

КРУГОЛ™. ИНТЕГРИРОВАННАЯ СРЕДА РАЗРАБОТКИ

Руководство Пользователя

ПРИЛОЖЕНИЯ

КРУГОЛ™. Интегрированная среда разработки. Руководство Пользователя/1-е изд. Приложения.

© 1992-2022. ООО НПФ «КРУГ». Все права защищены.

Никакая часть настоящего издания ни в каких целях не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами, будь то электронные или механические, включая фотографирование, магнитную запись или иные средства копирования или сохранения информации, без письменного разрешения владельцев авторских прав.

Все упомянутые в данном издании товарные знаки и зарегистрированные товарные знаки принадлежат своим законным владельцам.

ООО НПФ «КРУГ»

440028, г. Пенза, ул. Титова 1

Тел. +7 (8412) 49-97-75, 49-94-14

E-mail: support@krug2000.ru

http:// www.krug2000.ru



СОДЕРЖАНИЕ

Стр.

A. БАЗА ДАННЫХ РЕАЛЬНОГО ВРЕМЕНИ. ПРИЛОЖЕНИЕ А (Справочное). _____ **A-2**

A.1 Переменные _____ **A-2**

A.1.1 Входная аналоговая переменная _____ A-3

A.1.2 Входная дискретная переменная _____ A-7

A.1.3 Дискретная выходная переменная _____ A-11

A.1.4 Аналоговая выходная переменная _____ A-15

A.1.5 Переменная ручного ввода _____ A-22

B. ПРИМЕРЫ ПРОГРАММ СТ. ПРИЛОЖЕНИЕ В (Справочное). _____ **B-1**

B.1 Примеры _____ **B-1**

B.1.1 Пример 1. Формирование сообщений в протокол событий _____ B-1

B.1.2 Пример 2. Реализация расчетной задачи _____ B-2

B.1.3 Пример 3. Имитация входного аналогового параметра _____ B-3

B.1.4 Пример 4. Противоаварийные защиты и блокировки _____ B-4

B.1.5 Пример 5. Обмен паспортов переменных между абонентами ПТК _____ B-6

B.1.6 Пример 6. Управление задвижкой с электроприводом _____ B-7

B.1.7 Пример 7. Управление электрооборудованием (вентилятор) _____ B-15

B.1.8 Пример 8. Управление насосом _____ B-19

B.1.9 Пример 9. Нарботка оборудования _____ B-27

B.1.10 Пример 10. Расчет накапливаемых значений за различные отчетные интервалы
_____ B-28

B.1.11 Пример 11. Формирование печатных документов _____ B-30

B.1.12 Пример 12 Формирование точек трендов _____ B-34

C. ПРИМЕРЫ ПРОГРАММЫ ФБД. ПРИЛОЖЕНИЕ С (Справочное). _____ **C-1**

C.1 Пример _____ **C-1**

C.1.1 Пример создания схемы ФБД _____ C-1

D. СООБЩЕНИЯ ТРАНСЛЯТОРА ЯЗЫКА КРУГОЛ . ПРИЛОЖЕНИЕ D (Справочное). _____ **D-1**

D.1 Сообщения об ошибках _____ **D-1**

D.2 Предупреждение транслятора _____ **D-18**

А. БАЗА ДАННЫХ РЕАЛЬНОГО ВРЕМЕНИ. ПРИЛОЖЕНИЕ А (Справочное).

А.1 Переменные

В описании Базы данных реального времени используются следующие обозначения и определения:

№ в БД СО (СБД)	Номер атрибута переменной базы данных, используемый Сервером базы данных, Графическим интерфейсом и другими приложениями SCADA КРУГОЛ-2000
№ в СО	Номер атрибута, используемый КРУГОЛ в Станции оператора
№ в контр	Номер атрибута, используемый КРУГОЛ в контроллере
Доступ из КРУГОЛ	Показывает, доступен ли данный атрибут на чтение/запись из программ пользователя на языке КРУГОЛ
нд	Атрибут не доступен
чт	Атрибут доступен на чтение
зп	Атрибут доступен на запись
Режим обмена данными	Режим обмена данными между УСО и сервером БД
тек. знач.	Атрибут передается из УСО в каждом ответе на запрос сервера БД по данной переменной (т.е. с периодом обновления БД по соответствующему каналу)
паспорт	Атрибут передается в виде паспорта (полного перечня атрибутов) переменной. Паспорт передается в следующих ситуациях: <ul style="list-style-type: none"> - в УСО (контроллер) - при изменении любого атрибута переменной через таблицу настройки переменной в графическом интерфейсе, - в УСО (контроллер) - при выполнении реакции графического интерфейса «Установить значение», - в УСО (контроллер) - при выполнении в программе пользователя для сервера БД функции передачи паспорта соответствующего типа переменной ПОС (ВА) и т.п. - в сервер БД - при выполнении в программе пользователя для УСО функции передачи паспорта соответствующего типа переменной ПОС (ВА) и т.п. - в сервер БД - при изменении атрибута через интерфейс программы show в контроллере, - при обновлении БД по каналу связи с УСО после запуска сервера БД или при восстановлении связи по каналу
не передается	Атрибут не передается в любом направлении

Пустая ячейка таблицы означает, что данный атрибут отсутствует в УСО (контроллере).

ВХОДНАЯ АНАЛОГОВАЯ ПЕРЕМЕННАЯ

А.1.1 Входная аналоговая переменная

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длина (байт)	Доступ из КРУГОЛ	
0			RecordID	целый	4	нд.				
1	1	1	Номер канала	целый	2	чт.	целый	1	чт/зп	не пере- дается
2	2	2	Номер переменной в УСО	целый	2	чт/зп*3	целый	1	чт/зп	не пере- дается
3	3	3	Номер платы	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
4	4	4	Номер входа	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
5	5	5	Позиция	строка	8/24	нд	строка	8	нд	паспорт
6	6	6	Имя 1	строка	8	нд	строка	8	нд	паспорт
7	7	7	Имя 2	строка	8	нд	строка	8	нд	паспорт
8	8	8	Единица измерения	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
9	9	9	Тип датчика	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
10	10	10	Тип линеаризации шкалы	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
11	11	11	Начало шкалы	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
12	12	12	Конец шкалы	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
13	13	13	Нижняя предупредительная граница	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
14	14	14	Верхняя предупредительная граница	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
15	15	15	Нижняя предаварийная граница	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
16	16	16	Верхняя предаварийная граница	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
17	17	17	Гистерезис сигнализации	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
18	18	18	Время определения скорости изменения параметра	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
19	19	19	Граница сигнализации по скорости роста	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
20	20	20	Граница сигнализации по скорости падения	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
21	21	21	Постоянная фильтра от 0,000 до 1,000	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
22	22	22	Максимально возможная (достоверная) скорость	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
23	23	23	Тип замены недостоверного значения	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
24	24	24	Рабочее значение переменной	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
25	25	25	Отсечка нуля	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
26	26	26	Апертура	вещ.	4	чт/зп*3	вещ.	4	чт/зп	паспорт
27	27	27	Служебный атрибут №1	вещ.	4	чт/зп*3	вещ.	4	чт	паспорт
28	28	28	Текущее значение до преобразования (контроллер)	вещ.	4	чт/зп*2	вещ.	4	чт/зп*1	тек. знач

ВХОДНАЯ АНАЛОГОВАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длина (байт)	Доступ из КРУГОЛ	
29	29	29	Текущее значение после преобразования (контроллер)	вещ.	4	чт/зп*2	вещ.	4	чт/зп*1	тек. знач
30	30	30	Снятие переменной с опроса	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. знач
31	31	31	Снятие переменной с сигнализации	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. знач
32	32	32	Свободный атрибут №1	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. знач
33	33	33	Свободный атрибут №2	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. знач
34	34	34	Неисправность канала	лог.	1	чт/зп*3	лог.	1 бит	чт.	тек. знач
35	35	35	Новое нарушение <Неисправность канала>	лог.	1	чт/зп*3	лог.	1 бит	чт.	тек. знач
36	36	36	Служебный атрибут №2	целый	1	чт/зп*3	лог.	1 бит	чт.	тек. знач
37	37	37	Служебный атрибут №3	целый	1	чт/зп*3	лог.	1 бит	чт.	тек. знач
38	38	38	Нарушена нижняя предаварийная граница	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
39	39	39	Новое нарушение нижней предаварийной границы	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
40	40	40	Нарушена верхняя предаварийная граница	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
41	41	41	Новое нарушение верхней предаварийной границы	лог.	1	чт./зп*2	лог.	1 бит	чт.	тек. знач
42	42	42	Нарушена нижняя предупредительная граница	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
43	43	43	Новое нарушение нижней предупредительной границы	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
44	44	44	Нарушена верхняя предупредительная граница	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
45	45	45	Новое нарушение верхней предупредительной границы	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
46	46	46	Нарушена граница по скорости роста	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
47	47	47	Новое нарушение границы по скорости роста	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
48	48	48	Нарушена граница по скорости падения	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
49	49	49	Новое нарушение границы по скорости падения	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
50	50	50	Переменная в норме	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
51	51	51	Переход переменной в нормальное состояние	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
52	52	52	Сигнализация по достоверности	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
53	53	53	Новая сигнализация по достоверности	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
54	62	62	Цвет отображения сигнализации	целый	1	чт/зп*3	целый	1	чт.	тек. знач

ВХОДНАЯ АНАЛОГОВАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длина (байт)	Доступ из КРУГОЛ	
55	54	54	Значение сигнала ниже начала шкалы	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
56	55	55	Значение сигнала ниже начала шкалы (новое)	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
57	56	56	Значение сигнала выше конца шкалы	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
58	57	57	Значение сигнала выше конца шкалы (новое)	лог.	1	чт/зп*2	лог.	1 бит	чт.	тек. знач
59	58	58	Нарушение <Обрыв>	лог.	1	чт/зп*3	лог.	1 бит	чт.	тек. знач
60	59	59	Новое нарушение <Обрыв>	лог.	1	чт/зп*3	лог.	1 бит	чт.	тек. знач
61	60	60	Нарушение <Перегрузка>	лог.	1	чт/зп*3	лог.	1 бит	чт.	тек. знач
62	61	61	Новое нарушение <Перегрузка>	лог.	1	чт/зп*3	лог.	1 бит	чт.	тек. знач
63	63	63	Служебный атрибут №4	целый	1	чт/зп*3	целый	1	чт.	паспорт
64	64		Текущее значение до (СО)	вещ.	4	чт/зп*2				
65	65		Текущее значение после (СО)	вещ.	4	чт/зп*2				
66	66		Длинная позиция	строка	16	нд				
67	67		Длинное имя	строка	64	нд				
68	68		Физическое начало шкалы(код АЦП)	целый	4	чт/зп				
69	69		Физический конец шкалы (код АЦП)	целый	4	чт/зп				
70	70		Коррекция 1(тип/номер)	ссылка	20	нд				
71	71		Коррекция 2(тип/номер)	ссылка	20	нд				
72	72		Коррекция 3(Адрес хол Спая)	ссылка	20	нд				
73	73		№ алгоритма нестандарт обработки 1	целый	2	чт/зп*4				
74	74		№ алгоритма нестандарт обработки 2	целый	2	чт/зп*4				
75	75		№ алгоритма нестандарт обработки 3	целый	2	чт/зп*4				
76	76		№ алгоритма нестандарт обработки 4	целый	2	чт/зп*4				
77	77		Вкл/выкл нестанд обработок 1	лог.	1	чт/зп*4				
78	78		Вкл/выкл нестанд обработок 2	лог.	1	чт/зп*4				
79	79		Вкл/выкл нестанд обработок 3	лог.	1	чт/зп*4				
80	80		Вкл/выкл нестанд обработок 4	лог.	1	чт/зп*4				
81	81		Основной/Резерв1/Резерв2 /	целый	1	чт/зп*3				
82	82		Место обработки	целый	1	чт/зп				
83	83		Место расположения паспорта	целый	1	чт/зп				
84	84		Начало шкалы трендов	вещ.	4	чт/зп				
85	85		Конец шкалы трендов	вещ.	4	чт/зп				
86	86		Номер участка (главного)	целый	2	чт/зп				

ВХОДНАЯ АНАЛОГОВАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длина (байт)	Доступ из КРУГОЛ	
87	87		Номер канала в комсервере	целый	2	чт/зп				
88	88		Номер м/схемы (главной)	целый	2	чт/зп				
89	89		Метка времени	время	8	нд				
90	90		Снять с опроса в СО	лог.	1	чт/зп				
91	91		Снять с сигнализации в СО	лог.	1	чт/зп				
92	92		Номер УСО в канале	целый	2	чт/зп				
93	93		Тип регистра УСО	целый	2	чт/зп				
94	94		ALARM1	целый	2	чт/зп				
95	95		ALARM2	целый	2	чт/зп				
96	96		Текущее значение (Код АЦП)	целый	2	чт/зп				
97	97		Периодичность обработки (номер списка)	целый	1	чт/зп				
98	98		Цвет состояния	целый	1	чт/зп*2				
99	99		Качество	целый	4	чт/зп				

Примечание:

*1- атрибут может использоваться для записи только тогда, когда атрибут №3 равен 0.

*2 – атрибут может использоваться для записи только тогда, когда атрибут №1 равен 0 и не включены нестандартные обработки в сервере БД (т.е. атрибуты №77-80 равны 0).

*3 – атрибут может использоваться для записи только тогда, когда атрибут №1 равен 0

*4 – будьте внимательны, запись в данные атрибуты управляет алгоритмами обработки данной переменной

ВНИМАНИЕ!!!

При включенной нестандартной обработке по переменной для корректной работы алгоритма обработки необходимо из программы пользователя записывать значение переменной в атрибут №28

При выполнении команды КРУГОЛ ПосВА(XXX) в сервере БД в текущее значение переменной контроллера будет записываться значение из атрибута №65 переменной сервера БД.

ВХОДНАЯ ДИСКРЕТНАЯ ПЕРЕМЕННАЯ

А.1.2 Входная дискретная переменная

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
0			RecordID	целый	4	нд				
1	1	1	Номер канала	целый	2	чт	целый	1	чт/зп	не пере- дается
2	2	2	Номер УСО	целый	2	чт/зп*3	целый	1	чт/зп	не пере- дается
3	3	3	Номер платы	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
4	4	4	Номер входа	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
5	5	5	Позиция	строка	8/24	нд	строка	8	нд	паспорт
6	6	6	Имя 1	строка	8	нд	строка	8	нд	паспорт
7	7	7	Имя 2	строка	8	нд	строка	8	нд	паспорт
8	8	8	Код цвета состояния <0>	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
9	9	9	Код логического состояния <0>	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
10	10	10	Код цвета состояния <1>	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
11	11	11	Код логического состояния <1>	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
12	13	13	Признак необходимости инверсии	лог.	1	чт/зп*2	лог.	1	чт/зп	паспорт
13	14	14	Регистрация перехода из 0 в 1	лог.	1	чт/зп*3	лог.	1	чт/зп	паспорт
14	15	15	Регистрация перехода из 1 в 0	лог.	1	чт/зп*3	лог.	1	чт/зп	паспорт
15	12	12	Номер функции защиты от дребезга	целый	1	чт/зп*3	целый	1	чт/зп	паспорт
16	16	16	Звуковая сигнализация перехода из 1 в 0	лог.	1	чт/зп*3	лог.	1	чт/зп	паспорт
17	17	17	Звуковая сигнализация перехода из 0 в 1	лог.	1	чт/зп*3	лог.	1	чт/зп	паспорт
18	18	18	Тип звуковой сигнализации	целый	1	чт/зп*3	целый	1	чт/зп	паспорт
19	19	19	Свободный атрибут №1	целый	1	чт/зп*3	целый	1	чт/зп	паспорт
20	20	20	Свободный атрибут №2	целый	1	чт/зп*3	целый	1	чт/зп	паспорт
21	21	21	Служебный атрибут №1	целый	1	чт	целый	1	чт	паспорт
22	22	22	Свободный атрибут №3	целый	1	чт/зп*3	целый	1	чт/зп	паспорт
23	23	23	Допустимое время перехода из состояния 0 в 1	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
24	24	24	Допустимое время перехода из состояния 1 в 0	целый	2	чт/зп*3	целый	1	чт/зп	паспорт

ВХОДНАЯ ДИСКРЕТНАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
25	25	25	Снятие переменной с опроса	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. Знач.
26	26	26	Снятие переменной с сигнализации	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. Знач.
27	27	27	Текущее значение переменной	лог.	1	чт/зп*2	лог.	1 бит	чт/зп*1	тек. Знач.
28	28	28	Нарушение <Недостоверность>	лог.	1	чт/зп*3	лог.	1 бит	чт.	Тек. Знач.
29	29	29	Новое нарушение <Недостоверность>	лог.	1	чт/зп*3	лог.	1 бит	чт.	Тек. Знач.
30	30	30	Переход	лог.	1	чт/зп*2	лог.	1 бит	чт.	Тек. Знач.
31	31	31	Превышение времени перехода из 1 в 0	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. Знач.
32	32	32	Превышение времени перехода из 0 в 1	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. Знач.
33	33	33	Лог признак переднего фронта	лог.	1	чт/зп*2	лог.	1 бит	чт	тек. Знач.
34	34	34	Лог признак заднего фронта	лог.	1	чт/зп*2	лог.	1 бит	чт	тек. Знач.
35	35	35	Лог признак <Сигнализация из 1 в 0>	лог.	1	чт/зп*2	лог.	1 бит	чт.	Тек. Знач.
36	36	36	Лог признак <Сигнализация из 0 в 1>	лог.	1	чт/зп*2	лог.	1 бит	чт.	Тек. Знач.
37	37	37	Лог признак <Сигнализация>	лог.	1	чт/зп*2	лог.	1 бит	чт.	Тек. Знач.
38	38	38	Лог признак <Норма>	лог.	1	чт/зп*2	лог.	1 бит	чт.	Тек. Знач.
39	39	39	Лог признак <Новая сигнализация>	лог.	1	чт/зп*2	лог.	1 бит	чт.	Тек. Знач.
40	40	40	Лог признак <Новая норма>	лог.	1	чт/зп*2	лог.	1 бит	чт.	Тек. Знач.
41	41	41	Цвет отображения сигнализации	целый	1	чт/зп*2	целый	1	чт.	Тек. Знач.
42	42	42	Служебный атрибут №2	целый	1	чт/зп*2	целый	1	чт.	Паспорт
		43	Служебный атрибут				целый	1	чт.	
		44	Служебный атрибут				целый	2	чт.	
		45	Служебный атрибут				целый	1	чт.	
43	43		Адрес переменной задания	ссылка	20	нд				
44	44		Значение вх переменной в СО	лог.	1	чт/зп*2				
45	45		Текущее значение в СО	лог.	1	чт/зп*2				
46	46		Длинная позиция	строка	16	нд				
47	47		Длинное имя	строка	64	нд				
48	48		№ алгоритма нестандартной	целый	2	чт/зп*4				

ВХОДНАЯ ДИСКРЕТНАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРБК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
			обработки 1							
49	49		№ алгоритма нестандартной обработки 2	целый	2	чт/зп*4				
50	50		№ алгоритма нестандартной обработки 3	целый	2	чт/зп*4				
51	51		№ алгоритма нестандартной обработки 4	целый	2	чт/зп*4				
52	52		Вкл/выкл нестанд обработок 1	лог.	1	чт/зп*4				
53	53		Вкл/выкл нестанд обработок 2	лог.	1	чт/зп*4				
54	54		Вкл/выкл нестанд обработок 3	лог.	1	чт/зп*4				
55	55		Вкл/выкл нестанд обработок 4	лог.	1	чт/зп*4				
56	56		Основной/резерв1/рез ерв2/	целый	1	чт/зп*3				
57	57		Место обработки	целый	1	чт/зп				
58	58		Место расположения паспорта	целый	1	чт/зп				
59	59		Номер участка (главного)	целый	2	чт/зп				
60	60		Номер канала в комсервере	целый	2	чт/зп				
61	61		Номер м/схемы (главной)	целый	2	чт/зп				
62	62		Метка времени	время	8	нд				
63	63		Снять с опроса в СО	лог.	1	чт/зп				
64	64		Снять с сигнализации в СО	лог.	1	чт/зп				
65	65		Номер УСО в канале	целый	2	чт/зп				
66	66		Тип регистра УСО	целый	2	чт/зп				
67	67		ALARM1	целый	2	чт/зп*2				
68	68		ALARM2	целый	2	чт/зп*2				
69	69		Периодичность обработки (номер списка)	целый	1	чт/зп				
70	70		Цвет состояния	целый	1	чт/зп*2				
71	71		Качество	целый	4	чт/зп				

Примечание:

*1 – атрибут может использоваться для записи только тогда, когда атрибут №3 равен 0.

