах	Input	4	VDINT	Простое
CAI4F		Значения входных каналов в float			VREAL	Простое
CAI8	AI8	Значения входных каналов в условных единицах	Input	8	VDINT	Простое
CAI8F		Значения входных каналов в float			VREAL	Простое
CAIG16	AIG16	Значения входных каналов в условных единицах	Input	16	VDINT	Простое
CAIG16F		Значения входных каналов в float			VREAL	Простое
CAIG8	AIG8	Значения входных каналов в условных единицах	Input	8	VDINT	Простое
CAIG8F	ВХОДЫ ADO24	Значения входных каналов в float			VREAL	Простое
CLG16	LIG16	Значения входных каналов в условных единицах	Input	16	VDINT	Простое
CLG16F		Значения входных каналов в float			VREAL	Простое
CLG08	LIG8	Значения входных каналов в условных единицах	Input	8	VDINT	Простое
CLG08F		Значения входных каналов в float			VREAL	Простое
CLG04	LIG4	Значения входных каналов в условных единицах	Input	4	VDINT	Простое
CLG04F		Значения входных каналов в float			VREAL	Простое
CAO2	AOC2	Значения выходных каналов в условных единицах	Output	2	VDINT	Простое
CAO2F		Значения выходных каналов в float			VREAL	Простое
CAO4	AOC4	Значения выходных каналов в условных единицах	Output	4	VDINT	Простое
CAO4F		Значения выходных каналов в float			VREAL	Простое

Обозначение в ISaGRAF	Тип модуля	Назначение	Атрибуты ISaGRAF			
			Тип	Кол-во каналов	Тип переменных	Тип устройства
CDI16	DI16 входы DIO32	Значения дискретных входов	Input	16	VBOOL	Простое
CDI16CNT		Значения каналов 0-15 – значения 16 разрядных счетчиков. Значения каналов 16-31 – значения таймеров для 16 разрядных счетчиков. Значения канала 0 и 16 образуют пару - счетчик количества импульсов и таймер времени в мс, за которое эти импульсы пришли, для физического канала 0 модуля		32	VDINT	Простое
CDI16CNT2		Значения 32р счетчиков		16		Простое
CDI32	DI32	Значения дискретных входов	Input	32	VBOOL	Простое
CDI32CNT		Как для DI16		32	VDINT	Простое
CDI32CNT2		Значения 32р счетчиков		16		Простое
CDO16	DO16,	Значения дискретных выходов	Output	16	VBOOL	Простое
CDO16PWM	выходы ADO24 выходы DIO32	Значение длительности импульса на выходе канала ШИМ, в мс		8	VDINT	Простое
CDO32	DO32	Значение дискретных выходов	Output	32	VBOOL	Простое
CDO32PWM		Значение длительности импульса на выходе канала ШИМ, в мс		16	VDINT	Простое

Приложение И (обязательное) Драйвер ввода-вывода панели оператора V04M

Обозначение в ISaGRAF	Тип модуля	Назначение	Атрибуты ISaGRAF			
			Тип	Кол. каналов	Тип переменных	Тип устройства
V04M	V04M	Значения дискретных входов на устройстве	Input	8	BOOL	Сложное
		Значения дискретных выходов на устройстве	Output	8	BOOL	
V04MI		Значения дискретных входов на устройстве	Input	8	BOOL	Простое
V04MO		Значения дискретных выходов на устройстве	Output	8	BOOL	Простое

Панель оператора V04M позволяет использовать в проекте, разработанном в VisiBuilder [11], пользовательские переменные целевой задачи ISaGRAF.

Система обмена данными V04M↔«целевая задача ISaGRAF» включает сервер v04md, обрабатывающий запросы панели V04M, и драйвер ядра ISaGRAF, предоставляющий возможность использовать в прикладном проекте дискретные входы и выходы.

Для доступа к дискретным каналам панели оператора V04M (опция панели оператора) [10] используется драйвер V04M.

Сервер v04md предоставляет панели V04M следующие функции:

И.1 Чтение и запись переменных ISaGRAF.