*2 – атрибут может использоваться для записи только тогда, когда атрибут №1 равен 0 и не включены нестандартные обработки в сервере БД (т.е. атрибуты №77-80 равны 0).

*3 – атрибут может использоваться для записи только тогда, когда атрибут №1 равен 0

ВХОДНАЯ ДИСКРЕТНАЯ ПЕРЕМЕННАЯ

***4** – будьте внимательны, запись в данные атрибуты управляет алгоритмами обработки данной переменной

ВНИМАНИЕ!!!

При включенной нестандартной обработке по переменной для корректной работы алгоритма обработки необходимо из программы пользователя записывать значение переменной в атрибут №27

При выполнении команды КРУГОЛ ПосВД(XXX) в сервере БД в текущее значение переменной контроллера будет записываться значение из атрибута №45 переменной сервера БД.

ДИСКРЕТНАЯ ВЫХОДНАЯ ПЕРЕМЕННАЯ

А.1.3 Дискретная выходная переменная

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длина (байт)	Доступ из КРУГОЛ	
0			RecordID	целый	4	нд.				
1	1	1	Номер канала	целый	2	чт.	Целый	1	чт/зп	не пере- дается
2	2	2	Номер УСО	целый	2	чт/зп*3	целый	1	чт/зп	не пере- дается
3	3	3	Тип УСО	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
4	4	4	Номер платы	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
5	5	5	Номер выхода	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
6	6	6	Позиция	строка	8/24	нд	строка	8	нд	паспорт
7	7	7	Имя 1	строка	8	нд	строка	8	нд	паспорт
8	8	8	Имя 2	строка	8	нд	строка	8	нд	паспорт
9	9	9	Код цвета состояния <0>	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
10	10	10	Код логического состояния <0>	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
11	11	11	Код цвета состояния <1>	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
12	12	12	Код логического состояния <1>	целый	2	чт/зп*3	целый	1	чт/зп	паспорт
13	13	13	Номер переменной в УСО (старший разряд)	целый	1	чт/зп*3	целый	1	чт/зп	паспорт
14	14	14	Признак необходимости инверсии	лог.	1	чт/зп*3	лог.	1	чт/зп	паспорт
15	15	15	Регистрация перехода из 0 в 1	лог.	1	чт/зп*3	лог.	1	чт/зп	паспорт
16	16	16	Регистрация перехода из 1 в 0	лог.	1	чт/зп*3	лог.	1	чт/зп	паспорт
17	17	17	Режим ввода символа	целый	1	чт/зп*3	целый	1	чт/зп	паспорт
18	18	18	Служебный атрибут №1	целый	1	чт/зп*3	Целый	1	чт.	Паспорт
19	19	19	Свободный атрибут №1	целый	1	чт/зп	целый	1	чт/зп	паспорт
20	20	20	Значение выходной переменной в контроллере	лог.	1	чт/зп*3	лог.	1	чт./зп* 1	паспорт
21	21	21	Служебный атрибут №2	целый	1	Чт/зп* 3	целый	1	чт.	Паспорт
22	22	22	Свободный атрибут №2	целый	1	чт/зп*3	целый	1	чт/зп	паспорт
23	23	23	Свободный атрибут №3	целый	1	чт/зп*3	целый	1	чт/зп	паспорт
24		24- 31 ⁵	Служебный атрибут №3	целый	2	чт/зп*3				паспорт
		24*	Служебный атрибут				лог.	1 бит	чт.	Паспорт

ДИСКРЕТНАЯ ВЫХОДНАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длина (байт)	Доступ из КРУГОЛ	
		25*	Служебный атрибут				лог.	1 бит	чт.	Паспорт
		26*	Служебный атрибут				лог.	1 бит	чт.	Паспорт
		27*	Служебный атрибут				лог.	1 бит	чт.	Паспорт
		28*	Служебный атрибут				лог.	1 бит	чт.	Паспорт
		29*	Служебный атрибут				лог.	1 бит	чт.	Паспорт
		30*	Служебный атрибут				лог.	1 бит	чт.	Паспорт
		31*	Служебный атрибут				лог.	1 бит	чт.	Паспорт
25	32	32	Свободный атрибут №4	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. Знач.
26	33	33	Снятие с опроса	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. Знач.
27	34	34	Текущее значение переменной в контроллере	лог.	1	чт/зп*3	лог.	1 бит	чт.	Тек. Знач.
28	35	35	Свободный атрибут №5	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. Знач.
29	36	36	Переход	лог.	1	чт/зп*3	лог.	1 бит	чт.	Тек. Знач.
30	37	37	Свободный атрибут №6	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. Знач.
31	38	38	Передний фронт	лог.	1	чт/зп*3	лог.	1 бит	чт.	Тек. Знач.
32	39	39	Задний фронт	лог.	1	чт/зп*3	лог.	1 бит	чт.	Тек. Знач.
33	40	40	Свободный атрибут №7	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. Знач.
34	41	41	Свободный атрибут №8	лог.	1	чт/зп*3	лог.	1 бит	чт/зп	тек. Знач.
35	42	42	Лог признак <Норма>	лог.	1	чт/зп*3	лог.	1 бит	чт.	Тек. Знач.
36	43	43	Лог признак <Новая норма>	лог.	1	чт/зп*3	лог.	1 бит	чт.	Тек. Знач.
37	44	44	Текущее значение переменной	лог.	1	чт/зп*3	лог.	1 бит	чт	тек. Знач.
38	45	45	Новая сигнализация <Отказ>	лог.	1	чт/зп*3	лог.	1 бит	чт	тек. Знач.
39	46	46	Предыдущее состояние переменной	лог.	1	чт/зп*3	лог.	1 бит	чт.	Тек. Знач.
40	47	47	Сигнализация <Отказ>	лог.	1	чт/зп*3	лог.	1 бит	чт.	Тек. Знач.
41	48	48	Цвет отображения сигнализации	целый	1	чт/зп*3	целый	1	чт.	Тек. Знач.

ДИСКРЕТНАЯ ВЫХОДНАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длина (байт)	Доступ из КРУГОЛ	
42	49	49	Служебный атрибут №4	целый	1	чт/зп*3	целый	1	чт.	Паспорт
		50	Служебный атрибут				целый	1	чт.	Не пере- дается
		51	Служебный атрибут				целый	2	чт.	Не пере- дается
		52	Служебный атрибут				целый	1	чт.	Не пере- дается
		53	Служебный атрибут				целый	2	чт.	Не пере- дается
43	50		Адрес признака аппаратного управления	ссылка	20	нд.				
44	51		Адрес переменной задания	ссылка	20	нд.				
45	52		Значение выходной переменной в СО	лог.	1	чт/зп*3				
46	53		Текущее значение в СО	лог.	1	чт/зп*3				
47	54		Длинная позиция	строка	16	нд				
48	55		Длинное имя	строка	64	нд				
49	56		№ алгоритма нестандартной обработки 1	целый	2	чт/зп*3				
50	57		№ алгоритма нестандартной обработки 2	целый	2	чт/зп*3				
51	58		№ алгоритма нестандартной обработки 3	целый	2	чт/зп*3				
52	59		№ алгоритма нестандартной обработки 4	целый	2	чт/зп*3				
53	60		Вкл/выкл нестанд обработок 1	лог.	1	чт/зп*3				
54	61		Вкл/выкл нестанд обработок 2	лог.	1	чт/зп*3				
55	62		Вкл/выкл нестанд обработок 3	лог.	1	чт/зп*3				
56	63		Вкл/выкл нестанд обработок 4	лог.	1	чт/зп*3				
57	64		Основной/резерв1/ резерв2/	целый	1	чт/зп*3				
58	65		Место обработки	целый	1	чт/зп				
59	66		Место расположения паспорта	целый	1	чт/зп				

ДИСКРЕТНАЯ ВЫХОДНАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длина (байт)	Доступ из КРУГОЛ	
60	67		Номер участка (главного)	целый	2	чт/зп				
61	68		Номер канала в комсервере	целый	2	чт/зп				
62	69		Номер м/схемы (главной)	целый	2	чт/зп				
63	70		Метка времени	время	8	нд				
64	71		Снять с опроса в СО	лог.	1	чт/зп				
65	72		Снять с сигнализации в СО	лог.	1	чт/зп				
66	73		Номер УСО в канале	целый	2	чт/зп				
67	74		Тип регистра УСО	целый	2	чт/зп				
68	75		ALARM1	целый	2	чт/зп*3				
69	76		ALARM2	целый	2	чт/зп*3				
70	77		Периодичность обработки (номер списка)	целый	1	чт/зп				
71	78		Цвет состояния	целый	1	чт/зп*3				
72	79		Качество	целый	4	чт/зп				

Примечание:

*1 - атрибут может использоваться для записи только тогда, когда атрибут №4 равен 0.

*2 - атрибут может использоваться для записи только тогда, когда атрибут №1 равен 0 и не включены нестандартные обработки в сервере БД (т.е. атрибуты №77-80 равны 0).

*3 - атрибут может использоваться для записи только тогда, когда атрибут №1 равен 0

*4 - будьте внимательны, запись в данные атрибуты управляет алгоритмами обработки данной переменной

*5 - Восьми атрибутам в контроллере соответствует один атрибут в станции оператора. Звездочкой * помечена расшифровка названий и типов атрибутов для контроллера.

ВНИМАНИЕ!!!

При выполнении команды КРУГОЛ ПосДВ(ХХХ) в сервере БД в текущее значение переменной контроллера будет записываться значение из атрибута №46 переменной сервера БД.

АНАЛОГОВАЯ ВЫХОДНАЯ ПЕРЕМЕННАЯ

А.1.4 Аналоговая выходная переменная

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
0			RecordID	целый	4	нд				
1	1	1	Номер канала	целый	2	чт	целый	1	чт/зп	не пере дается
2	2	2	Номер УСО	целый	2	чт/зп*4	целый	1	чт/зп	не пере дается
3	3	3	Номер платы	целый	2	чт/зп*4	целый	1	чт/зп	паспорт
4	4	4	Номер выхода	целый	2	чт/зп*4	целый	1	чт/зп	паспорт
5	5	5	Позиция	строка	8/24	нд	строка	8	нд	паспорт
6	6	6	Имя 1	строка	8	нд	строка	8	нд	паспорт
7	7	7	Имя 2	строка	8	нд	строка	8	нд	паспорт
8	8	8,9	Позиция (адрес) переменной 1 (ПОЗ)	ссылка	20	нд				паспорт
		8	Тип переменной ПОЗ				целый	1	чт/зп	паспорт
		9	Номер переменной ПОЗ				целый	2	чт/зп	паспорт
9	10	10,1	Позиция (адрес) задания (ПОЗД)	ссылка	20	нд				паспорт
		10	Тип переменной ПОЗД				целый	1	чт/зп	паспорт
		11	Номер переменной ПОЗД				целый	2	чт/зп	паспорт
10	12	12,1	Позиция (адрес) упр Воздействия (ПУВ)	ссылка	20	нд				паспорт
		12	Тип переменной ПУВ				целый	1	чт/зп	паспорт
		13	Номер переменной ПУВ				целый	2	чт/зп	паспорт
11	14	14,1	Позиция (адрес) признака ап Упр (ПАУ)	ссылка	20	нд				паспорт
		14	Тип переменной ПАУ				целый	1	чт/зп	паспорт
		15	Номер переменной ПАУ				целый	2	чт/зп	паспорт
12	16	16,1	Позиция (адрес) переменной 2 (ВИМ)	ссылка	20	нд				паспорт
		16	Тип переменной ВИМ				целый	1	чт/зп	паспорт
		17	Номер переменной ВИМ				целый	2	чт/зп	паспорт
13	18	18,1	Позиция (адрес) переменной 3 (ИМВ)	ссылка	20	нд				паспорт
		18	Тип переменной ИМВ				целый	1	чт/зп	паспорт
		19	Номер переменной ИМВ				целый	2	чт/зп	паспорт
14	20	20	Тип регулятора (аналоговый, импульсный)	целый	1	чт/зп*4	целый	1	чт/зп	паспорт
15	21	21	Величина задания	вещ.	4	чт/зп*4	вещ.	4	чт/зп*1	тек. знач.
16	22	22	Значение ручного Задатчика /	вещ.	4	чт/зп*4	вещ.	4	чт/зп*2	тек. знач.

АНАЛОГОВАЯ ВЫХОДНАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
			Положение МЭО							
17	23	23	Величина коэффиц Пропорцион	вещ.	4	чт/зп*4	вещ.	4	чт/зп	тек. знач.
18	24	24	Постоянная времени интегриров	вещ.	4	чт/зп*4	вещ.	4	чт/зп	тек. знач.
19	25	25	Величина коэффиц Дифференц	вещ.	4	чт/зп*4	вещ.	4	чт/зп	тек. знач.
20	26	26	Зона нечувствительности	вещ.	4	чт/зп*4	вещ.	4	чт/зп	тек. знач.
21	27	27	Коэффиц ослабл для зоны нечувств	вещ.	4	чт/зп*4	вещ.	4	чт/зп	паспорт
22	28	28	Верхняя граница отклонения от задания	вещ.	4	чт/зп*4	вещ.	4	чт/зп	паспорт
23	29	29	Нижняя граница отклонения от задания	вещ.	4	чт/зп*4	вещ.	4	чт/зп	паспорт
24	30	30	Верхнее огранич хода ИМ	вещ.	4	чт/зп*4	вещ.	4	чт/зп	тек. знач.
25	31	31	Нижнее огранич хода ИМ/ПР управление ИМ	вещ.	4	чт/зп*4	вещ.	4	чт/зп	тек. знач.
26	32	32	Верхняя граница сигнализ хода ИМ/люфт <Больше>	вещ.	4	чт/зп*4	вещ.	4	чт/зп	паспорт
27	33	33	Нижняя граница сигнализ хода ИМ/люфт <Меньше>	вещ.	4	чт/зп*4	вещ.	4	чт/зп	паспорт
28	34	34	Тактовая частота / длительность импульса	вещ.	4	чт/зп*4	вещ.	4	чт/зп	тек. знач.
29	35	35	Переход к новому заданию	целый	1	чт/зп*4	целый	1	чт/зп	паспорт
30	36	36	Постоянная времени по заданию	целый	4	чт/зп*4	целый	2	чт/зп	паспорт
31	37	37	Коэффициент для форсир перехода	вещ.	4	чт/зп*4	вещ.	4	чт/зп	паспорт
32	38	38	Скорость хода ИМ / время полного хода ИМ	вещ.	4	чт/зп*4	вещ.	4	чт/зп	тек. знач.
33	39	39	Текущее значение параметра	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	тек. знач.
34	40	40	Текущее значение параметра (демасштаб)	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	паспорт
35	41	41	Текущее значение задания (демасштаб)	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	паспорт
36	42	42	Предыдущее задание, отличное от текущего	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	паспорт
37	43	43	Промежуточная величина задания	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	паспорт

КРУГОЛ™. ИНТЕГРИРОВАННАЯ СРЕДА РАЗРАБОТКИ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
38	44	44	Рассогласование	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	тек. знач.
39	45	45	Рассогласование предыдущее (i-1)	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	паспорт
40	46	46	Рассогласование предыдущее (i-2)	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	паспорт
41			Рассогласование предыдущее (i-3)	вещ.	4	нд.				
42			Рассогласование предыдущее (i-4)	вещ.	4	нд				
43			Рассогласование предыдущее (i-5)	вещ.	4	нд				
44	47	47	Промежуточное значение вых сигнала	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	паспорт
45	48	48	Значение выходного сигнала (для аналог р- ра)	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	тек. знач.
46	49	49	Значение при ручном дистанц управлении	вещ.	4	чт/зп*4	вещ.	4	чт/зп	тек. знач.
47	50	50	Текущий режим управления (ТРУ)	строка	8	нд	строка	2	нд	тек. знач.
48	51	51	Текущий режим ввода задания (ТРВЗ)	строка	8	нд	строка	2	нд	тек. знач.
49	52	52	Дополнит сигнал на ИМ (с учетом рассогл)	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	паспорт
50	53	53	Предыдущ значение вых сигнала	вещ.	4	чт/зп*4	вещ.	4	чт/зп*5	паспорт
51	54	54	Интеграл рассогласования	вещ.	4	чт/зп*4	вещ.	4	чт/зп	паспорт
52	55	55	Служебный атрибут №1	целый	1	чт./зп* 4	целый	1	чт/зп*5	паспорт
53	56	56	Служебный атрибут №2	целый	1	чт/зп*4	целый	1	чт/зп*5	паспорт
54	57	57	Служебный атрибут №3	целый	1	чт/зп*4	целый	1	чт/зп*5	паспорт
55	58	58	Служебный атрибут №4	целый	1	чт/зп*4	целый	1	чт/зп*5	паспорт
56	59	59	Отклонение от верхней границы задания	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
57	60	60	Новое отклонение от верхней границы задания	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
58	61	61	Отклонение от нижней границы задания	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
59	62	62	Новое отклонение от нижней границы задания	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
60	63	63	Нарушение по верхней границе сигн хода ИМ	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.

АНАЛОГОВАЯ ВЫХОДНАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
61	64	64	Новое нарушение по верхней границе сигн хода ИМ	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
62	65	65	Нарушение по нижней границе сигн хода ИМ	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
63	66	66	Новое нарушение по нижней границе сигн хода ИМ	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
64	67	67	Запрет перехода в ДУ по недоверности	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
65	68	68	Инверсия выходного сигнала	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
66	69	69	Диагностика ЦАП	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
67	70	70	Диагностика ЦАП (новая)	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
68	71	71	Признак обрыва цепи	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
69	72	72	Признак перехода на ДУ	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
70	73	73	Служебный атрибут №5	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
71	74	74	Служебный атрибут №6	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
72	75	75	Внешний сигнал <Б>	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
73	76	76	Внешний сигнал <М>	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
74	77	77	Вид действия (1-прямой / 0-обратный)	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
75	78	78	Вид действия ИМ (НЗ - 1/НО - 0) (для аналог р-ра)	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
76	79	79	Новый переход на ДУ	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.
77	80	80	Признак точного управления	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
78	81	81	Отключение <Б> от регулятора	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
79	82	82	Отключение <М> от регулятора	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
80	83	83	Снятие с сигнализации по заданию	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
81	84	84	Снятие с сигнализации по положению ИМ	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
82	85	85	Дистанция Б	лог.	1	чт/зп*4	лог.	1 бит	чт/зп	тек. знач.
83	86	86	<Больше> с регулятора	лог.	1	чт/зп*4	лог.	1 бит	чт/зп*5	тек. знач.

КРУГОЛ™. ИНТЕГРИРОВАННАЯ СРЕДА РАЗРАБОТКИ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длина (байт)	Доступ из КРУГОЛ	
84	87	87	Дистанция М	лог	1	чт/зп*4	лог	1 бит	чт/зп	тек. знач.
85	88	88	<Меньше> с регулятора	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
86	89	89	<Б> на ПБР	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
87	90	90	<М> на ПБР	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
88	91	91	Режим регулятора <Руч аппаратный>	лог	1	чт/зп*4	лог	1 бит	чт/зп*3	тек. знач.
89	92	92	Режим регулятора <Руч дистанционный>	лог	1	чт/зп*4	лог	1 бит	чт/зп	тек. знач.
90	93	93	Режим регулятора <Автоматический>	лог	1	чт/зп*4	лог	1 бит	чт	тек. знач.
91	94	94	Служебный атрибут №7	лог	1	чт/зп4	лог	1 бит	чт/зп*5	тек. знач.
92	95	95	Режим ввода задания <Ручной>	лог	1	чт/зп*4	лог	1 бит	чт/зп	тек. знач.
93	96	96	Режим ввода задания <Внешний> (<Каскад>)	лог	1	чт/зп*4	лог	1 бит	чт/зп	тек. знач.
94	97	97	Служебный атрибут №8	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
95	98	98	Служебный атрибут №9	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
96	99	99	Служебный атрибут №10	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
97	100	100	Служебный атрибут №11	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
98	101	101	Служебный атрибут №12	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
99	102	102	Служебный атрибут №13	лог	1	Чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
100	103	103	Служебный атрибут №14	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
101	104	104	Служебный атрибут №15	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
102	105	105	Служебный атрибут №16	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
103	106	106	Служебный атрибут №17	лог	1	чт/зп*4	лог	1 бит	чт/зп*5	тек. знач.
104	107	107	Цвет отображения сигнализации	целый	1	чт/зп*4	целый	1	чт/зп*5	тек. знач.
105	108	108	Служебный атрибут №18	целый	1	чт/зп*4	целый	1	чт/зп*5	паспорт
106			Служебный атрибут №19	целый	1	нд				
107	109		Тек значение 1 переменной (на СО)	вещ.	4	чт/зп*4				
108	110		Величина задания (на СО)	вещ.	4	чт/зп*4				
109	111		Рассогласование (на	вещ.	4	чт/зп*4				

АНАЛОГОВАЯ ВЫХОДНАЯ ПЕРЕМЕННАЯ

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
			СО)							
110	112		Значение вых сигнала (на СО)	вещ.	4	чт/зп*4				
111	113		Величина коэф пропорцион (Кп)(на СО)	вещ.	4	чт/зп*4				
112	114		Пост времени интегриров (Т)(на СО)	вещ.	4	чт/зп*4				
113	115		Величина коэф дифференц (Кд)(на СО)	вещ.	4	чт/зп*4				
114	116		Зона нечувствительности (НЕЧ)(на СО)	вещ.	4	чт/зп*4				
115	117		Знач при ручном дистанц (ИМР)(на СО)	вещ.	4	чт/зп*4				
116	118		Верх огранич хода ИМ (МАКС)(на СО)	вещ.	4	чт/зп*4				
117	119		Нижнее огранич хода ИМ (МИН)(на СО)	вещ.	4	чт/зп*4				
118	120		Тактовая частота (ТКТ) (на СО)	вещ.	4	чт/зп*4				
119	121		Знач ручного задатчика (ЗПМ)(на СО)	вещ.	4	чт/зп*4				
120	122		Скорость хода ИМ (СИМ)(на СО)	вещ.	4	чт/зп*4				
121	123		Тек значение 2 переменной (на СО)	вещ.	4	чт/зп*4				
122	124		Тек значение 3 переменной (на СО)	вещ.	4	чт/зп*4				
123	125		Периодичность обработки (номер списка)	целый	1	чт/зп				
124	126		Длинная позиция	строка	16	нд				
125	127		Длинное имя	строка	64	нд				
126	128		№ алгоритма нестандарт обработки 1	целый	2	чт/зп				
127	129		№ алгоритма нестандарт обработки 2	целый	2	чт/зп				
128	130		№ алгоритма нестандарт обработки 3	целый	2	чт/зп				
129	131		№ алгоритма нестандарт обработки 4	целый	2	чт/зп				
130	132		Вкл/выкл нестанд обработок 1	лог	1	чт/зп				
131	133		Вкл/выкл нестанд	лог	1	чт/зп				

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРБК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
			обработок 2							
132	134		Вкл/выкл нестандарт обработок 3	лог	1	чт/зп				
133	135		Вкл/выкл нестандарт обработок 4	лог	1	чт/зп				
134	136		Основной/Резерв1/Ре зерв2/	целый	1	чт/зп*4				
135	137		Место обработки	целый	1	чт/зп				
136	138		Место расположения паспорта	целый	1	чт/зп				
137	139		Номер участка (главного)	целый	2	чт/зп				
138	140		Номер канала в комсервере	целый	2	чт/зп				
139	141		Номер м/схемы (главной)	целый	2	чт/зп				
140	142		Метка времени	время	8	нд				
141	143		Снять с опроса в СО	лог	1	чт/зп				
142	144		Снять с сигнализации в СО	лог	1	чт/зп				
143	145		Номер УСО в канале	целый	2	чт/зп				
144	146		Тип регистра УСО	целый	2	чт/зп				
145	147		ALARM1	целый	2	чт				
146	148		ALARM2	целый	2	чт				
147	149		Цвет состояния	целый	1	чт/зп*4				
148	150		Качество	целый	4	чт/зп				

Примечания:

- *1. Атрибут доступен для записи только тогда, когда атрибут №96 (Режим ввода задания «Внешний») равен 0.
- *2. Атрибут доступен для записи только тогда, когда в Генераторе базы данных по этой переменной не указан **адрес управляющего воздействия** (атрибуты №12 и 13).
- *3. Атрибут доступен для записи только тогда, когда в Генераторе базы данных по этой переменной не указан **адрес признака аппаратного управления** (атрибуты №14 и 15).
- *4. Атрибут доступен для записи только тогда, когда атрибут №1 равен 0.
- *5. Атрибут доступен для записи только для типа регулятора 100 (аналоговый).

ВНИМАНИЕ!!!

При выполнении команды КРУГОЛ ПосАВ(XXX) в сервере БД в значение выходного сигнала контроллера (атрибут №48) будет записываться значение из атрибута №117 переменной сервера БД, если значение атрибута №92 в контроллере =1, в противном случае – будет записываться в атрибут контроллера №49.

ПЕРЕМЕННАЯ РУЧНОГО ВВОДА

А.1.5 Переменная ручного ввода

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРВК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
0			RecordID	целый	4	нд				
1			Номер канала	целый	2	нд				
2			Номер переменной в УСО	целый	2	нд				
3	1	1	Позиция	строка	8/24	нд	строка	8	нд	паспорт
4	2	2	Имя 1	строка	8	нд	строка	8	нд	паспорт
5	3	3	Имя 2	строка	8	нд	строка	8	нд	паспорт
6	4	4	Тип переменной	целый	1	чт/зп*4	целый	1	чт	паспорт
7	5	5	Единица измерения	целый	1	чт/зп*4	целый	1	чт/зп *1	паспорт
8	6	6	Максимум	вещ.	4	чт/зп*4	вещ	4	чт/зп *1	паспорт
9	7	7	Минимум	вещ.	4	чт/зп*4	вещ	4	чт/зп *1	паспорт
10	8		Начальное значение	вещ.	4	чт/зп*4				тек. знач
11	27	8	Текущее значение	вещ.	4	чт/зп*4	вещ	4	чт/зп *1	тек. знач.
12	9	9	Текущее значение 2	вещ.	4	чт/зп	вещ	4	чт/зп *1	не передает ся
13	10	10	Свободный атрибут №1	целый	1	нд	целый	1	нд	тек. знач.
14	11	11	Служебный атрибут №1	целый	1	чт/зп*4	целый	1	чт	паспорт
15	12		Начальное значение (Строка)	строка	16	нд				тек. знач
16	13	12	Текущее значение (Строка)	строка	16	нд	строка	16	нд*2	тек. знач.
17	38		Текущее значение (Длинная строка)	строка	63	чт/зп				
18	16	13	Цвет отображения строки	целый	1	чт/зп*4	целый	1	чт/зп *2	паспорт
19	15		Начальное значение (логич)	лог	1	чт/зп				тек. знач
20	14	14	Текущее значение (логич)	лог	1	чт/зп*4	лог	1	чт/зп *3	тек. знач
21	17	16	Код цвета состояния <0>	целый	1	чт/зп*4	целый	1	чт/зп *3	паспорт
22	18	15	Код логического состояния <0>	целый	1	чт/зп*4	целый	1	чт/зп. *3	паспорт
23	19	18	Код цвета состояния <1>	целый	1	чт/зп*4	целый	1	чт/зп. *3	паспорт
24	20	17	Код логического состояния <1>	целый	1	чт/зп*4	целый	1	чт/зп *3	паспорт
25	21	19	Служебный атрибут №2	целый	1	чт/зп*4	целый	1	чт/зп *3	паспорт
26	22		Начальное значение 2	вещ.	4	чт/зп				

№ в БДСО (СБД)	№ в СО	№ в контр	НАИМЕНОВАНИЕ АТРИБУТА	SCADA «КРУГ-2000» (СО)			СРБК «КРУГ» (Контроллер)			Режим обмена данным
				Тип	Длина (байт)	Доступ из КРУГОЛ	Тип	Длин а (байт)	Доступ из КРУГОЛ	
			(вещественное)							
27	23		Основной/резерв1/резерв2/	целый	1	чт/зп				
28	24		Место расположения паспорта	целый	1	чт/зп				
29	25		Номер участка (главного)	целый	2	чт/зп				
30	26		Номер группы (главной)	целый	2	чт/зп				
31	28		Номер м/схемы (главной)	целый	2	чт/зп				
32	29		Метка времени	время	8	нд				
33	30		Снять с опроса в СО	лог.	1	чт/зп				
34	31		ALARM1	целый	2	чт/зп*4				
35	32		ALARM2	целый	2	чт/зп*4				
36	33		Цвет состояния	целый	1	чт/зп*4				
37	37		Длинное имя	строка	64	нд				
38	39		Качество	целый	4	чт/зп				
39	40		Текущее значение двойной точности	вещ	8	чт/зп				

Примечание:

*1 - Атрибуты для переменных вещественного формата

*2 - Атрибуты для переменных строкового формата

*3 - Атрибуты для переменных логического формата

*4 - Атрибут доступен для записи только тогда, когда атрибут №1 равен 0.

В. ПРИМЕРЫ ПРОГРАММ СТ. ПРИЛОЖЕНИЕ В (Справочное).**В.1 Примеры****В.1.1 Пример 1. Формирование сообщений в протокол событий**

В программе показано использование функции **message** для формирования сообщений

Программа РОЛЛИНГ

Начало

: Программа - пример

: Формирование сообщений в "ПРОТОКОЛ СОБЫТИЙ"

: Пример 1. Регистрация события из ПрП

: вд2 - дискретный сигнал от датчика пожара "0" - норма, "1" - пожар

: пл2 - промежуточная переменная - флаг формирования сообщения

Если пл2 = 0

{ Если вд2 = 1 : Если сработал датчик пожара

{

пл2 = 1

message(" ПОЖАР В ОПЕРАТОРНОЙ ") : Выдать сообщение в протокол

}

}

Если вд2 = 0 : Если датчик пожара "НОРМА"

{

пл2 = 0

}

: Пример 2. Регистрация времени нахождения параметра

: за предупредительными границами

: ВА1 - контролируемый параметр

: ВА1.а14 - верхняя предупредительная граница (ВПГ)

: ВА1.а13 - нижняя предупредительная граница (НПГ)

: тс1, тс2 - секундный счетчик

Если ВА1 > ВА1.а14

{

Вкл тс1

пл3 = 1

}

Иначе

{

Если пл3 = 1

{

message(" Давление газа. Нарушение ВПГ, сек", тс1)

Выкл тс1

тс1 = 0

пл3 = 0

}

}

Если ВА1 < ВА1.а13

{

Вкл тс2

пл4 = 1

}

```

Иначе
{
  Если пл4 = 1
  {
    message(" Давление газа. Нарушение НПГ, сек .....",тс2)
    Выкл тс2
    тс2 = 0
    пл4 = 0
  }
}
Конец

```

В.1.2 Пример 2. Реализация расчетной задачи

Реализация расчетной задачи рассмотрена на примере вычисления расхода газа по перепаду давления на сужающем устройстве, температуре и давлению в трубопроводе.

Программа Расход

Начало

: Программа - пример

: Реализация расчетной задачи "Расчет расхода газа"

:

: ВА1[АР] - давление газа в трубопроводе, Мпа (P)

: ВА2[АТ] - температура газа в трубопроводе, град. С (Т)

: ВА3[АdР] - перепад на сужающем устройстве, кг/кв.см (dP)

: ВА4[АF] - расход газа, куб.м/час (F)

: пв10 - масштабный коэффициент (константа) (K)

: пв11, ВА4[АF].а12 - конец шкалы, куб.м/час (Кш)

$$: F = (((P + dP * 1.033 * K) / (T + 273.2))^{0.5}) * Kш \quad (1)$$

Для AdP[1,5],

AP[2,6],

AT[3,7],

AF[4,8]

{

пв11 = ва[АF].а12 : Определяем конец шкалы

Если ва[АF].а12 = 50000 : Определяем масштабный коэффициент

{ пв10 = 96.29 } : в зависимости

Если ва[АF].а12 = 40000 : от типа

{ пв10 = 59.94 } : сужающего устройства,

Если ва[АF].а12 = 32000 : определяемый

{ пв10 = 39.94 } : (в данной задаче)

Если ва[АF].а12 = 20000 : концом шкалы

{ пв10 = 15.41 } :

$$пв12 = (ва[АР] + ва[АdР] * 1.033 * пв10) / (ва[АТ] + 273.2)$$

$$ва[АF] = \text{sqr}(пв12) * пв11$$

}

Конец

В.1.3 Пример 3. Имитация входного аналогового параметра

В некоторых случаях, например при разработке демо-системы, необходима имитация значений аналоговых параметров. В данном примере параметр изменяется синусоидально. Таким способом можно реализовать практически любые, как линейные, так и нелинейные зависимости.

Программа Имитатор
Начало

: Программа - пример
: Имитация "ВХОДНОГО АНАЛОГОВОГО" сигнала
: без выхода за границы шкалы
: (в данном примере $F = \sin(x)$, $x = 0 \dots 360$ град)

: Задаем аргумент синусу $0 \dots 360$ град
: В качестве аргумента используем секундный счетчик

Вкл tc1

Если tc1 > 360
{ tc1 = 0 }

: Параметр можно имитировать индивидуально (Пример1),
: а так же группой, используя оператор "ДЛЯ" (Пример1)
: BA1.a11 - Начало шкалы
: BA1.a12 - Конец шкалы
: $BA = Hш + (Kш - Hш)/2 + (Kш - Hш)*\sin(x)/2$

: Пример 1

$BA1 = BA1.a11 + (BA1.a12 - BA1.a11)*0.5 + (BA1.a12 - BA1.a11)*\sin(tc1)*0.5$

: Пример 2

Для spVA[2...8]
{
 $BA[spVA] = BA[spVA].a11 + (BA[spVA].a12 - BA[spVA].a11)*0.5 + (BA[spVA].a12 - BA[spVA].a11)*\cos(tc1)*0.5$
}
Конец

В.1.4 Пример 4. Противоаварийные защиты и блокировки

Алгоритмы противоаварийных защит и блокировок (ПАЗ) описывают порядок работы технологического оборудования при возникновении различных аварийных ситуаций. В данном приложении рассмотрены простейшие примеры реализации алгоритмов такого типа.

Программа ПАЗ

Начало

: Программа - пример

: Реализация Противоаварийных защит (ПАЗ) и блокировок

: Защита оборудования насосной при пожаре и загазованности

ПОЖАР

: Защита резервуаров по минимуму

РЕЗЕРВУАР

Выход

Конец

ПРОЦЕДУРА ПОЖАР

НАЧАЛО

: Защита оборудования насосной

: При пожаре или загазованности в насосной

: необходимо аварийно отключить все насосы

:

: вд2 - Пожар в насосной

: вд3 - Загазованность II порог в насосной

: пл60 - Флаг аварийной ситуации в насосной

: пл61 - промежуточная переменная

: дв[stA] - дискретный сигнал отключения насоса при аварии

:

пл60 = вд2|вд3 : Если пожар или загазованность в насосной

Если пл60 = 1

{

Если пл61 = 0

{

Для stA[1,2,3,4]

{

дв[stA] = 1 : Выключить все насосы при аварии

}

пл61 = 1

message("ОСТАНОВ НАСОСНОЙ") }}

Иначе

{ пл61 = 0 }

КОНЕЦ

ПРОЦЕДУРА РЕЗЕРВУАР

НАЧАЛО

```
: Защита резервуаров 1...16 по минимуму
: При достижении минимума в резервуаре
: закрыть выходную задвижку
:
: ва[Lmin] - Датчик уровня резервуара
: ва[Lmin].a15 - Нижняя аварийная граница (НАГ) - уставка блокировки
: дв[Azadv] - сигнал управления "закрыть" задвижку ("1")
: пл[Aflag] - флаг блокировки
: пц[Aflag] - промежуточная переменная для хранения № резервуара (1,2 ...)
```

Для Lmin[1...16],

Azadv[5...20], : или [6...21], диапазоны должны быть одинаковы по размеру, иначе будет ошибка транслятора.

Aflag[101...116]

```
{
  Если пл[Aflag] = 0
  {
    Если ва[Lmin] < ва[Lmin].a15 : Если уровень ниже НАГ
    {
      пл[Aflag] = 1
      message("БЛОКИРОВКА L min РЕЗЕРВУАР № ",пц[Aflag])
```

```
      дв[Azadv] = 1 : Закрыть задвижку
```

```
      : Сброс сигнала управления происходит из алгоритма управления
      : соответствующей задвижки
    }
  }
}
```

Если ва[Lmin] > ва[Lmin].a15 : Если уровень вернулся в норму

```
{
  пл[Aflag] = 0
}
}
```

КОНЕЦ

В.1.5 Пример 5. Обмен паспортов переменных между абонентами ПТК

Для программного изменения паспортов переменных в ИСР «КРУГОЛ» существуют специальные функции ПосВА, ПосВД, ПосДВ, ПосРВ. Эти функции позволяют в приоритетном порядке осуществить обмен паспортами переменных контроллера и Станции оператора (Сервера). Пример использования данных функций приведен в программе «PASPORT».

Программа PASPORT

Начало

: Пример изменения значения различных атрибутов и

: текущих значений контроллерных переменных со станции оператора (СО)

Если рв91 = 1 : Управление тестовым примером из СО

```
{
  рв91 = 0
: Пример для переменной типа "РУЧНОЙ ВВОД"
РВ7 = 4.55 : Изменяем текущее значение переменной
РВ7.а6 = 9.5 : или один из ее атрибутов (в данном случае "максимум")
              : чтобы изменения попали в контроллер
Если пл7 = 0
{
  посРВ(7) : посылаем паспорт переменной в контроллер
  пл7 = 1 : Оператор условия ЕСЛИ позволяет выполнить эту операцию
}
: единожды. Это необходимо, так как ф-я посХХ имеет
: наивысший приоритет и в случае если этих ф-й много
: и вызываются они при каждом проходе программы, то
: обмен между СО и контроллером будет затруднен
: (они будут только и делать, что обмениваться между собой)
: При следующем изменении атрибутов РВ вниз по тексту необходимо "обнулить"
: флаг пл7 и паспорт уйдет вновь в контроллер
:
```

: Пример для переменной типа "ВХОДНАЯ ДИСКРЕТНАЯ" и

: "ДИСКРЕТНАЯ ВЫХОДНАЯ"

вд2 = 1 : Изменяем текущее значение переменных

дв1 = 1 : вд должна быть "виртуальной"

Если пл6 = 0

```
{
  посВД(2) : посылаем паспорта переменных в контроллер
  посДВ(521) : "2" и "521" номер переменной в БД СО, а не УСО
  пл6 = 1 : аналогично взводим флаг запрещающий посылку
}
```

: Пример для переменной типа "ВХОДНАЯ АНАЛОГОВАЯ"

: В данной версии мы можем изменять все доступные атрибуты

: напрямую кроме текущего значения переменной (ТЗ)

ва10.а12 = 50 : Например изменяем конец шкалы

Если пл5 = 0

```
{
  посВА(10) : Аналогично посылаем паспорт
  пл5 = 1
}
```

: Для контроллера операции аналогичны, только в параметре функции посХХ нужно

: передавать номер переменной в УСО

}

КОНЕЦ

В.1.6 Пример 6. Управление задвижкой с электроприводом

Данный алгоритм описывает порядок работы управляющего контроллера при обслуживании электрозадвижек с двумя командами **управления «Открыть задвижку»** и **«Закрыть задвижку»** и двумя состояниями задвижки: **«Открыта»** и **«Закрыта»**. Задвижки не имеют сигнала о заклинивании, и поэтому их неисправность может быть определена только по времени выполнения команд управления.

Алгоритмы не исключают возможности как местного, так и дистанционного управления задвижками и обеспечивают выполнение процедуры диагностики исправности задвижки путем проверки времени перехода задвижки из одного крайнего положения в другое крайнее положение.

Для работы алгоритмов используются следующие сигналы и информация:

- дискретный сигнал «Наличие напряжения в цепи управления задвижкой»
- внутренний параметр системы «Задвижка неисправна»
- внутренний параметр системы «Задвижка заблокирована»
- внутренний параметр системы «Задвижка открывается»
- внутренний параметр системы «Задвижка закрывается»
- дискретный сигнал с исполнительного механизма управления задвижкой «Задвижка открыта»
- дискретный сигнал с исполнительного механизма управления задвижкой «Задвижка закрыта».