Для чтения и записи переменных ISaGRAF необходимо, чтобы вместе с прикладным проектом ISaGRAF на контроллер загружалась символьная таблица ресурса.

Символьная таблица может быть полной или усеченной. В случае усеченной таблицы необходимо присвоить пользовательские адреса всем переменным, которые будет запрашивать панель, чтобы ISaGRAF включил их в усеченную таблицу.

Чтобы вывести значение переменной ISaGRAF, в проекте VisiBuilder необходимо завести переменную сервера с именем, составленным по определенной схеме, приведенной ниже.

Переменным типа BOOL, SINT, DINT, REAL, STRING соответствуют переменные в VisiBuilder с тем же именем.

Для указания элемента массива используется запись NAME[index], где NAME - имя переменной-массива, index - номер элемента в массиве.

Для указания поля в структуре используется запись NAME:FIELD, где NAME - имя переменной-структуры, FIELD - имя поля (двоеточие вместо общепринятой точки

используется из-за того, что VisiBuilder некорректно обрабатывает загрузку ранее сохраненного проекта с именами переменных, содержащими точку).

Если имя переменной ISaGRAF содержит строчные латинские символы – например, run123, то в VisiBuilder они должны быть заменены на соответствующие прописные - например, RUN123.

После объявления переменной в VisiBuilder можно использовать это имя в полях ввода-вывода на экранах проекта VisiBuilder.

Массивы структур и структуры, поля которых являются массивами или структурами, не поддерживаются.

Если переменная является локальной для программы ресурса прикладного проекта ISaGRAF, тогда после имени (но до индекса массива или поля структуры) необходимо добавить @ и имя программы, в которой определена переменная.

Пример —

LOCALVAR@PROGRAM

LOCALSTR@PROGRAM:VALUE

LOCALARR@PROGRAM[3]

И.2 Поддержка дискретных входов/выходов в ISaGRAF.

Для поддержки дискретных входов/выходов в целевой задаче ISaGRAF необходимо подключить в проекте сложное устройство V04M, состоящее из двух простых (V04MI для входов, V04MO для выходов). Каждое из простых устройств предоставляет 8 дискретных каналов.

	<p>ВНИМАНИЕ</p> <p>Из-за особенностей протокола обмена данными панели V04M изменения выходных каналов в ISaGRAF передаются на панель с определенной задержкой. В зависимости от количества и периода обновлений переменных, которые отображает панель, задержка в передаче может составлять от 0.1 секунды (когда переменных несколько или они обновляются часто) до 1 секунды (когда никаких переменных на экране нет).</p>
---	---