В результате выполнения алгоритмов формируются управляющие сигналы и информация о текущем состоянии задвижки:

- дискретный сигнал «Открыть задвижку »
- дискретный сигнал «Закрыть задвижку »
- дискретный сигнал «Ручной режим управления»
- дискретный сигнал «Дистанционный режим управления ».

Алгоритм контроля состояния задвижки. При изменении состояния сигналов, получаемых с задвижки, осуществляется выдача соответствующих сообщений. Система информирует оператора о состоянии задвижки и устанавливает внутренние программные флаги по следующим критериям:

- наличие напряжения в цепи управления задвижкой
- задвижка не открылась (не закрылась)
- неисправность (при неопределённом сигнале, снимаемом с задвижки в течение времени, большем необходимого для осуществления операции полного открытия или полного закрытия, а так же при получении недостоверного сигнала)
- задвижка открывается (закрывается).

Алгоритм выполняется по следующей схеме:

1. Происходит проверка изменения состояния наличия напряжения в цепи управления задвижкой. В случае изменения сигнала о наличии напряжения система фиксирует это соответствующим протоколом и информирует оператора текстовым сообщением
2. Происходит выявление изменения состояния задвижки.

- При переходе состояния задвижки из крайнего закрытого положения в неопределённое система даёт временную выдержку, необходимую для осуществления операции полного открытия. Во время открытия система устанавливает параметр задвижки – **задвижка открывается**. В случае не открытия задвижки, при истечении срока временной выдержки,

здвижка идентифицируется как **неисправная** и дальнейшее управление последней становится невозможно. После устранения неисправности вышеуказанный параметр возвращается в норму при достижении задвижкой любого конечного положения

- При переходе состояния задвижки из крайнего открытого положения в неопределённое система даёт временную выдержку, необходимую для осуществления операции полного закрытия. Во время выполнения процедуры закрытия система устанавливает параметр задвижки – **здвижка закрывается**. Аналогично, в случае не закрытия задвижки, при истечении срока временной выдержки, задвижка идентифицируется как **неисправная** и дальнейшее управление последней становится невозможно. После устранения неисправности вышеуказанный параметр возвращается в норму при достижении задвижкой любого конечного положения

Алгоритм открыть задвижку предназначен для выполнения системой операции дистанционного открытия задвижки. Алгоритм выполняется следующим образом: Производится проверка состояния данной задвижки. В процессе проверки состояния задвижки выясняется следующее:

- проверяется флаг исправности указанной задвижки (в случае если задвижка оказывается неисправной, оператор получает соответствующее сообщение и выполнение алгоритма при этом завершается операцией протоколирования)
- проверяются состояние, в котором в данный момент находится указанная задвижка: открыта, закрыта, открывается, закрывается (в случае если задвижка оказывается не закрытой, оператор получает соответствующее сообщение и выполнение алгоритма при этом завершается операцией протоколирования)
- проверяется флаг блокировки управления задвижкой. При его наличие оператор получает соответствующее сообщение, и выполнение алгоритма на этом заканчивается операцией протоколирования, как и в предыдущем случае
- проверяется флаг управления задвижкой. При наличии флага ручного управления задвижкой выполнение алгоритма на этом заканчивается операцией протоколирования, как и в предыдущем случае
- проверяется наличие напряжения в цепи управления задвижкой. При его отсутствии оператор получает соответствующее сообщение, и выполнения алгоритма на этом заканчивается операцией протоколирования, как и в предыдущем случае
- выдаётся сигнал на открытие задвижки
- определяется временная выдержка (необходимая для полного открытия задвижки по регламенту) в течение которой проверяется условие открытия задвижки. При достижении указанного условия сигнал, выданный на открытие задвижки, снимается и происходит протоколирование по окончании алгоритма. Информацию о том, что задвижка открылась, оператор получает из алгоритма контроля состояния задвижки. При не открытии задвижки за обозначенный промежуток времени так же происходит снятие сигнала, выданного на открытие задвижки, оператор получает информационное сообщение о том, что алгоритм открытия задвижки не выполнен и происходит протоколирование. Протоколирование событий по выходу из алгоритма производится аналогично протоколированию событий в начале выполнения алгоритма, как было рассмотрено выше. Кроме того, при данном протоколировании система записывает отчет о результате выполнения алгоритма: был ли алгоритм выполнен с положительным результатом – действие, на которое направлен соответствующий алгоритм, совершилось (например, задвижка открылась) или алгоритм был выполнен с отрицательным результатом – действие, на которое был направлен соответствующий алгоритм, не совершилось.

Алгоритм закрыть задвижку предназначен для выполнения системой операции дистанционного закрытия задвижки. Алгоритм выполняется следующим образом. Первым шагом происходит протоколирование события, когда и кем был запущен алгоритм. Далее

производится проверка состояния данной задвижки. В процессе проверки состояния задвижки выясняется следующее:

- проверяется флаг исправности указанной задвижки (в случае если задвижка оказывается неисправной, оператор получает соответствующее сообщение и выполнение алгоритма при этом завершается операцией протоколирования)
 - проверяются состояние, в котором в данный момент находится указанная задвижка: открыта, закрыта, открывается, закрывается (в случае если задвижка оказывается не открытой, оператор получает соответствующее сообщение и выполнение алгоритма при этом завершается операцией протоколирования)
 - проверяется флаг блокировки управления задвижкой. При его наличии оператор получает соответствующее сообщение, и выполнение алгоритма на этом заканчивается операцией протоколирования, как и в предыдущем случае
 - проверяется флаг управления задвижкой. При наличии флага ручного управления задвижкой выполнение алгоритма на этом заканчивается операцией протоколирования, как и в предыдущем случае
 - проверяется наличие напряжения в цепи управления задвижкой. При его отсутствии оператор получает соответствующее сообщение, и выполнение алгоритма на этом заканчивается операцией протоколирования, как и в предыдущем случае
 - выдаётся сигнал на закрытие задвижки
 - определяется временная выдержка (необходимая для полного закрытия задвижки по регламенту) в течение которой проверяется условие закрытия задвижки. При достижении указанного условия сигнал, выданный на закрытие задвижки, снимается и происходит протоколирование по окончании алгоритма. Информацию о том, что задвижка закрылась, оператор получает из алгоритма контроля состояния задвижки. При не закрытии задвижки за обозначенный промежуток времени так же происходит снятие сигнала, выданного на закрытие задвижки, оператор получает информационное сообщение о том, что алгоритм закрытия задвижки не выполнен и происходит протоколирование.
- **Алгоритм закрыть задвижку при аварии** предназначен для выполнения системой операции дистанционного закрытия задвижки в аварийной ситуации. Алгоритм выполняется следующим образом.
- первым шагом происходит протоколирование события, когда и кем был запущен алгоритм
 - независимо от состояния задвижки снимается сигнал открытия (если он был ранее установлен)
 - задвижка переводится в режим дистанционного управления (если до этого она находилась в ручном режиме)
 - устанавливается блокировка на управление задвижкой, как в ручном, так и дистанционном режимах
 - выдаётся сигнал на аварийное закрытие задвижки
 - определяется временная выдержка (необходимая для полного закрытия задвижки по регламенту) в течение которой проверяется условие закрытия задвижки. При достижении указанного условия сигнал, выданный на аварийное закрытие задвижки, снимается, оператор получает соответствующее информационное сообщение и происходит протоколирование по окончании алгоритма. При не закрытии задвижки за обозначенный промежуток времени так же происходит снятие сигнала, выданного на аварийное закрытие задвижки, оператор получает информационное сообщение, о том, что аварийного закрытия задвижки не произошло и происходит протоколирование.

ПРОГРАММА ГЛАВНАЯ НАЧАЛО

: Программа - пример
: управление задвижки с электроприводом

Вкл тс1 : Глобальный счетчик секундный №1

Если тс1 > 1.0

{ пв1 = 1 : Флаг новая секунда = 1
тс1 = 0
}

ЗАДВИЖКА

пв1 = 0 : Флаг новая секунда = 0

КОНЕЦ

ПРОЦЕДУРА ЗАДВИЖКА НАЧАЛО

: Алгоритм управления задвижки с электроприводом

: вд[open] концевик "открыто" задвижки
: вд[close] концевик "закрыто" задвижки
: вд[power] сигнал наличия напряжения в цепи управления
: дв[openO] сигнал управления "ОТКРЫТЬ" (физический)
: дв[closeO] сигнал управления "ЗАКРЫТЬ" (физический)
: дв[status] сигнал управления "РЕЖИМ"
: рв[rv] состояние задвижки
: пц[Record] номер задвижки
: пц[RecRV] номер рв управления
: пв[time] счетчик временной задвижки

: рв[rv].a6 макс. время хода задвижки, с (в)

: рв[rv] = 0 задвижка "НОРМА" (в)
: рв[rv] = 1 задвижка "НЕИСПРАВНА" (в)
: 1000 < рв[rv] < 1999 задвижка "ЗАКРЫВАЕТСЯ" (в)
: 2000 < рв[rv] < 2999 задвижка "ОТКРЫВАЕТСЯ" (в)

: вд[open].a32 Не открылась
: вд[close].a32 Не закрылась

: рв[rv].a7 = 0 задвижка "НОРМА" (нет управления) (в)
: рв[rv].a7 = 1 задвижку "ЗАКРЫТЬ" (вирт.) (в)
: рв[rv].a7 = 2 задвижку "ОТКРЫТЬ" (вирт.) (в)
: рв[rv].a7 = 3 задвижку "ЗАКРЫТЬ при АВАРИИ" (вирт.) (в)

: дв[openO].a33 = 1 задвижка "ЗАБЛОКИРОВАНА" (л)

: Задвижка 701 02 03 04 05 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 21
722 723 724 726 727 728 729 730 731 732 733 735 736 737 740 741

Для

VDopen[82,85,88,91,94,100,103,106,109,112,118,121,124,127,130,136,139,142,145,148,97,115,
133,151,166,169,172,175,178,181,184,187,154,196,199,190,193],

closeVD[83,86,89,92,95,101,104,107,110,113,119,122,125,128,131,137,140,143,146,149,98,116,134,152,167,170,173,176,179,182,185,188,155,197,200,191,194],

power[84,87,90,93,96,102,105,108,111,114,120,123,126,129,132,138,141,144,147,150,99,117,135,153,168,171,174,177,180,183,186,189,156,198,201,192,195],

Open[1, 3, 5, 7, 9, 13, 15, 17, 19, 21, 25, 27, 29, 31, 33, 37, 39, 41, 43, 45,11, 23, 35, 47, 59, 61, 63, 65, 67, 69, 71, 73, 49, 79, 81, 75, 77],

Oclose[2, 4, 6, 8,10, 14, 16, 18, 20, 22, 26, 28, 30, 32, 34, 38, 40, 42, 44, 46,12, 24, 36, 48, 60, 62, 64, 66, 68, 70, 72, 74, 50, 80, 82, 76, 78],

status[99...103,105...109,111...115,117...121,104,110,116,122,124...131,123,134,135,132,133],

Record[701...724,726...733,735,736,737,740,741],

move[701...724,726...733,735,736,737,740,741],

RVecord[601...624,626...633,635,636,637,640,641],

Nopen[501...524,526...533,535,536,537,540,541],

Nclose[401...424,426...433,435,436,437,440,441],

rv[1...35,38,39]

{

Если пл[Nclose] = 1

{ вд[closeVD].a32 = 1 } : Флаг не закрылась

Если пл[Nopen] = 1

{ вд[VDopen].a32 = 1 } : Флаг не открылась

Если рв[rv].a6 > 999 : Ограничение максимума хода ИМ 999 сек

{ рв[rv].a6 = 999 }

: ***** Обработка сигнала "НЕИСПРАВНОСТЬ" *****

Если 1=(вд[VDopen]&вд[closeVD]) : Если два концевика "1" "1"

{ пл[Nopen] = 0 : Флаг превышено время открытия

пл[Nclose] = 0 : Флаг превышено время закрытия

рв[rv] = 1 } : Флаг задвижка неисправна

Иначе

{ Если вд[VDopen] = 1 : Если концевики "0" "1"

{ Если вд[closeVD] = 0

{ пл[Nopen] = 0 : Флаг превышено время открытия

пл[Nclose] = 0 : Флаг превышено время закрытия

рв[rv] = 0 }}

Если вд[VDopen] = 0 : Если концевики "1" "0"

{ Если вд[closeVD] = 1

{ пл[Nopen] = 0 : Флаг превышено время открытия

пл[Nclose] = 0 : Флаг превышено время закрытия

рв[rv] = 0 }}}

Если рв[rv].a7 = 2.0 : Если пробуют открыть

{ Если дв[Oclose] = 1

{ рв[rv].a7 = 0.0

пл[Record]= 1 }}

Если рв[rv].a7 = 1.0 : Если пробуют закрыть

{ Если дв[Oopen] = 1

{ рв[rv].a7 = 0.0

пл[Record]= 1 }}

: ***** Обработка сигнала оператора "ОТКРЫТЬ" *****

```

Если рв[rv].a7 = 2.0      : Если пробуют открыть
{ Если рв[rv] # 1.0      : Если задвижка исправна
  { Если вд[VDopen] = 0   : Если задвижка не открыта
    { Если вд[power] = 1   : Если есть напряжение в цепи упр
      { пл[Record]= 0 }
      Иначе
      { message("Управление невозможно НЕТ НАПРЯЖЕНИЯ №",пц[Record])
        пл[Record]= 1 }}
      Иначе
      { message(" Уже открыта ЗАДВИЖКА №",пц[Record])
        пл[Record]= 1 }}
    Иначе
    { message("Управление невозможно НЕИСПРАВНА №",пц[Record])
      пл[Record]= 1 }
  Если рв[rv] > 1000.0
  { пл[Record]= 1 }

```

```

Если пл[Record] = 0
{ Если дв[status] = 1   : Если задвижка "ДИСТАНЦИЯ"
  { пл[Record]= 0 }
  Иначе
  { пл[Record]= 1 }}

```

```

Если пл[Record] = 0
{ пл98 = дв[Oopen].a33

```

```

  Если пл98 = 0      : Если задвижка не заблокирована
  { дв[Oopen] = 1 }  : Выдать сигнал управления "ОТКРЫТЬ"
  Иначе
  { message("Управление невозможно ЗАБЛОКИРОВАНА 0 №",пц[Record])
    пл[Record]= 1 }}
  пл98 = 0
}

```

: ***** Обработка сигнала управления "ОТКРЫТЬ" *****

```

Если дв[Oopen] = 1      : Если дан сигнал открытия
{ пв[move] = пв[move] + пв1
  рв[rv] = 1000.0 + пв[move]      : Включить счетчик открытия

Если вд[VDopen] = 0      : Не сработал концевик "ОТКРЫТО"
{ Если рв[rv] > (рв[rv].a6 + 1000.0) : Превышено время открытия
  { дв[Oopen] = 0      : Снять сигнал открытия
    рв[rv] = 1.0
    пл[Record] = 1
    пл[Nopen] = 1      : Флаг превышено время открытия
    message("Задвижка не открылась НЕИСПРАВНА №",пц[Record])
    пв[move] = 0 }}
  Иначе
  { дв[Oopen] = 0      : Снять сигнал открытия
    рв[rv] = 0.0
    пл[Record] = 1
    пв[move] = 0 }}      : Обнулить счетчик открытия

```

```

: ***** Обработка сигнала "ЗАКРЫТЬ" при АВАРИИ *****
Если рв[rv].a7 = 3.0          : Если сигнал закрыть при аварии
{ дв[Oopen] = 0              : Снять сигнал управления
  дв[status] = 1             : Дистанционный режим
  дв[Oclose] = 1             : Закрыть задвижку
  message("АВАРИЯ закрытие ЗАДВИЖКА № ",пц[Record])
  рв[rv].a7 = 4.0 }

Если рв[rv].a7 = 4.0          : Если сигнал закрыть при аварии
{ Если дв[Oopen] = 0
  { дв[Oopen].a33 = 1         : Заблокировать задвижку
    пл[Record] = 1
    message("ЗАБЛОКИРОВАНА ЗАДВИЖКА № ",пц[Record]) }}
: ***** Обработка сигнала оператора "ЗАКРЫТЬ" *****
Если рв[rv].a7 = 1.0          : Если пробуют закрыть
{ Если рв[rv] # 1.0            : Если задвижка исправна
  { Если вд[closeVD] = 0       : Если задвижка не закрыта
    { Если вд[power] = 1       : Если есть напряжение в цепи упр
      { пл[Record]= 0 }
      Иначе
      { message("Управление невозможно НЕТ НАПРЯЖЕНИЯ №",пц[Record])
        пл[Record]= 1 }}
      Иначе
      { message(" Уже закрыта ЗАДВИЖКА №",пц[Record])
        пл[Record]= 1 }}
    Иначе
    { message("Управление невозможно НЕИСПРАВНА №",пц[Record])
      пл[Record]= 1 }

Если рв[rv] > 1000.0
{ пл[Record]= 1 }

Если пл[Record] = 0
{ Если дв[status] = 1          : Если задвижка "ДИСТАНЦИЯ"
  { пл[Record]= 0 }
  Иначе
  { пл[Record]= 1 }}

Если пл[Record] = 0
{ пл97 = дв[Oopen].a33

  Если пл97 = 0 : Если задвижка не заблокирована
  { дв[Oclose] = 1 }
  Иначе
  { message("Управление невозможно ЗАБЛОКИРОВАНА 3 №",пц[Record])
    пл[Record]= 1 }}
  пл97 = 0 }
: ***** Обработка сигнала управления "ЗАКРЫТЬ" *****
Если дв[Oclose] = 1          : Если дан сигнал закрытия
{ пв[move] = пв[move] + пв1
  рв[rv] = 2000.0 + пв[move]   : Включить счетчик закрытия
  Если вд[closeVD] = 0        : Не сработал концевик "ЗАКРЫТО"
  { Если рв[rv] > (рв[rv].a6 + 2000.0) : Превышено время закрытия
    { дв[Oclose] = 0           : Снять сигнал закрытия

```

```
    рв[rv] = 1.0
    пл[Record] = 1
    пл[Nclose] = 1           : Флаг превышено время закрытия
    message("Задвижка не закрылась НЕИСПРАВНА №",пц[Record])
    пв[move] = 0 }}
Иначе
{ дв[Oclose] = 0           : Снять сигнал закрытия
  рв[rv] = 0.0
  пл[Record] = 1
  пв[move] = 0 }}         : Обнулить счетчик закрытия
Если пл[Record] = 1
{ пл[Record] = 0
  рв[rv].a7 = 0.0
  посРВ(пц[RVecord]) }
}
КОНЕЦ
```


В.1.7 Пример 7. Управление электрооборудованием (вентилятор)

Данный алгоритм описывает порядок работы управляющего контроллера при обслуживании различного электрооборудования с двумя командами управления **«Включить»** и **«Выключить»** и двумя состояниями оборудования: **«Включено»** и **«Выключено»**, на примере вытяжных вентиляторов.

Алгоритмы не исключают возможности как местного, так и дистанционного управления вентиляторами и обеспечивают выполнение процедуры диагностики исправности вентилятора путем проверки достоверности снимаемых с него сигналов.

Для работы алгоритмов используются следующие сигналы и информация:

- дискретный сигнал «Наличие напряжения в цепи управления вентилятором»
- дискретный сигнал «Сработало тепловое реле»
- внутренний параметр системы «Вентилятор неисправен»
- внутренний параметр системы «Вентилятор заблокирован»
- дискретный сигнал **«Вентилятор включен»**;
- дискретный сигнал **«Вентилятор выключен»**.

В результате выполнения алгоритмов формируются управляющие сигналы и информация о текущем состоянии вентилятора:

- дискретный сигнал **«Включить»**
- дискретный сигнал **«Выключить»**.

Алгоритм контроля состояния вентилятора. Выполнение алгоритма происходит при изменении значения хотя бы одного сигнала, снимаемого с вентилятора. При изменении состояния сигналов получаемых с вентилятора осуществляется выдача соответствующих сообщений. Алгоритм выполняется по следующей схеме:

- происходит проверка изменения состояния наличия напряжения в цепи управления вентилятором
- происходит проверка изменения состояния сигнала о срабатывании теплового реле
- происходит проверка изменения сигналов **«Вентилятор включен»**, **«Вентилятор выключен»** и установление параметров неисправности вентилятора при получении недостоверных вышеназванных сигналов **«-1» «1»** или **«0» «0»**.