И.3. Поддержка экранов аварий V04M

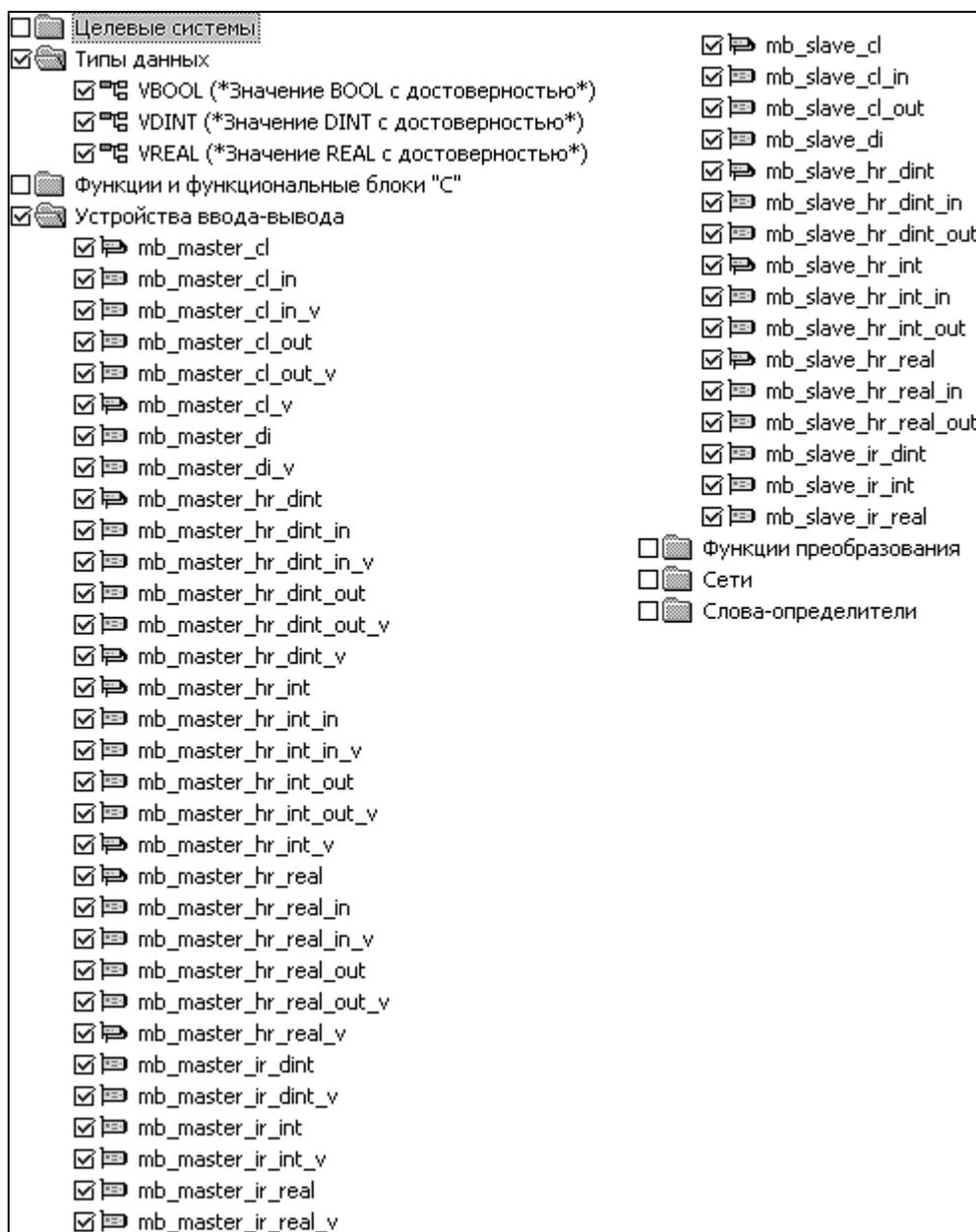
VisiBuilder позволяет при создании проекта использовать один или несколько экранов тревог и определить один или несколько кодов тревог.

Для использования этой возможности в прикладном проекте ISaGRAF необходимо создать глобальную переменную V04M_ALARM. Эта переменная должна быть равна коду тревоги (заданному в VisiBuilder), или 0, если тревоги нет.

И.4 Поддержка авторизации экранов V04M

Среда разработки VisiBuilder предоставляет возможность задания пароля для перехода между экранами проекта. Для поддержки данной функции в проекте ISaGRAF необходимо создать глобальную внутреннюю DINT переменную с именем PASS.

Приложение К (обязательное) Определение драйверов системы Modbus



Приложение Л (обязательное) Драйверы системы Modbus

Обозначение в ISaGRAF	Регистр	Каналов max	Назначение	Атрибуты ISaGRAF		
				Тип	Тип канала	Тип драйвера
mb_master_cl	Coils	-	mb_master_cl in mb_master_cl out	Сложное		
mb_master_cl in		2000	Чтение каналов	Input	BOOL	Простое
mb_master_cl out		2000	Запись каналов	Output	BOOL	Простое
mb_master_cl_v		-	mb_master_cl in v mb_master_cl out v	Сложное		
mb_master_cl in v		2000	Чтение каналов с валидностью	Input	VBOOL	Простое
mb_master_cl out v		2000	Запись каналов с валидностью	Output	VBOOL	Простое
mb_master_hr_int	Holding Registers	-	mb_master_hr_int in mb_master_hr_int out	Сложное		
mb_master_hr_int in		125	Чтение каналов	Input	DINT	Простое
mb_master_hr_int out		125	Запись каналов	Output	DINT	Простое
mb_master_hr_int_v		-	mb_master_hr_int in v mb_master_hr_int out v	Сложное		
mb_master_hr_int in v		125	Чтение каналов с валидностью	Input	VDINT	Простое
mb_master_hr_int out v		125	Запись каналов с валидностью	Output	VDINT	Простое
mb_master_hr_dint		-	mb_master_hr_dint in mb_master_hr_dint out	Сложное		
mb_master_hr_dint in		62	Чтение каналов	Input	DINT	Простое
mb_master_hr_dint out		62	Запись каналов	Output	DINT	Простое
mb_master_hr_dint_v		-	mb_master_hr_dint in v mb_master_hr_dint out v	Сложное		
mb_master_hr_dint in v		62	Чтение каналов с валидностью	Input	VDINT	Простое
mb_master_hr_dint out v		62	Запись каналов с валидностью	Output	VDINT	Простое
mb_master_hr_real		-	mb_master_hr_real in mb_master_hr_real out	Сложное		
mb_master_hr_real in		62	Чтение каналов	Input	REAL	Простое
mb_master_hr_real out		62	Запись каналов	Output	REAL	Простое
mb_master_hr_real_v		-	mb_master_hr_real in v mb_master_hr_real out v	Сложное		
mb_master_hr_real in v		62	Чтение каналов с валидностью	Input	VREAL	Простое
mb_master_hr_real out v		62	Запись каналов с валидностью	Output	VREAL	Простое
mb_master_ir_int	Input Registers	125	Чтение каналов	Input	DINT	Простое
mb_master_ir_int v		125	Чтение каналов с валидностью		VDINT	Простое
mb_master_ir_dint		62	Чтение каналов		DINT	Простое
mb_master_ir_dint v		62	Чтение каналов с валидностью		VDINT	Простое
mb_master_ir_real		62	Чтение каналов		REAL	Простое
mb_master_ir_real v		62	Чтение каналов с валидностью		VREAL	Простое
mb_master_di	Discrete Inputs	2000	Чтение каналов	Input	BOOL	Простое
mb_master_di v		2000	Чтение каналов с валидностью		VBOOL	Простое
mb_slave_cl	Coils	-	mb_slave_cl in mb_slave_cl out	Сложное		
mb_slave_cl in		2000	Чтение каналов	Input	BOOL	Простое
mb_slave_cl out		2000	Запись каналов	Output	BOOL	Простое
mb_slave_ir_int	Input Registers	10000	Чтение каналов	Output	DINT	Простое
mb_slave_ir_dint		5000	Чтение каналов		DINT	Простое
mb_slave_ir_real		5000	Чтение каналов		REAL	Простое
mb_slave_di	Discrete Inputs	2000	Чтение каналов	Output	BOOL	Простое

Обозначение в ISaGRAF	Регистр	Каналов max	Назначение	Атрибуты ISaGRAF		
				Тип	Тип канала	Тип драйвера
mb_slave_hr_int	Holding Registers	-	mb_slave_hr_int_in mb_slave_hr_int_out	Сложное		
mb_slave_hr_int_in		10000	Чтение каналов	Input	DINT	Простое
mb_slave_hr_int_out		10000	Запись каналов	Output	DINT	Простое
mb_slave_hr_dint		-	mb_slave_hr_dint_in mb_slave_hr_dint_out	Сложное		
mb_slave_hr_dint_in		5000	Чтение каналов	Input	DINT	Простое
mb_slave_hr_dint_out		5000	Запись каналов	Output	DINT	Простое
mb_slave_hr_real		-	mb_slave_hr_real_in mb_slave_hr_real_out	Сложное		
mb_slave_hr_real_in		5000	Чтение каналов	Input	REAL	Простое
mb_slave_hr_real_out		5000	Запись каналов	Output	REAL	Простое