Алгоритм включения вентилятора предназначен для выполнения системой операции дистанционного включения вентилятора. Алгоритм выполняется следующим образом:

- проверяется флаг исправности указанного вентилятора
- проверяется сигнал с теплового реле. При его наличии оператор получает сообщение, информирующее его о том, что дальнейший пуск вентилятора невозможен
- проверяются состояние, в котором в данный момент находится указанный вентилятор: включен, выключен (если вентилятор оказывается включен, дальнейшее выполнение алгоритма прекращается, информируя оператора текстовым сообщением)
- проверяется флаг блокировки управления вентилятора. При его наличии оператор получает соответствующее сообщение, и выполнение алгоритма на этом заканчивается операцией протоколирования, как и в предыдущем случае
- проверяется наличие напряжения в цепи управления вентилятором. При его отсутствии оператор получает соответствующее сообщение, и выполнение алгоритма на этом заканчивается операцией протоколирования, как и в предыдущем случае;
- выдаётся сигнал на включение вентилятора
- даётся временная выдержка, необходимая для срабатывания исполнительного механизма и сигнал включения снимается

- проверяются сигналы **«Вентилятор №... включен»**, **«Вентилятор №... выключен»**. В результате проверки выявляется, включился ли указанный вентилятор. В случае его не включения оператор получает информационное сообщение о невыполнении алгоритма. В противном случае, если вентилятор включился, информацию о включении последнего поступит на экран оператора из алгоритма контроля состояния вентилятора.

Алгоритм выключения вентилятора предназначен для выполнения системой операции дистанционного выключения вентилятора. Алгоритм выполняется следующим образом:

- проверяется флаг исправности указанного вентилятора (в случае если вентилятор оказывается неисправным, оператор получает соответствующее сообщение и выполнение алгоритма при этом завершается операцией протоколирования)
- проверяются состояние, в котором в данный момент находится указанный вентилятор: включен, выключен (если вентилятор выключен, дальнейшее выполнение алгоритма прекращается, информируя оператора текстовым сообщением)
- проверяется наличие напряжения в цепи управления вентилятором. При его отсутствии оператор получает соответствующее сообщение, и выполнение алгоритма на этом заканчивается операцией протоколирования, как и в предыдущем случае
- выдаётся сигнал на выключение вентилятора
- даётся временная выдержка, необходимая для срабатывания исполнительного механизма выключения вентилятора и сигнал выключения снимается
- проверяются сигналы **«Вентилятор включен»**, **«Вентилятор выключен»**. В результате проверки выявляется, выключился ли указанный вентилятор. В случае его не выключения, оператор получает информационное сообщение о невыполнении алгоритма. В противном случае, если вентилятор выключился, информацию о выключении последнего поступит на экран оператора из алгоритма контроля состояния вентилятора.

Алгоритм выключения вентилятора при аварии предназначен для выполнения системой операции дистанционного выключения вентилятора в аварийной ситуации. Алгоритм одновременно блокирует возможность дальнейшего управления вентилятором как дистанционно, так и по месту. Алгоритм выполняется следующим образом:

- независимо от состояния вентилятора устанавливается сигнал выключения вентилятора
- устанавливается флаг блокировки местного и дистанционного включения
- производится временная выдержка, необходимая для срабатывания исполнительных механизмов отключения вентилятора
- проверяются сигналы **«Вентилятор включен»**, **«Вентилятор выключен»**. В результате проверки выявляется, выключился ли указанный вентилятор. В любом случае оператор получает информационное сообщение о выполнении алгоритма аварийного отключения.

ПРОГРАММА ГЛАВНАЯ
НАЧАЛО

: Программа - пример
: управление электрооборудованием

Вкл тс1 : Глобальный счетчик секундный №1
Если тс1 > 1.0
{ пв1 = 1 : Флаг новая секунда = 1
тс1 = 0 }
ВЕНТИЛ

пв1 = 0 : Флаг новая секунда = 0
ВЫХОД
КОНЕЦ

ПРОЦЕДУРА ВЕНТИЛ

НАЧАЛО

: Алгоритм управления вытяжными и аварийными вентиляторами

:

: вд[VDon] дискретный сигнал "Включен"

: вд[offVD] дискретный сигнал "Отключен"

: вд[Apow] дискретный сигнал "Нет напряжения"

: дв[Oon] дискретный сигнал "Включить"

: дв[Ooff] дискретный сигнал "Отключить"

: рв[Vrv] переменная управления вентилятором

:: рв[Vrv] = 0 вентилятор "НОРМА" (в)

: рв[Vrv] = 1 вентилятор "НЕИСПРАВЕН" (в)

: рв[Vrv].a7 = 0 вентилятор "НОРМА" (нет управления) (в)

: рв[Vrv].a7 = 1 вентилятор "ОТКЛЮЧИТЬ" (вирт.) (в)

: рв[Vrv].a7 = 2 вентилятор "ВКЛЮЧИТЬ" (вирт.) (в)

: рв[Vrv].a7 = 3 вентилятор "ОТКЛЮЧИТЬ при АВАРИИ" (вирт.) (в)

: дв[Ooff].a33 = 1 вентилятор "ЗАБЛОКИРОВАН" (л)

: вент B1 B2 B3 B4 BA1

Для VDon[40,44,48,52, 56],

offVD[41,45,49,53, 57],

Arele[42,46,50,54, 58],

Apow[43,47,51,55, 59],

Oon[24,26,28,30, 32],

Ooff[25,27,29,31, 33],

ARec[551...555],

BRec[551...555],

CRec[571...575],

Vrv[4...8]

{

Если 1=(вд[VDon]&вд[offVD])

{ рв[Vrv] = 1.0 }

Если 0=(вд[VDon]&вд[offVD])

{ рв[Vrv] = 1.0 }

Если вд[VDon] = 1

{ Если вд[offVD] = 0

{ рв[Vrv] = 0.0 }}

Если вд[VDon] = 0

{ Если вд[offVD] = 1

{ рв[Vrv] = 0.0 }}

Если рв[Vrv].a7 = 2.0

: Если пробуют включить

{ Если рв[Vrv] # 1.0

: Если вентилятор исправен

{ Если вд[VDon] = 0

: Если не включен

{ Если вд[Apow] = 1

: Если есть напряжение в цепи упр

{ пл[ARec]= 0 }

Иначе

{ message("Управление невозможно НЕТ НАПРЯЖЕНИЯ ВЕНТИЛЯТОР")

пл[ARec]= 1 }}

Иначе

{ message("Уже включен ВЕНТИЛЯТОР")

```

    пл[ARec]= 1 }}
Иначе
{ message("Управление невозможно НЕИСПРАВЕН ВЕНТИЛЯТОР")
  пл[ARec]= 1 }}

Если рв[Vrv].a7 = 1.0      : Если пробуют выключить
{ Если рв[Vrv] # 1.0      : Если вентилятор исправен
  { Если вд[offVD] = 0      : Если не выключен
    { Если вд[Arow] = 1      : Если есть напряжение в цепи упр
      { пл[ARec]= 0 }
    Иначе
      { message("Управление невозможно НЕТ НАПРЯЖЕНИЯ ВЕНТИЛЯТОР")
        пл[ARec]= 1 }}
  Иначе
    { message("Уже выключен ВЕНТИЛЯТОР")
      пл[ARec]= 1 }}
Иначе
  { message("Управление невозможно НЕИСПРАВЕН ВЕНТИЛЯТОР")
    пл[ARec]= 1 }}
: ***** Обработка сигнала "ВЫКЛЮЧИТЬ" при АВАРИИ *****
Если рв[Vrv].a7 = 3.0
{ дв[Oon] = 0
  дв[Ooff] = 1
  пл[ARec]= 1 }

Если рв[Vrv].a7 = 2.0
{ Если пл[ARec]= 0
  { пл[ARec]= 1
    дв[Oon] = 1 }}

Если рв[Vrv].a7 = 1.0
{ Если пл[ARec]= 0
  { пл[ARec]= 1
    дв[Ooff] = 1 }}

Если дв[Oon] = 1      : Если сигнал включения
{ Если пв1 = 1      : Если прошла секунда
  { дв[Oon] = 0      : Снять сигнал управления
    Если вд[VDon] = 0      : Если не сработал датчик включения
      { message("Не включился ВЕНТИЛЯТОР") }}}

Если дв[Ooff] = 1      : Если сигнал выключения
{ Если пв1 = 1      : Если прошла секунда
  { дв[Ooff] = 0      : Снять сигнал управления
    Если вд[offVD] = 0      : Если не сработал датчик выключения
      { message("Не выключился ВЕНТИЛЯТОР" ) }}}

Если пл[ARec] = 1
{ пл[ARec] = 0
  рв[Vrv].a7 = 0.0
  посРВ(пц[BRec]) }
}

```

КОНЕЦ

В.1.8 Пример 8. Управление насосом

Данный алгоритм описывает порядок работы управляющего контроллера при обслуживании насосов с задвижками в обвязке. Алгоритм позволяет управлять включением (выключением) насоса с регламентированным автоматическим открытием (закрытием) задвижки на выкиде и проверкой состояния входной задвижки.

Алгоритмы не исключают возможности как местного, так и дистанционного управления агрегатами откачки и перекачки.

Этот пример является обобщающим и вмещает в себя практически все вышеописанные примеры.

Для работы алгоритмов используются следующие модули:

- алгоритм контроля состояния и управления задвижкой (пример 6)
- алгоритм контроля состояния и управления насосом (на базе примера 7)
- алгоритмы защит и блокировок насоса

В результате выполнения алгоритмов формируются управляющие сигналы для насоса и задвижки на выкиде, вырабатываемые вышеуказанными алгоритмами.

Алгоритм включения насоса с задвижками в обвязке предназначен для выполнения системой операции включения насоса с автоматическим открытием задвижки на выкиде. Алгоритм выполняется следующим образом:

- осуществляется проверка оборудования. Проверка оборудования включает в себя проверку входной задвижки насоса, задвижки на выкиде и состояния насоса. Для осуществления пуска агрегата задвижка на входе должна быть открыта, насос должен быть выключен и готов к дистанционному пуску (присутствует напряжение в цепи управления двигателем насоса, насос не заблокирован, исправен, и т.п.), задвижка на выкиде должна быть закрыта и готова к выполнению операции дистанционного открытия (должна быть исправна, в цепи управления должно присутствовать напряжение, режим управления должен соответствовать дистанционному, блокировка должна быть снята). При несоответствии вышеуказанных параметров норме система запрещает возможность автоматического пуска, информируя об этом оператора текстовым сообщением с указанием причин.
- осуществляется проверка внешних параметров. В ходе этой проверки проверяется
 - давление охлаждающей жидкости
 - температура охлаждающей жидкости
 - давления уплотнительной жидкости
 - Температура подшипников
 - Давление в уравнильной линии.

Кроме этого, используется сигнал:

- дискретный сигнал «Минимальное давление на выкиде»

При несоответствии этих параметров норме система запрещает возможность автоматического пуска, информируя об этом оператора текстовым сообщением с указанием причин.

- выполняется алгоритм включения насоса. В случае если двигатель насоса не пустился, оператор получит текстовое сообщение и выполнение алгоритма на этом прекратится
- при успешном выполнении алгоритма пуска насоса алгоритм устанавливает временную выдержку, в течение которой проверяет давление на выкиде насоса. При

достижении последнего не нулевой отметки выполняется алгоритм открытия задвижки на выкиде.

- в случае если насос не набрал давления на выкиде или задвижка на выкиде не открылась, оператор получит информационное сообщение и двигатель насоса будет автоматически выключен.

Алгоритм выключения насоса с задвижками в обвязке предназначен для выполнения операции выключения насоса с автоматическим закрытием задвижки на выкиде. Алгоритм выполняется следующим образом:

- осуществляется проверка оборудования, в ходе которой устанавливается состояние оборудования и возможность управления им. Если насос находится во включенном состоянии и задвижки на входе и на выкиде насоса открыты, а так же разрешено дистанционное управление, вышеуказанным оборудованием, то осуществляется дальнейшее выполнение алгоритма. В противном случае оператор получает информационное сообщение, и выполнение алгоритма прекращается. В данной ситуации отключение насоса и закрытие задвижки необходимо выполнять дистанционно (путём выполнения соответствующих алгоритмов) или по месту.
- при успешной проверке оборудования выполняется алгоритм выключения насоса
- выполняется проверка условия выключения насоса. В случае если отключения не произошло, оператор получит сообщение, и задвижка на выкиде закрываться не будет. Дальнейшее управление оборудованием возможно производить как дистанционно так и по месту.
- при успешном выключении насоса задвижка на выкиде будет закрыта и оператор будет проинформирован об успешном выполнении алгоритма текстовым сообщением.

Алгоритм отключения насоса с задвижками в обвязке при аварии предназначен для выполнения операции аварийного отключения агрегата. Алгоритм выполняется следующим образом:

- выполняется алгоритм выключения насоса при аварии
- выполняется алгоритм закрытия задвижки на выкиде при аварии

Информацию об аварийном выключении оборудования оператор будет получать из алгоритма аварийного выключения насоса и алгоритма аварийного закрытия задвижки.

Программа ГЛАВНАЯ
НАЧАЛО

: Программа - пример
: управление насосом в обвязке

Вкл тс1 : Глобальный счетчик секундный №1

Если тс1 > 1.0
{ пв1 = 1 : Флаг новая секунда = 1
тс1 = 0 }

ЗАДВИЖКА : текст процедуры приведен в примере №6
НАСОС

пв1 = 0 : Флаг новая секунда = 0

ВЫХОД
КОНЕЦ

ПРОЦЕДУРА НАСОС : Алгоритм управления насосами
НАЧАЛО

АГРЕГАТ

```
:насосы 701/1 /2 /3 /4 704 705
Для VDon[202,206,210,214,218,222], : Концевик "ВКЛЮЧЕН"
  offVD[203,207,211,215,219,223], : Концевик "ВЫКЛЮЧЕН"
    Oon[ 83, 86, 89, 92, 95, 97], : Сигнал "ВКЛЮЧИТЬ"
    Ooff[ 84, 87, 90, 93, 96, 98], : Сигнал "ОТКЛЮЧИТЬ"
    AOoff[ 85, 88, 91, 94,159,160], : Сигнал "ОТКЛЮЧИТЬ ПРИ АВАРИИ"
  Astatus[136,137,138,139,157,158], : Сигнал "РЕЖИМ"
  Auout[ 26, 28, 30, 32, 39, 35], : Переменная управления задвижки на нагнетании
  Arpag[38...43], : Давление на нагнетании (максимум)
  ARec[751...756],
  IRec[751...756],
  CRec[761...766],
  BRec[771...776],
  DRec[781...786],
  ERec[971...976],
  JRec[991...996],
  Vrv[53...58],
  Vxrv[71...76]
{
  Если 1=(вд[VDon]&вд[offVD]) : Если оба концевика "1"
  { рв[Vrv] = 1.0 } : Насос неисправен

  Если 0=(вд[VDon]&вд[offVD]) : Если оба концевика "0"
  { рв[Vrv] = 1.0 } : Насос неисправен

  Если вд[VDon] = 1
  { Если вд[offVD] = 0 : Если оба концевика #
    { рв[Vrv] = 0.0 } : Насос норма

  Если вд[VDon] = 0
  { Если вд[offVD] = 1 : Если оба концевика #
    { рв[Vrv] = 0.0 } : Насос норма

  Если рв[Vrv].a7 = 2.0 : Если пробуют включить
  { Если рв[Vxrv] = 0.0 : Если агрегат готов
    { пл[ARec]= 0 }
    Иначе
    { message("ПУСК НЕ ВОЗМОЖЕН насос",пл[CRec])

  Если рв[Vxrv].a7 # 0
  { message("Задвижка на выкиде НЕ ГОТОВА К ПУСКУ !") }
  Если рв[Vxrv].a6 # 0
  { message("Задвижка на всасе НЕ ГОТОВА К ПУСКУ !") }
  Если рв[Vxrv] = 5.0
  { message("Внешние параметры насоса АВАРИЙНЫ !") }

  пв901 = рв[Vxrv].a7 + рв[Vxrv].a6
  пв902 = рв[Vxrv]
```

```

Если пв901 = 0
{ Если пв902 = 1.0
  { message("Насос НЕ ГОТОВ К ПУСКУ ") }}
пл[ARec]= 1 }}

Если рв[Vrv].a7 = 1.0      : Если пробуют выключить
{ Если рв[Vxrv].a7 = 0.0   : Если задвижка на выкиде готова
  { пл[ARec] = 0 }
  Иначе
  { message("ДИСТАНЦИОННЫЙ ОСТАНОВ НЕ ВОЗМОЖЕН насос",пц[CRec])
    message("Задвижка на выкиде НЕ ГОТОВА !") }}
Если рв[Vrv].a7 = 2.0      : Формирование сигнала Включить
{ Если пл[ARec]= 0
  { пл[ARec]= 1
    дв[Oon] = 1 }}

Если рв[Vrv].a7 = 1.0      : Формирование сигнала Выключить
{ Если пл[ARec]= 0
  { пл[ARec] = 1
    дв[Ooff] = 1 }}

Если дв[Oon] = 1           : Если сигнал включения
{ Если пв1 = 1             : Если прошла 1 секунда
  { пл[BRec] = 1
    дв[Oon] = 0 }}        : Снять сигнал управления

Если пл[BRec] = 1
{ Если вд[VDon] = 1        : Если насос включился
  { пл[BRec] = 0
    пл[ERec] = 1 }
Если пв3 = 1
{ Если вд[VDon] = 0        : Если не сработал датчик включения
  { пл[BRec] = 0
    message("НЕ ВКЛЮЧИЛСЯ двигатель насоса",пц[CRec]) }}}

Если пл[ERec] = 1          : Если включили насос
{ Если пв1 = 1
  { пв[ERec] = пв[ERec] + 1 }

Если пв[ERec] < 20
{ Если вд[Arpag] = 1        : Если есть сигнал высокого давление
  { рв[Auout].a7 = 2.0      : Открыть задвижку на выкиде
    message("Агрегат набрал необходимое давление за (сек), ", пв[ERec])
    пл[JRec] = 1            : Разрешить блокировку по PL на выкиде
    пв[ERec] = 0
    пл[ERec] = 0 }}

Если пв[ERec] > 20
{ Если вд[Arpag] = 0
  { дв[Ooff] = 1           : Отключить двигатель насоса
    message("НАСОС НЕ НАБРАЛ НЕОБХОДИМОЕ ДАВЛЕНИЕ ",пц[CRec])
    пв[ERec] = 0
    пл[ERec] = 0 }}}

```



```

Если дв[Ooff] = 1          : Если сигнал выключения
{ Если пв1 = 1              : Если прошла 1 секунды
  { пл[DRec] = 1           : Флаг выключения
    дв[Ooff] = 0 }}        : Снять сигнал управления

Если дв[AOoff] = 1         : Если сигнал выключения при аварии
{ дв[Oon] = 0              : Снять сигнал включения
  пл[CRec] = 1             : Флаг аварийного отключения

Если пл765 = 1             : Для насоса 704
{ дв96 = дв159 }
Если пл766 = 1             : Для насоса 705
{ дв98 = дв160 }

Если пв1 = 1              : Если прошла 1 секунды
{ пл[DRec] = 1            : Флаг выключения
  message("АВАРИЙНОЕ отключение насоса",пц[CRec])
  дв[Astatus] = 1         : Режим "ДИСТАНЦИЯ"
  дв[AOoff] = 0           : Снять сигнал управления

Если пл765 = 1            : Для насоса 704
{ дв96 = дв159 }
Если пл766 = 1            : Для насоса 705
{ дв98 = дв160 }}}

Если пл[DRec] = 1
{ Если пв3 = 1
  { Если вд[offVD] = 0      : Если не сработал датчик выключения
    { пл[DRec] = 0
      message("НЕ ОТКЛЮЧИЛСЯ двигатель насоса",пц[CRec]) }}

Если вд[offVD] = 1        : Если выключился двигатель насоса
{ пл[JRec] = 0            : Байпасировать блокировку по PL на выкиде
  пл[DRec] = 0
  Если пл[CRec] = 1
  { дв[Oon].a33 = 1        : Заблокировать насос
    message("ЗАБЛОКИРОВАН НАСОС ",пц[CRec])
    рв[Auout].a7 = 3.0 }    : Закрыть задвижку на выкиде при аварии
  Иначе
  { рв[Auout].a7 = 1.0 }}} : Закрыть задвижку на выкиде

Если пл[ARec] = 1
{ пл[ARec] = 0
  рв[Vrv].a7 = 0.0
  посРВ(пц[IRec]) }}

КОНЕЦ

ПРОЦЕДУРА АГРЕГАТ          : Пуск / останов насосов в объёмке
НАЧАЛО

Для TN[17...40],Trl[911...934]
{ Если ва[TN] > ва[TN].a16   : Если температура подш > ВАГ
  { пл[Trl] = 1 }

```

Если $va[TN] < va[TN].a16$: Если температура подш > ВАГ
 { $пл[Trp] = 0$ }

$пл941 = пл911|пл912|пл913|пл914$: Флаг Тмах подшипников насоса 701/1
 $пл942 = пл915|пл916|пл917|пл918$: Флаг Тмах подшипников насоса 701/2
 $пл943 = пл919|пл920|пл921|пл922$: Флаг Тмах подшипников насоса 701/3
 $пл944 = пл923|пл924|пл925|пл926$: Флаг Тмах подшипников насоса 701/4
 $пл945 = пл927|пл928|пл929|пл930$: Флаг Тмах подшипников насоса 704
 $пл946 = пл931|пл932|пл933|пл934$: Флаг Тмах подшипников насоса 705
 :нас.701/1 /2 /3 /4 704 705

:Задв. 726 728 730 732 740 736 на всасе

:Задв. 727 729 731 733 741 737 на выкиде

Для $Nh[202,206,210,214,218,222]$, : Концевик "ВКЛЮЧЕН"

$Nl[203,207,211,215,219,223]$, : Концевик "ВЫКЛЮЧЕН"

$Ni[204,208,212,216,220,224]$, : Концевик "Наличие напряжения"

$Nx[205,209,213,217,221,225]$, : Концевик "Реле х/х"

$Nd[83,86,89,92,95,97]$, : Сигнал "ВКЛЮЧИТЬ"

$NA[85,88,91,94,159,160]$, : Сигнал "ОТКЛЮЧИТЬ ПРИ АВАРИИ"

$Nuin[25,27,29,31,38,34]$, : Переменная управления задвижки на всасе

$Ncin[167,173,179,185,191,197]$, : Концевик "ЗАКРЫТА"

$Nein[168,174,180,186,192,198]$, : Сигнал "Наличие напряжения"

$Nuout[26,28,30,32,39,35]$, : Переменная управления задвижки на нагнетании

$Noout[169,175,181,187,193,199]$, : Концевик "ОТКРЫТА"

$Ncout[170,176,182,188,194,200]$, : Концевик "ЗАКРЫТА"

$Neout[171,177,183,189,195,201]$, : Сигнал "Наличие напряжения"

$Nbout[61,65,69,73,77,81]$, : Сигнал "ОТКРЫТЬ"

$Nrout[125,127,129,131,133,135]$, : Сигнал "Режим управления"

$Ncogl[32...37]$, : Давление охлаждающей жидкости (минимум)

$Nfogh[26...31]$, : Температура охлаждающей жидкости (максимум)

$Nkogl[52...57]$, : Уровень уплотнительной жидкости (минимум)

$Npnag[44...49]$, : Давление на нагнетании (минимум)

$Nrvpr[71...76]$, : Обобщенный параметр агрегата

$Nsnas[53...58]$, : Переменная управления насоса

$Nypl[901...906]$, : Флаг внешних параметров насоса

$Ntpl[941...946]$, : Флаг Т подшипников насоса

$Nwon[951...956]$, : Флаг останова насоса

$Nqec[761...766]$,

$NJec[991...996]$

{
 $рв[Nrvpr] = 0.0$: Внутренние параметры насоса

$рв[Nrvpr].a6 = 0.0$: Параметры задвижки на всасе

$рв[Nrvpr].a7 = 0.0$: Параметры задвижки на нагнетании

: Проверка внешних параметров насоса

$пл[Nypl] = вд[Ncogl]|вд[Nfogh]|вд[Nkogl]|вд66$

Если $пл[NJec] = 1$

{ $пл[Nwon] = пл[Nypl]|вд[Npnag]|пл[Ntpl]$ }

Иначе

{ $пл[Nwon] = пл[Nypl]|пл[Ntpl]$ }

Если $пл[Nypl] = 0$: Если внешние параметры в норме

{

: Проверка внутренних параметров насоса (рв[Nrvpr])

```

Если вд[Nh] = 1          : Если насос включен
{ рв[Nrvpr] = 1.0 }
Если рв[Nsnas] = 1.0     : Если насос неисправен
{ рв[Nrvpr] = 1.0 }
Если дв[Nd].a33 = 1      : Если насос заблокирован
{ рв[Nrvpr] = 1.0 }
Если вд[Ni] = 0          : Если нет напряжения
{ рв[Nrvpr] = 1.0 }
Если вд[Nx] = 1          : Если сработало реле
{ рв[Nrvpr] = 1.0 }

```

: Проверка задвижки на всасе (рв[Nrvpr].a6)

```

Если вд[Ncin] = 1        : Если задвижка закрыта
{ рв[Nrvpr].a6 = 1.0 }
Если рв[Nuin] > 1000.0    : Если задвижка открывается/закрывается
{ рв[Nrvpr].a6 = 1.0 }
Если рв[Nuin] = 1.0       : Если задвижка неисправна
{ рв[Nrvpr].a6 = 1.0 }
Если вд[Nein] = 0         : Если нет напряжения на задвижке
{ рв[Nrvpr].a6 = 1.0 }

```

: Проверка задвижки на нагнетании (рв[Nrvpr].a7)

```

Если рв[Nrvpr].a7 = 2.0
{ Если вд[Noout] = 1      : Если задвижка открыта
  { рв[Nrvpr].a7 = 1.0 } }
Если рв[Nrvpr].a7 = 1.0
{ Если вд[Ncout] = 1      : Если задвижка закрыта
  { рв[Nrvpr].a7 = 1.0 } }
Если рв[Nuout] > 1000.0   : Если задвижка открывается/закрывается
{ рв[Nrvpr].a7 = 1.0 }
Если рв[Nuout] = 1.0      : Если задвижка неисправна
{ рв[Nrvpr].a7 = 1.0 }
Если дв[Nbout].a33 = 1    : Если задвижка заблокирована
{ рв[Nrvpr].a7 = 1.0 }
Если дв[Nrout] = 0        : Если задвижка в местном управлении
{ рв[Nrvpr].a7 = 1.0 }
Если вд[Neout] = 0        : Если нет напряжения на задвижке
{ рв[Nrvpr].a7 = 1.0 }

```

```

рв[Nrvpr] = рв[Nrvpr] + рв[Nrvpr].a7 + рв[Nrvpr].a6 }

```

Иначе

```

{ рв[Nrvpr] = 5.0 }

```

```

Если пл[Nwon] = 1          : Если параметры насоса аварийны
{ Если пл[Nqec] = 0
  { Если вд[Nh] = 1          : Если включен насос
    { Если дв[NA] = 0
      { message("ЗАЩИТА ОБОРУДОВАНИЯ НАСОСА !")
        Если вд[Ncogl] = 1
          { message("Давление охлаждающей жидкости минимум") }
      }
    }
  }
}

```

```
Если вд[Nfogh] = 1
{ message("Температура охлаждающей жидкости (максимум)") }
Если вд[NkogI] = 1
{ message("Уровень уплотнительной жидкости (минимум)") }
Если вд[Npnag] = 1
{ message("Давление на нагнетании (минимум)") }
Если вд66 = 1
{ message("Давление в уравнительной линии") }
Если пл[Ntpl] = 1
{ message("Температура подшипников насоса (максимум)") }}
```

дв[NA] = 1 }}} : Выключить насос при аварии

```
Если пл[Nwon] = 0
{ пл[Nqec] = 0 }}
```

КОНЕЦ

В.1.9 Пример 9. Нарботка оборудования

Задачи расчета наработки различного оборудования (насосы, вентиляторы и т.д.) часто встречаются в производстве. Ниже приведен пример расчета месячной и годовой наработки высоковольтных насосов.

ПРОГРАММА НАРАБОТКА

НАЧАЛО

```
: Программа - пример наработка оборудования
: : рв[нарabM]      Нарботка насоса за текущий месяц
: рв[нарabY].а6     Нарботка насоса за текущий год
: вд[Non]           Состояние включенности насоса (концевик)
Если пц6 # 1
{ пл2 = 0 }
Вкл тм1           : минутный счетчик
: ч м с
(пц1,пц2,пц3)=stime() : Определяем текущее время
: год мес день
(пц4,пц5,пц6)=sdate() : Определяем текущую дату
```

Если тм1 > 1.0

```
{ пв2 = 1           : Флаг новая минута = 1
```

```
  тм1 = тм1-1.0 }
```

```
:Насос  1 2 3 4 5 6
```

```
Для Non[2,4,6,8,10,12],
```

```
  нарabM[1...6],
```

```
  нарabY[7...12]
```

```
{
```

```
  Если вд[Non] = 1           : Если насос включен
```

```
  {
```

```
    Если пв2 = 1           : Если новая минута
```

```
    {
```

```
      рв[нарabM] = рв[нарabM] + 0.016666 : Нарботка, ч
```

```
      рв[нарabY] = рв[нарabY] + 0.016666 : Нарботка, ч
```

```
    }
```

```
  }
```

```
  Если пц6 = 1           : Если первое число месяца
```

```
  {
```

```
    Если пц1 = 0           : 0 часов
```

```
    {
```

```
      Если пц2 = 5           : 5 минут
```

```
      {
```

```
        рв[нарabM] = 0.0      : Сброс месячного счетчика
```

```
        пл2 = 1
```

```
        Если пц5 = 1           : Если месяц - январь
```

```
        {
```

```
          рв[нарabY] = 0      : Сброс годового счетчика
```

```
        }
```

```
      }
```

```
    }
```

```
  }
```

```
}
```

```
пв2 = 0           : Флаг новая минута = 0
```

КОНЕЦ

В.1.10 Пример 10. Расчет накапливаемых значений за различные отчетные интервалы

Реализация на станции оператора расчета накапливаемых значений за различные отчетные интервалы времени рассматривается на примере программы учёта теплоносителя и тепловой энергии, в которой осуществляется накопление за час, сутки, месяц, год.

Тотальные сумматоры теплоносителя и тепловой энергии реализованы в контроллере и здесь не рассматриваются.

Для каждой переменной (интервального сумматора) в генераторе базы данных назначается перо в «часовом» самописце. Эти перья могут быть использованы при создании видеокадров печатного документа «Суточный отчет» и «Отчет за месяц»

ПРОГРАММА MAIN

НАЧАЛО

```
Если ва137.а52=0
{
  Если пл1=0
  {
    пв1=pause()
    Если пв1>60
    {
      рв337=33.0
    }
    пл1=1
  }
}
```

Расчёт выполняется только в том случае, если есть связь с контроллером. Атрибут а52 – сигнализация по достоверности (0 – связь есть, 1 –связь отсутствует).

Осуществляется чтение времени простоя станции оператора. Если данное время заведомо больше часа (60 мин), то формируется значение переменной рв337, которая используется в процедуре обнуления часовых счётчиков и контроля сброса остальных сумматоров (см. процедуру TIME). Данный участок программы выполняется единожды при перезапуске станции оператора.

TIME

```
Если пц501<59
{
  ...
}
```

Данное условие разрешает выполнение обработки значений сумматоров только в интервале до 59 секунд. Синхронизация КРУГОЛА с записью в тренды

Для

```
Sm1[103...119], : Тотальный счётчик массы
Sm2[301...317], : Часовой счётчик массы
Sm3[335...351], : Суточный счётчик массы
Sm4[369...385], : Месячный счётчик массы
Sm5[403...419], : Годовой счётчик массы
St1[120...136], : Тотальный счётчик тепла
St2[318...334], : Часовой счётчик тепла
St3[352...368], : Суточный счётчик тепла
St4[386...402], : Месячный счётчик тепла
St5[420...436], : Годовой счётчик счётчик массы тепла
aM[301...317], : Архив счётчика массы
aQ[318...334] : Архив счётчика тепла
{
  Если вд51=1 : Проверка признака обнуления часовых счётчиков (см. процедуру TIME)
  {
    ва[Sm2]=0 : Обнуление часовых счётчиков массы
    ва[St2]=0 } : Обнуление часовых счётчиков тепла
  Если вд52=1 : Проверка признака обнуления суточных счётчиков
  {
    ва[Sm3]=0 : Обнуление суточных счётчиков массы
    ва[St3]=0 } : Обнуление суточных счётчиков тепла
  Если вд53=1 : Проверка признака обнуления месячных счётчиков
  {
    ва[Sm4]=0 : Обнуление месячных счётчиков массы
    ва[St4]=0 } : Обнуление месячных счётчиков тепла
  Если вд54=1 : Проверка признака обнуления годовых счётчиков
  {
    ва[Sm5]=0 : Обнуление годовых счётчиков массы
    ва[St5]=0 } :Обнуление годовых счётчиков тепла
}
```

Значение тотальных счётчиков берётся из контроллера

Ниже реализована проверка обнуления тотального счётчика в контроллере, где 100000 – терминатор значений счётчика, 99000– значение тотального счётчика после которого разрешен сброс счётчиков.

```

Если ва[Sm1]<рв[аМ] : Проверяет меньше ли текущее значение тотального счётчика
                     : предыдущего значения
    { Если рв[аМ]>99000.0 : Проверяет - предыдущее значение больше предела после
                        : которого следует выполнять данную защиту
        { рв[аМ]=рв[аМ]-100000.0 } }
Если ва[St1]<рв[аQ]
    { Если рв[аQ]>99000.0
        { рв[аQ]=рв[аQ]-100000.0 } }

```

```

пв101=ва[Sm1]-рв[аМ]
рв[аМ]=ва[Sm1]
пв102=ва[St1]-рв[аQ]
рв[аQ]=ва[St1]

```

: Приращение между текущим ва[Sm1] и архивным значением рв[аМ] тотального счётчика

```

ва[Sm2]=ва[Sm2]+пв101
ва[Sm3]=ва[Sm3]+пв101
ва[Sm4]=ва[Sm4]+пв101
ва[Sm5]=ва[Sm5]+пв101
ва[St2]=ва[St2]+пв102
ва[St3]=ва[St3]+пв102
ва[St4]=ва[St4]+пв102
ва[St5]=ва[St5]+пв102
}

```

По приращению организованны собственно сами счётчики соответственно за час, за сутки, за месяц и за год. В генераторе базы данных на данные сумматоры необходимо назначить перья, после чего их (перья сумматоров) можно использовать для создания печатных документов.

```

вд51=0
вд52=0
вд53=0
вд54=0
}
}

```

После того, как все необходимые счётчики обнулялись, признак обнуления каждого типа счётчика возвращается в исходное состояние

КОНЕЦ

ПРОЦЕДУРА TIME : Проверяет наступление времени обнуления каждого типа счётчиков
НАЧАЛО : и формирует соответствующий признак на обнуление

```

(пц503,пц502,пц501)=stime()
(пц506,пц505,пц504)=sdate()
пв501=цв(пц501) :сек
пв502=цв(пц502) :мин
пв503=цв(пц503) :час
пв504=цв(пц504) :день
пв505=цв(пц505) :месяц
пв506=цв(пц506) :год
пв507=пв505+пв504/100 : месяц_день

```

: Формируется признак обнуления, если запомненное значение часа, в предыдущем сеансе обнуления, будет отличаться от текущего.

```

Если пв503#рв337
    { вд51=1

```

Далее аналогично и для остальных интервальных счётчиков

```

Если пв504#рв338 : Для суточного сумматора
    { рв338=пв504
      вд52=1 }

```

Если пв505#рв339 : Для месячного сумматора

```
{ рв339=пв505  
  вд53=1 }
```

Если пв506#рв340 : Для годового сумматора

```
{ рв340=пв506  
  вд54=1 }}
```

КОНЕЦ

В.1.11 Пример 11. Формирование печатных документов

Печатные документы в системе, на базе SCADA КРУГ-2000 формируются из программы пользователя. В данном примере рассмотрена программа формирования режимных листов и документов произвольной формы (наработка оборудования).

Программа ПД

Начало

Если пл1 = 0 : Флаг первого прохода

```
{ пл1 = 1
```

 : Индексация

```
  пц101 = 1 : сменная ведомость
```

```
  пц102 = 2 : суточная ведомость
```

```
  пц103 = 3 : двухчасовка
```

```
  пц104 = 4 : наработка электрооборудования
```

```
}
```

: год мес день

```
(пц1,пц2,пц3)=sdate()
```

: час мин сек

```
(пц4,пц5,пц6)=stime()
```

:Проверка времени на формирование двухчасовой ведомости

Если пл2 = 0

```
{ пл2 = 1
```

```
  пв3 = цв(пц4)
```

```
  пв3 = пв3/2
```

```
  (пв4,пв5)=modf(пв3)
```

```
  Если пв5 = 0.0
```

```
  { пц7 = пц4 + 2 }
```

```
  Иначе
```

```
  { пц7 = пц4 + 1 }
```

```
}
```

РЛИСТЫ

Конец

Процедура РЛИСТЫ : Формирование сменных и суточных отчетов

Начало

```
.*****
```

: Сменная ведомость

Если пц4 < 9

```
{ Если пл101 = 0
```



```

{ Если пц4 = 8          : Время формирования 8 часов
  { Если пц5 = 5        : 5 минут
    { рв91 = 1.0 }}}    : Флаг формирования 8 часовой сменной ведомости
Если пц4 > 19
{ Если пл101 = 0
  { Если пц4 = 20      : Время формирования 20 часов
    { Если пц5 = 5      : 5 минут
      { рв91 = 1.0 }}} : Флаг формирования 20 часовой сменной ведомости
Если пц4 = 9
{ пл101 = 0 }

Если пц4 = 21
{ пл101 = 0 }

: Суточная ведомость

Если пл102 = 0
{ Если пц4 = 8          : Время формирования 8 часов
  { Если пц5 = 5        : 5 минут
    { рв92 = 1.0 }}}    : Флаг формирования суточной ведомости

Если пц4 > 8
{ пл102 = 0 }

: Двухчасовка

Если пц4 = 22
{ пц7 = 0 }

пц8 = пц7 - пц4
Если пл103 = 0
{ Если пц8 = 0
  { Если пц5 = 5        : 5 минут
    { рв93 = 1.0
      пц7 = пц7 + 2 }}}
пц9 = пц7 - 1

Если пц4 = пц9
{ пл103 = 0 }

: Нарботка

Если пл104 = 0
{ Если пц3 = 1          : 1 - ое число месяца
  { Если пц5 = 5        : 5 минут
    { рв94 = 1.0 }}}    : Флаг формирования отчета по наработке

Если пц3 > 1
{ пл104 = 0 }

:Формирование и печать сменной ведомости
Если рв91 = 1.0        : Флаг формирования сменной ведомости
{ рв91 = 0.0
  пл101 = 1

```

```
пц105 = ппд(пц101)
message("Сформирован печатный документ - Сменная ведомость")

Если рв91.а6 = 1.0
{ пц109 = пчпд(пц101,1) }

Если пц105 # 0
{ message("ОШИБКА формирования печатный документ - Сменная ведомость ") }
Если пц109 # 0
{ message("ОШИБКА печати печатный документ - Сменная ведомость ") }
}

:Формирование и печать суточной ведомости
Если рв92 = 1.0 : Флаг формирования суточной ведомости
{ рв92 = 0.0
  пл102 = 1
  пц106 = ппд(пц102)
  message("Сформирован печатный документ - Суточной ведомости")

  Если рв92.а6 = 1.0
  { пц110 = пчпд(пц102,1) }

  Если пц106 # 0
  { message("ОШИБКА формирования печатный документ - Суточной ведомости") }
  Если пц110 # 0
  { message("ОШИБКА печати печатный документ - Суточной ведомости") }
}

:Формирование и печать двухчасовой ведомости
Если рв93 = 1.0 : Флаг формирования двухчасовой ведомости
{ рв93 = 0.0
  пл103 = 1
  пц107 = ппд(пц103)
  message("Сформирован печатный документ - Двухчасовой ведомости")

  Если рв93.а6 = 1.0
  { пц111 = пчпд(пц103,1) }

  Если пц107 # 0
  { message("ОШИБКА формирования печатный документ - Двухчасовой ведомости") }
  Если пц111 # 0
  { message("ОШИБКА печати печатный документ - Двухчасовой ведомости") }
}

:Формирование и печать ПД - Нарботка электрооборудования
Если рв94 = 1.0 : Флаг формирования ПД - Нарботка электрооборудования
{ рв94 = 0.0
  пл104 = 1
  пц108 = ппд(пц104)
  message("Сформирован печатный документ - Нарботка электрооборудования")

  Если рв94.а6 = 1.0
```

```
{ пц112 = пчпд(пц104,1) }  
  
Если пц108 # 0  
{ message("ОШИБКА формирования печатный документ - Нарботка  
электрооборудования") }  
Если пц112 # 0  
{ message("ОШИБКА печати печатный документ - Нарботка электрооборудования") }  
}  
  
Конец
```

В.1.12 Пример 12 Формирование точек трендов

Пользователь может создавать, хранить и корректировать уже имеющиеся целевые, сменные и суточные тренды, с помощью функции записи в тренды. Пример решения данной задачи приведен в программе IZM.

ПРОГРАММА IZM : Определение средних суточных значений

НАЧАЛО

пц10= 1

пц11= 0

Для

J1[1...12]

{ пв[J1]= 0 }

Для : Считывание часовых значений за сутки

J2[1...24]

{
пц11= пц11+ 1 : Формирование условного час суток для каждого прохода

пв1=чтТр(3,2,пц10,пц11)

ПВ2=чтТр(3,3,пц10,пц11)

ПВ3=чтТр(3,4,пц10,пц11)

ПВ4=чтТр(3,5,пц10,пц11)

Для : Суммы средних значений [rz1]

пр1[1...4],rz1[5...8], : и число слагаемых суммы [sl1]

sl1[9...12]

{
Если пв[пр1]> 0
{ пв[rz1]= пв[rz1]+ пв[пр1]
пв[sl1]= пв[sl1]+ 1 }

}

}

Для : Средние значения за сутки

rz2[5...8],sl2[9...12]

{
Если пв[sl2]> 0 : Проверка на ноль знаменателя
{ пв[rz2]= пв[rz2]/ пв[sl2] }

}

(пц1, пц2, пц3)=sdate()

пц4= пц3- пц10

пц12= 2 : Начальный номер пера в суточном самописце

Для : Запись искомых точек в суточные тренды

J3[5...8]

{
зпТр(5,пц12, 0, пц4, пв[J3]) : Формирование номера переменной для каждого прохода

}

КОНЕЦ

С. ПРИМЕРЫ ПРОГРАММЫ ФБД. ПРИЛОЖЕНИЕ С (Справочное).

С.1 Пример

С.1.1 Пример создания схемы ФБД

Реализация блокировок на отключение насосов Н301-1,Н301-2 (основного и резервного) подачи бензина в атмосферную колонну.

Прекращение подачи сырья при возникновении аварийных ситуаций. Схема реализована в контроллере.

Предусмотрены следующие управляющие сигналы:

- ключ режима управления: автомат/ручное (0/1) ВД91,92;
- ключ: выключить насос ДВ12,13;

Входные аналоговые сигналы:

- Температура в картере насоса ВА16,17;
- Уровень в емкости ВА928;

Входные дискретные сигналы:

- Уровень разделительной жидкости ВД55,56;
- Давление нагнетания низкое ВД49,50;
- Давление разделительной жидкости ВД47,48;
- Состояние насоса (включен) ВД73,74;

Выходные дискретные сигналы:

- Выключить насос ДВ12,13;
- Сигнализация «авария» ВД95,96.

В схеме должна быть предусмотрена выработка сигнала отключения в следующих случаях:

- при появлении любого из перечисленных входных дискретных сигналов, причем по сигналам ВД47,ВД48 должна быть предусмотрена задержка на срабатывание длительностью 15с на время набора давления при включении насоса;
- при снижении уровня в сосуде для разделительной жидкости ниже значения, указанного в качестве нижней аварийной границы по этому параметру;
- при повышении давления в сосуде с разделительной жидкостью выше заданного верхнего аварийного значения;

При выработке сигнала останова должен формироваться сигнал «авария» для сигнализации на станции оператора. Блокировка не должна работать в режиме ручного управления.

Возможный вариант реализации блокировок приведен на рисунке С.1. ввиду необходимости использования оператором условия основную программу удобно разбить на подпрограммы или процедуры, в каждой из которых реализуется алгоритмы с использованием операторов «Для».

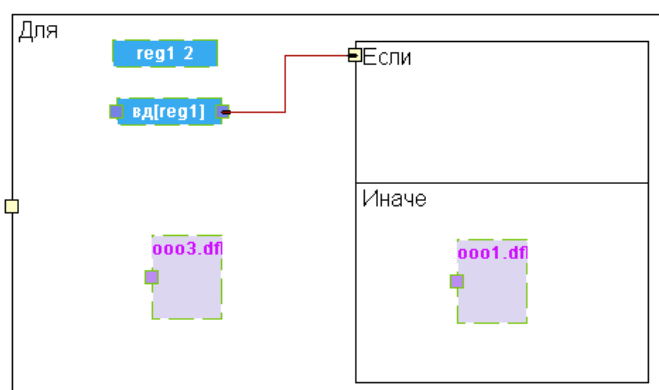


Рисунок С.1

Основная программа ооо обеспечивает безусловное выполнение процедуры ооо3, осуществляющую блокировку сигнала: «низкое давление на выкиде насоса» в течение 15 секунд после пуска, а также выполнение условия по состоянию сигнала «режим». В случае автоматического режима работы насоса, т.е. если переменные массива режима работы reg1 =0, выполняется процедура ооо1, осуществляющая непосредственно выработку сигнала «останов». При единичном значении переменной режима происходит выход из программы. Процедура ооо3 приведена на рисунке С.2.

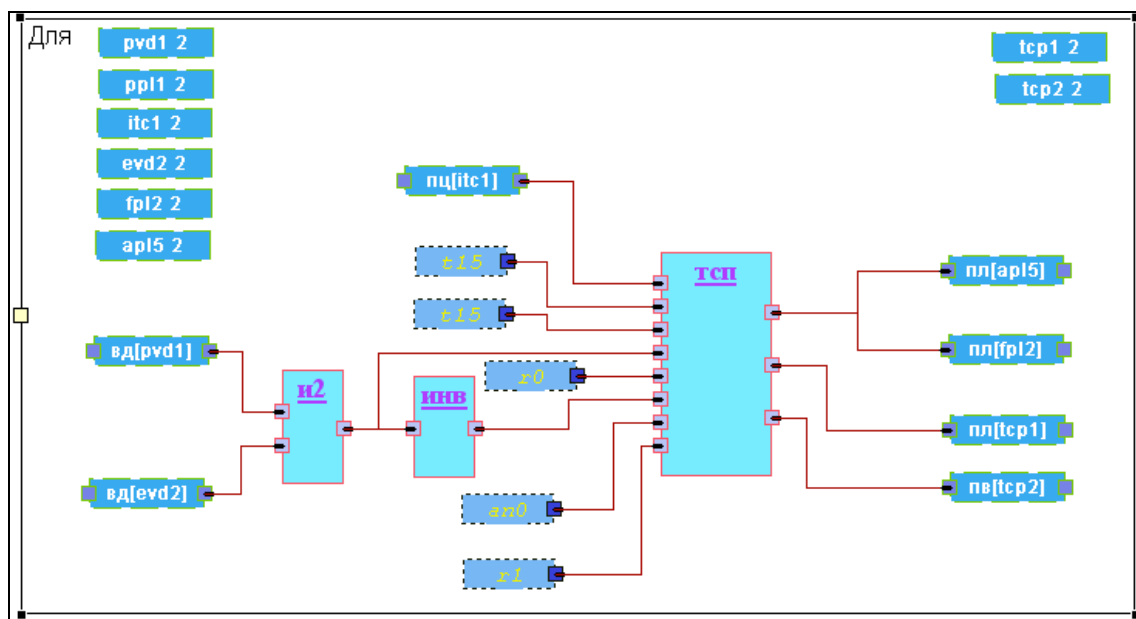


Рисунок С.2

Значения уставок заданы с помощью констант: аналоговых t15, an0, логических r0. Номер программируемого таймера задан с помощью массива целой переменной itc1.e15. Неиспользуемые выходные сигналы таймера привязаны к массивам: логическому пл[tcp1] и вещественному пв[tcp2].

Процедура ооо1 приведена на рисунке С.3.

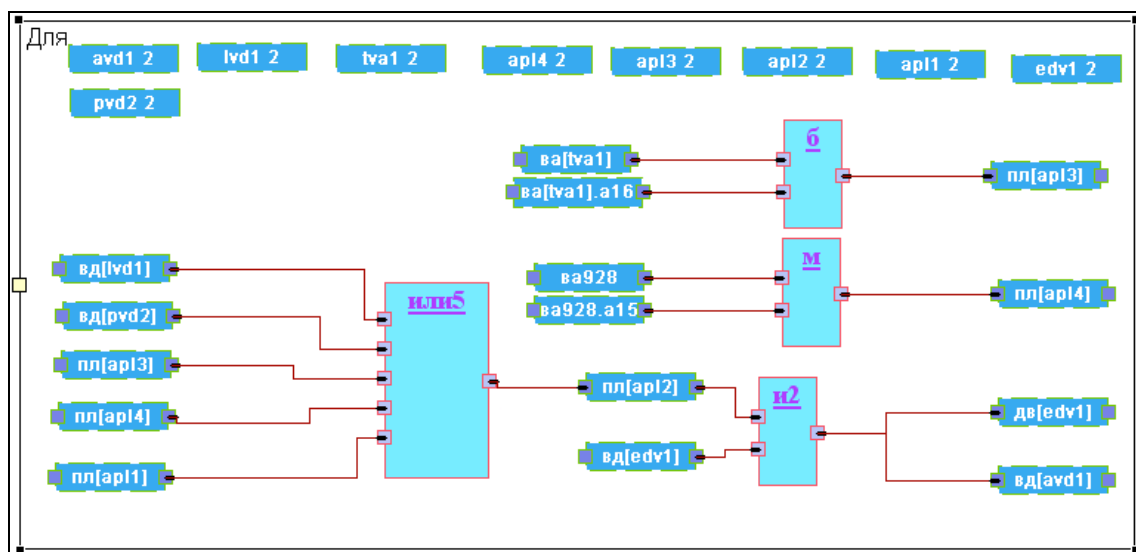


Рисунок С.3

Передача промежуточных значений переменных от одной процедуры к другой выполняется с помощью массивов, описывающих одни и те же переменные. В данном случае массивы `apl1` и `apl5` используют одинаковые логические переменные 20,21.

В процедуре `ooo1` вырабатывается сигнал – «стоп насоса», описываемым массивом `edv1`, сигнал «авария», определяемым массивом `adv1` при одновременном наличии сигналов «насос включен» (массив `edv1`) и суммарном сигнале «аварийная ситуация» (массив `apl2`). Формирование аварийных сигналов при превышении давления (массив `apl3`) и понижении уровня (массив `apl4`) происходит при помощи функций «больше» и «меньше» путем сравнения значения переменной с верхней (атрибут `.a16`) и нижней (атрибут `.a15`) границами аварийной сигнализации для каждой из переменных.

D. СООБЩЕНИЯ ТРАНСЛЯТОРА ЯЗЫКА КРУГОЛ . ПРИЛОЖЕНИЕ D (Справочное).

D.1 Сообщения об ошибках

Ниже приведена сводная таблица сообщений транслятора об обнаруженных ошибках.

Номер ошибки	Текст сообщения
Ошибка 2	Имя программы '<имя программы>' не является уникальным.
Ошибка 3	Имя процедуры '<имя процедуры>' не является уникальным.
Ошибка 4	Имя функции '<имя функции>' не является уникальным.
Ошибка 5	При объявлении переменной использован недопустимый тип данных.
Ошибка 6	Идентификатор '<имя переменной>', используемый при объявлении переменной, не является уникальным.
Ошибка 7	Несоответствие типов данных при инициализации переменной.
Ошибка 8	При объявлении константы использован недопустимый тип данных.
Ошибка 9	Идентификатор '<имя константы>', используемый при объявлении константы, не является уникальным.
Ошибка 10	Несоответствие типов данных при задании значения константы.
Ошибка 11	Константе не присвоено значение.
Ошибка 12	При объявлении массива использован недопустимый тип данных.
Ошибка 13	Идентификатор '<имя массива>', используемый при объявлении массива, не является уникальным.
Ошибка 14	Недопустимый размер массива.
Ошибка 15	Отсутствует объявление массива '<имя массива>'.
Ошибка 16	Значение индекса выходит за пределы массива '<имя массива>'.
Ошибка 17	Недопустимый индекс массива.
Ошибка 18	Необъявленный идентификатор '<строка символов>'.
Ошибка 19	Недопустимый индекс переменной языка КРУГОЛ.
Ошибка 20	Деление на ноль в арифметическом выражении.
Ошибка 21	Несоответствие типов данных в выражении.
Ошибка 22	В выражении использован недопустимый вызов функции, возвращающей несколько значений.
Ошибка 23	Несоответствие типов данных в операторе присваивания.
Ошибка 24	Несоответствие количества значений, возвращаемых функцией, и количества переменных списка.
Ошибка 25	Отсутствует функция с именем '<имя функции>'.
Ошибка 26	Несоответствие количества параметров, передаваемых функции.
Ошибка 27	Несоответствие типов данных при передаче параметров функции.
Ошибка 28	Для отсутствующего параметра функции не задано значение по умолчанию.
Ошибка 29	Отсутствует процедура с именем '<имя процедуры>'.
Ошибка 30	Оператор последовательности содержит переменную с именем '<имя переменной>', не являющимся уникальным.
Ошибка 32	Списки значений, принимаемых переменными оператора последовательности, должны содержать одинаковое количество элементов.
Ошибка 33	Оператор ПРЕРВАТЬ использован за пределами оператора последовательности.
Ошибка 34	Условное выражение должно возвращать значение логического типа.

Номер ошибки	Текст сообщения
Ошибка 35	Синтаксическая ошибка.
Ошибка 36	Синтаксическая ошибка объявления программы.
Ошибка 37	Синтаксическая ошибка объявления процедуры.
Ошибка 38	Синтаксическая ошибка объявления функции.
Ошибка 39	Синтаксическая ошибка в секции объявления входных переменных функции.
Ошибка 40	Синтаксическая ошибка в секции объявления выходных переменных функции.
Ошибка 41	Синтаксическая ошибка в секции объявления констант.
Ошибка 42	Синтаксическая ошибка в секции объявления переменных.
Ошибка 43	При объявлении константы использован недопустимый тип данных.
Ошибка 44	При объявлении константы использован недопустимый идентификатор.
Ошибка 45	Синтаксическая ошибка при задании значения константы.
Ошибка 46	При объявлении переменной использован недопустимый тип данных.
Ошибка 47	При объявлении переменной использован недопустимый идентификатор.
Ошибка 48	Синтаксическая ошибка при инициализации переменной.
Ошибка 49	Синтаксическая ошибка в списке входных параметров функции.
Ошибка 50	Синтаксическая ошибка в списке выходных параметров функции.
Ошибка 51	Синтаксическая ошибка в списке переменных оператора последовательности.
Ошибка 52	Синтаксическая ошибка при задании списка значений переменной оператора последовательности.
Ошибка 53	Синтаксическая ошибка в условном выражении.
Ошибка 54	Неподключенный элемент ПЕРЕМЕННАЯ.
Ошибка 55	Неподключенный элемент КОНСТАНТА.
Ошибка 56	Неподключенный вход элемента ЕСЛИ.
Ошибка 59	Рекурсивный вызов процедуры или функции.
Ошибка 60	Переменная '<тип><номер>' отсутствует в базе данных контроллера.
Ошибка 61	Переменная '<тип><номер>' отсутствует в базе данных станции оператора.
Ошибка 62	Переменные '<тип переменных>' не содержат атрибута с номером <номер>.
Ошибка 63	Не запущен сервер базы данных.
Ошибка 67	Недопустимый параметр, используемый для задания номера алгоблока.
Ошибка 69	Для выбранной версии ядра КРУГОЛ не поддерживается использование глобальных переменных.
Ошибка 70	Для выбранной версии ядра КРУГОЛ не поддерживается использование глобальных массивов.
Ошибка 71	Для выбранной версии ядра КРУГОЛ не поддерживается индексация локальных массивов переменной оператора последовательности.
Ошибка 72	Для выбранной версии ядра КРУГОЛ не поддерживается чтение значения переменной оператора последовательности.
Ошибка 74	Имя ФБД схемы '<имя схемы>' не является уникальным.
Ошибка 75	Значение номера внутренней переменной КРУГОЛ '<тип><номер>' выходит за допустимые пределы.
Ошибка 76	Значение индекса '<имя индексного операнда>' при обращении к внутренним переменным КРУГОЛ '<тип переменных>' выходит за допустимые пределы.
Ошибка 77	Синтаксическая ошибка при объявлении массива.
Ошибка 78	Синтаксическая ошибка при обращении к элементу массива с помощью индекса.

Номер ошибки	Текст сообщения
Ошибка 79	Синтаксическая ошибка при обращении к переменной КРУГОЛ с помощью индекса.
Ошибка 80	Синтаксическая ошибка в форматной строке элемента СООБЩЕНИЕ.
Ошибка 81	Константное значение выходит за допустимые пределы.
Ошибка 82	Присваивание значения константе не является допустимым.
Ошибка 83	Неподключенный вход элемента СООБЩЕНИЕ.
Ошибка 84	Недоступна база данных системных переменных контроллера.
Ошибка 85	Не удастся открыть файл <имя файла>
Ошибка 86	В выражении использован недопустимый вызов функции, не возвращающей значений.
Ошибка 87	Синтаксическая ошибка объявления алгоблока.
Ошибка 88	Имя алгоблока '<имя алгоблока>' не является уникальным.
Ошибка 89	Пользовательские алгоблоки не поддерживаются выбранной платформой.
Ошибка 90	Неподключенный элемент ВЫРАЖЕНИЕ.
Ошибка 91	Синтаксическая ошибка в элементе ВЫРАЖЕНИЕ.
Ошибка 92	Элемент подключен к контактам разных типов данных.
Ошибка 93	Недопустимое соединение контактов неопределенного типа данных.
Ошибка 94	Автоматическое формирование номера алгоблока не поддерживается выбранной платформой.
Ошибка 95	Количество номеров системных переменных превысило максимально допустимое значение.
Ошибка 96	При объявлении использован тип данных, не поддерживаемый выбранной платформой.
Ошибка 97	Слишком много обращений к алгоблоку '<имя алгоблока>'.
Ошибка 98	Невозможно создать переменную для хранения состояния линии обратной связи, т.к. тип данных выходного контакта не может быть однозначно определен.
Ошибка 99	Невозможно создать переменную для хранения результата вызова функции, т.к. тип данных выходного контакта не может быть однозначно определен.

Ошибка 2

Текст сообщения:

Имя программы '<имя программы>' не является уникальным.

Описание:

Данное сообщение выдается при совпадении использованного при определении программы имени с каким-либо другим именем в глобальной области видимости.

Ошибка 3

Текст сообщения:

Имя процедуры '<имя процедуры>' не является уникальным.

Описание:

Данное сообщение выдается при совпадении использованного при определении процедуры имени с каким-либо другим именем в глобальной области видимости.

Ошибка 4

Текст сообщения:

Имя функции '<имя функции>' не является уникальным.

Описание:

Данное сообщение выдается при совпадении использованного при определении функции имени с каким-либо другим именем в глобальной области видимости.

Ошибка 5

Текст сообщения:

При объявлении переменной использован недопустимый тип данных.

Описание:

Данное сообщение выдается при использовании в объявлении переменной спецификатора типа данных, отличного от следующих:

- Лог - логический тип данных;
- Цел, Цел16 - целочисленный 16-битовый тип данных;
- Цел8 - целочисленный 8-битовый тип данных;
- Цел32 - целочисленный 32-битовый тип данных;
- Вещ, Вещ32 - вещественный 32-битовый тип данных;
- Вещ64 - вещественный 64-битовый тип данных.

Ошибка 6

Текст сообщения:

Идентификатор '<имя переменной>', используемый при объявлении переменной, не является уникальным.

Описание:

Данное сообщение выдается при совпадении имени переменной, использованного при объявлении, с каким-либо другим именем.

Ошибка 7

Текст сообщения:

Несоответствие типов данных при инициализации переменной.

Описание:

Данное сообщение выдается при попытке присвоить переменной начальное значение, тип данных которого не совпадает с типом данных переменной, указанным в объявлении. Примером такой ситуации является инициализация переменной логического типа целочисленным значением, отличным от нуля или единицы.

Ошибка 8

Текст сообщения:

При объявлении константы использован недопустимый тип данных.

Описание:

Данное сообщение выдается при использовании в объявлении константы спецификатора типа данных, отличного от следующих:

- Лог - логический тип данных;
- Цел, Цел16 - целочисленный 16-битовый тип данных;
- Цел8 - целочисленный 8-битовый тип данных;
- Цел32 - целочисленный 32-битовый тип данных;
- Вещ, Вещ32 - вещественный 32-битовый тип данных;
- Вещ64 - вещественный 64-битовый тип данных.

Ошибка 9

Текст сообщения:

Идентификатор '<имя константы>', используемый при объявлении константы, не является уникальным.

Описание:

Данное сообщение выдается при совпадении имени константы, использованного при объявлении, с каким-либо другим именем.

Ошибка 10

Текст сообщения:

Несоответствие типов данных при задании значения константы.

Описание:

Данное сообщение выдается при попытке присвоить константе значение, тип данных которого не совпадает с типом данных, указанным в объявлении. Примером такой ситуации является использование вещественного значения при объявлении константы целочисленного типа.

Ошибка 11

Текст сообщения:

Константе не присвоено значение.

Описание:

Данное сообщение выдается в случае отсутствия инициализации константы при ее объявлении. В этом сообщении указывается на необходимость обязательного присвоения константе какого-либо значения.

Ошибка 12

Текст сообщения:

При объявлении массива использован недопустимый тип данных.

Описание:

Данное сообщение выдается при использовании в объявлении массива спецификатора типа данных, отличного от следующих:

- Лог - логический тип данных;
- Цел, Цел16 - целочисленный 16-битовый тип данных;
- Цел8 - целочисленный 8-битовый тип данных;
- Цел32 - целочисленный 32-битовый тип данных;
- Вещ, Вещ32 - вещественный 32-битовый тип данных;
- Вещ64 - вещественный 64-битовый тип данных.

Ошибка 13

Текст сообщения:

Идентификатор '<имя массива>', используемый при объявлении массива, не является уникальным.

Описание:

Данное сообщение выдается при совпадении имени массива, использованного при объявлении, с каким-либо другим именем.

Ошибка 14

Текст сообщения:

Недопустимый размер массива.

Описание:

Данное сообщение выдается при задании размера массива отрицательным или нулевым значением.

Ошибка 15

Текст сообщения:

Отсутствует объявление массива '<имя массива>'.

Описание:

Данное сообщение выдается при использовании в качестве операнда элемента массива, который не был предварительно объявлен.

Ошибка 16

Текст сообщения:

Значение индекса выходит за пределы массива '<имя массива>'.

Описание:

При обращении к элементу массива использовано недопустимое значение индекса. В качестве индекса должно использоваться положительное значение в диапазоне от единицы до значения размера массива, указанного при его объявлении.

Ошибка 17

Текст сообщения:

Недопустимый индекс массива.

Описание:

Для индексации элементов массива допускается использование переменных оператора последовательности, пользовательских констант или непосредственно заданных константных значений. В остальных случаях выдается данное сообщение об ошибке.

Ошибка 18

Текст сообщения:

Необъявленный идентификатор '<строка символов>'.

Описание:

Данное сообщение выдается при обнаружении транслятором операнда, для которого отсутствует объявление соответствующей константы, переменной или переменной оператора последовательности.

Ошибка 19

Текст сообщения:

Недопустимый индекс переменной КРУГОЛ.

Описание:

Для индексации системных и внутренних переменных КРУГОЛ допускается использование только переменных оператора последовательности, при этом значения переменных должны находиться в диапазоне от 1 до 9999. В остальных случаях выдается данное сообщение об ошибке.

Ошибка 20

Текст сообщения:

Деление на ноль в арифметическом выражении.

Описание:

Данное сообщение выдается при использовании нулевого константного значения в качестве правого операнда операции деления.

Ошибка 21

Текст сообщения:

Несоответствие типов данных в выражении.

Описание:

Данное сообщение выдается при использовании в выражении операндов с разными типами данных или с типом данных, неподдерживаемым оператором. Например, при попытке выполнить операцию логического И над вещественными операндами.

Ошибка 22

Текст сообщения:

В выражении использован недопустимый вызов функции, возвращающей несколько значений.

Описание:

При вычислении выражений допускается использование вызовов функций, возвращающих только одно значение, т.к. при этом не возникает неоднозначности с выбором одного из нескольких результатов.

Ошибка 23**Текст сообщения:**

Несоответствие типов данных в операторе присваивания.

Описание:

Данное сообщение выдается, если операнд-источник и операнд-приемник имеют разные типы данных.

Ошибка 24**Текст сообщения:**

Несоответствие количества значений, возвращаемых функцией, и количества переменных списка.

Описание:

Данное сообщение выдается, если количество переменных списка, которым присваивается результат вызова функции, не совпадает с количеством значений, возвращаемых функцией.

Ошибка 25**Текст сообщения:**

Отсутствует функция с именем '<имя функции>'.

Описание:

Сообщение выдается при попытке вызова несуществующей функции, т.е. которая не определена пользователем и отсутствует в библиотеке функций КРУГОЛ.

Ошибка 26**Текст сообщения:**

Несоответствие количества параметров, передаваемых функции.

Описание:

Данное сообщение выдается, если количество параметров функции, указанных при ее вызове, не совпадает с количеством параметров, которое она принимает в действительности.

Ошибка 27**Текст сообщения:**

Несоответствие типов данных при передаче параметров функции.

Описание:

Данное сообщение выдается при несовпадении типа данных операнда, используемого в качестве параметра функции, и типа данных в объявлении соответствующей входной переменной пользовательской функции или типа данных параметра в описании библиотечной функции.

Ошибка 28**Текст сообщения:**

Для отсутствующего параметра функции не задано значение по умолчанию.

Описание:

Данное сообщение выдается, если при вызове функции отсутствует какой-либо из параметров, а при объявлении соответствующей входной переменной пользовательской функции (или при описании библиотечной функции) не задано значение по умолчанию.

Ошибка 29

Текст сообщения:

Отсутствует процедура с именем '<имя процедуры>'.

Описание:

Сообщение выдается при попытке вызова несуществующей процедуры, т.е. не определенной пользователем.

Ошибка 30

Текст сообщения:

Оператор последовательности содержит переменную с именем '<имя переменной>', не являющимся уникальным.

Описание:

Данное сообщение выдается при объявлении переменной оператора последовательности, имя которой совпадает с каким-либо другим именем.

Ошибка 32

Текст сообщения:

Списки значений, принимаемых переменными оператора последовательности, должны содержать одинаковое количество элементов.

Описание:

Данное требование обусловлено тем, что количеством переменных определяется количество циклов выполнения оператора последовательности.

Ошибка 33

Текст сообщения:

Оператор ПРЕРВАТЬ использован за пределами оператора последовательности.

Описание:

Данный оператор должен использоваться только внутри оператора последовательности, поскольку предназначен для досрочного завершения его выполнения.

Ошибка 34

Текст сообщения:

Условное выражение должно возвращать значение логического типа.

Описание:

Данное требование обусловлено тем, что условное выражение обеспечивает выбор оператором условия одного из двух возможных вариантов.

Ошибка 35

Текст сообщения:

Синтаксическая ошибка.

Описание:

Данное сообщение выдается при несоответствии транслируемой программы правилам грамматики технологического языка КРУГОЛ.

Ошибка 36

Текст сообщения:

Синтаксическая ошибка объявления программы.

Описание:

Сообщение выдается при несоответствии объявления программы правилам грамматики технологического языка КРУГОЛ.

Ошибка 37

Текст сообщения:

Синтаксическая ошибка объявления процедуры.

Описание:

Сообщение выдается при несоответствии объявления процедуры правилам грамматики технологического языка КРУГОЛ.

Ошибка 38

Текст сообщения:

Синтаксическая ошибка объявления функции.

Описание:

Сообщение выдается при несоответствии объявления функции правилам грамматики технологического языка КРУГОЛ.

Ошибка 39

Текст сообщения:

Синтаксическая ошибка в секции объявления входных переменных функции.

Описание:

Сообщение выдается при несоответствии объявления входных переменных функции правилам грамматики технологического языка КРУГОЛ.

Ошибка 40

Текст сообщения:

Синтаксическая ошибка в секции объявления выходных переменных функции.

Описание:

Сообщение выдается при несоответствии объявления выходных переменных функции правилам грамматики технологического языка КРУГОЛ.

Ошибка 41

Текст сообщения:

Синтаксическая ошибка в секции объявления констант.

Описание:

Сообщение выдается при несоответствии объявления констант правилам грамматики технологического языка КРУГОЛ.

Ошибка 42

Текст сообщения:

Синтаксическая ошибка в секции объявления переменных.

Описание:

Сообщение выдается при несоответствии объявления переменных правилам грамматики технологического языка КРУГОЛ.

Ошибка 43

Текст сообщения:

При объявлении константы использован недопустимый тип данных.

Описание:

Сообщение выдается, если тип данных, использованный при объявлении константы, не предусмотрен правилами грамматики технологического языка КРУГОЛ.

Ошибка 44

Текст сообщения:

При объявлении константы использован недопустимый идентификатор.

Описание:

Сообщение выдается, если идентификатор (имя) константы не соответствует правилам грамматики технологического языка КРУГОЛ.

Ошибка 45

Текст сообщения:

Синтаксическая ошибка при задании значения константы.

Описание:

Сообщение выдается, если для инициализации константы используется конструкция, которая не соответствует правилам грамматики технологического языка КРУГОЛ.

Ошибка 46

Текст сообщения:

При объявлении переменной использован недопустимый тип данных.

Описание:

Сообщение выдается, если тип данных, использованный при объявлении переменной, не предусмотрен правилами грамматики технологического языка КРУГОЛ.

Ошибка 47

Текст сообщения:

При объявлении переменной использован недопустимый идентификатор.

Описание:

Сообщение выдается, если идентификатор (имя) переменной не соответствует правилам грамматики технологического языка КРУГОЛ.

Ошибка 48

Текст сообщения:

Синтаксическая ошибка при инициализации переменной.

Описание:

Сообщение выдается, если для инициализации переменной используется конструкция, которая не соответствует правилам грамматики технологического языка КРУГОЛ.

Ошибка 49

Текст сообщения:

Синтаксическая ошибка в списке входных параметров функции.

Описание:

Сообщение выдается при неправильном с точки зрения грамматики задании параметров вызова функции.

Ошибка 50

Текст сообщения:

Синтаксическая ошибка в списке выходных параметров функции.

Описание:

Сообщение выдается при неправильном с точки зрения грамматики задании списка переменных, которым присваивается результат вызова функции, возвращающей несколько значений.

Ошибка 51

Текст сообщения:

Синтаксическая ошибка в списке переменных оператора последовательности.

Описание:

Сообщение выдается при несоответствии списка объявлений переменных оператора последовательности правилам грамматики технологического языка КРУГОЛ.

Ошибка 52

Текст сообщения:

Синтаксическая ошибка при задании списка значений переменной оператора последовательности.

Описание:

Сообщение выдается при несоответствии списка значений переменной оператора последовательности правилам грамматики технологического языка КРУГОЛ.

Ошибка 53

Текст сообщения:

Синтаксическая ошибка в условном выражении.

Описание:

Сообщение выдается при неправильной с точки зрения грамматики записи условного выражения оператора условия.

Ошибка 54

Текст сообщения:

Неподключенный элемент ПЕРЕМЕННАЯ.

Описание:

Сообщение выдается при трансляции ФБД схемы, в которой имеется элемент ПЕРЕМЕННАЯ, к которому не подключено ни одной линии связи.

Ошибка 55

Текст сообщения:

Неподключенный элемент КОНСТАНТА.

Описание:

Сообщение выдается при трансляции ФБД схемы, в которой имеется элемент КОНСТАНТА, к которому не подключено ни одной линии связи.

Ошибка 56

Текст сообщения:

Неподключенный вход элемента ЕСЛИ.

Описание:

Сообщение выдается при трансляции ФБД схемы, в которой имеется элемент ЕСЛИ, к которому не подключено ни одной линии связи.

Ошибка 59

Текст сообщения:

Рекурсивный вызов процедуры или функции.

Описание:

Сообщение выдается при обнаружении транслятором циклической цепочки вызовов процедур или функций.

Например:

Процедура Proc1

Начало

Proc2

Конец

Процедура Proc2

Начало

Proc3

Конец

Процедура Proc3

Начало

Proc1

Конец

Ошибка 60

Текст сообщения:

Переменная '<тип><номер>' отсутствует в базе данных контроллера.

Описание:

Сообщение выдается в случае обращения к системной переменной, которая не описана в системной базе данных контроллера. Данная проверка выполняется при включенном режиме контроля использования системных переменных. Включение режима задается в свойствах проекта, при этом также указывается путь к файлам, в которых описана база данных системных переменных контроллера.

Ошибка 61

Текст сообщения:

Переменная '<тип><номер>' отсутствует в базе данных станции оператора.

Описание:

Сообщение выдается в случае обращения к системной переменной, которая не описана в системной базе данных станции оператора. Данная проверка выполняется при включенном режиме контроля использования системных переменных. Включение режима задается в свойствах проекта. Для обеспечения доступа к системной базе данных необходимо произвести запуск сервера базы данных.

Ошибка 62

Текст сообщения:

Переменные '<тип переменных>' не содержат атрибута с номером <номер>.

Описание:

Сообщение выдается при обращении к несуществующему атрибуту системной переменной.

Ошибка 63

Текст сообщения:

Не запущен сервер базы данных.

Описание:

Сообщение выдается при трансляции программ проекта, в настройках которого включен режим контроля использования системных переменных, целевой платформой является станция оператора и при этом не запущен сервер базы данных. Для нормальной трансляции программ необходимо запустить сервер базы данных.

Ошибка 67

Текст сообщения:

Недопустимый параметр, используемый для задания номера алгоблока.

Описание:

Для задания номера алгоблока (как правило, передаваемого в качестве первого параметра) допускается использовать целочисленные константы или переменные оператора последовательности. В остальных случаях выдается данное сообщение об ошибке.

Ошибка 69

Текст сообщения:

Для выбранной версии ядра КРУГОЛ не поддерживается использование глобальных переменных.

Описание:

Сообщение выдается при использовании глобальных переменных в проектах, предназначенных для устаревших версий среды исполнения.

Ошибка 70

Текст сообщения:

Для выбранной версии ядра КРУГОЛ не поддерживается использование глобальных массивов.

Описание:

Сообщение выдается при использовании глобальных массивов в проектах, предназначенных для устаревших версий среды исполнения.

Ошибка 71

Текст сообщения:

Для выбранной версии ядра КРУГОЛ не поддерживается индексация локальных массивов переменной оператора последовательности.

Описание:

Сообщение выдается при косвенном обращении к элементам массива с помощью переменной оператора последовательности в проектах, предназначенных для устаревших версий среды исполнения.

Ошибка 72

Текст сообщения:

Для выбранной версии ядра КРУГОЛ не поддерживается чтение значения переменной оператора последовательности.

Описание:

Сообщение выдается при использовании переменной оператора последовательности в качестве операнда в проектах, предназначенных для устаревших версий среды исполнения.

Ошибка 74

Текст сообщения:

Имя ФБД схемы '<имя схемы>' не является уникальным.

Описание:

Сообщение выдается при совпадении имени программы, процедуры или функции, представленной данной схемой, с каким-либо другим именем в глобальной области видимости проекта.

Ошибка 75

Текст сообщения:

Значение номера внутренней переменной КРУГОЛ '<тип><номер>' выходит за допустимые пределы.

Описание:

Сообщение выдается при обращении к несуществующей внутренней переменной КРУГОЛ.

Ошибка 76

Текст сообщения:

Значение индекса '<имя индексного операнда>' при обращении к внутренним переменным КРУГОЛ '<тип переменных>' выходит за допустимые пределы.

Описание:

Сообщение выдается в случае косвенного обращения к внутренним переменным КРУГОЛ с помощью переменной оператора последовательности, среди значений которой имеются номера несуществующих переменных.

Ошибка 77

Текст сообщения:

Синтаксическая ошибка при объявлении массива.

Описание:

Сообщение выдается при несоответствии объявления массива правилам грамматики технологического языка КРУГОЛ.

Ошибка 78

Текст сообщения:

Синтаксическая ошибка при обращении к элементу массива с помощью индекса.

Описание:

Сообщение выдается при неправильном, с точки зрения грамматики языка КРУГОЛ, обращении к элементу массива.

Ошибка 79

Текст сообщения:

Синтаксическая ошибка при обращении к переменной КРУГОЛ с помощью индекса.

Описание:

Сообщение выдается при неправильном, с точки зрения грамматики языка КРУГОЛ, обращении к системной или внутренней переменной КРУГОЛ.

Ошибка 80

Текст сообщения:

Синтаксическая ошибка в форматной строке элемента СООБЩЕНИЕ.

Описание:

Форматная строка ФБД элемента СООБЩЕНИЕ должна соответствовать правилам записи параметров функции message при обращении к ней из программ на языке СТ. В противном случае выдается данное сообщение об ошибке.

Ошибка 81

Текст сообщения:

Константное значение выходит за допустимые пределы.

Описание:

При непосредственном задании константных значений используются следующие правила:

1. Значения 0 и 1 в зависимости от контекста могут рассматриваться как значения логического, любого целочисленного или вещественного типов.
2. Значения в диапазоне от -128 до 127 могут рассматриваться как значения целочисленного 8-битового, целочисленного 16-битового, целочисленного 32-битового, вещественного 32-битового или вещественного 64-битового типов.

3. Значения в диапазоне от -32768 до 32767 могут рассматриваться как значения целочисленного 16-битового, целочисленного 32-битового, или вещественного 32-битового или вещественного 64-битового типов.
4. Значения в диапазоне от -2147483648 до 2147483647 могут рассматриваться как значения целочисленного 32-битового, вещественного 32-битового или вещественного 64-битового типов.
5. Значения в диапазоне от 1.175494351e-38 до 3.402823466e+38 рассматриваются как значения вещественного 32-битового или 64-битового типа.
6. Значения в диапазоне от 1.7976931348623158e+308 до 2.2250738585072014e-308 рассматриваются как значения вещественного 64-битового типа.

Если диапазон значений константы не соответствует ни одному из приведенных правил, выдается данное сообщение об ошибке.

Ошибка 82

Текст сообщения:

Присваивание значения константе не является допустимым.

Описание:

Данное сообщение выдается при попытке присваивания значения константе. Значение константы задается при ее инициализации и доступно в дальнейшем только для чтения.

Ошибка 83

Текст сообщения:

Неподключенный вход элемента СООБЩЕНИЕ.

Описание:

Сообщение выдается при трансляции ФБД схемы, в которой имеется элемент СООБЩЕНИЕ, к которому не подключено ни одной линии связи.

Ошибка 84

Текст сообщения:

Недоступна база данных системных переменных контроллера.

Описание:

Данное сообщение выдается в случае невозможности открытия транслятором файлов базы данных системных переменных контроллера, путь к которым указывается в свойствах проекта. Необходимость получения доступа к файлам базы данных контроллера возникает при трансляции программ проекта, в настройках которого включен режим контроля использования системных переменных и целевой платформой является контроллер.

Ошибка 85

Текст сообщения:

Не удается открыть файл <имя файла>

Описание:

Сообщение выдается при неудачной попытке открытия файла транслятором.

Ошибка 86

Текст сообщения:

В выражении использован недопустимый вызов функции, не возвращающей значений.

Описание:

Сообщение выдается при попытке использования в качестве операнда вызова функции, которая не имеет выходных параметров.

Ошибка 87

Текст сообщения:

Синтаксическая ошибка объявления алгоблока.

Описание:

Сообщение выдается при несоответствии объявления алгоблока правилам грамматики технологического языка КРУГОЛ.

Ошибка 88

Текст сообщения:

Имя алгоблока '<имя алгоблока>' не является уникальным.

Описание:

Данное сообщение выдается при совпадении использованного при определении алгоблока имени с каким-либо другим именем в глобальной области видимости.

Ошибка 89

Текст сообщения:

Пользовательские алгоблоки не поддерживаются выбранной платформой.

Описание:

Данное сообщение выдается при попытке трансляции алгоблоков пользователя в проекте, целевой платформой которого являются СРВК версии менее чем 7.1 или среды исполнения КРУГОЛ версии менее 2.1, предназначенной для станции оператора. Выбор целевой платформы производится в диалоге свойств проекта.

Ошибка 90

Текст сообщения:

Неподключенный элемент ВЫРАЖЕНИЕ.

Описание:

Данное сообщение выдается в случае неподключенного выхода элемента ФБД схемы, обеспечивающего вычисление выражения на языке СТ.

Ошибка 91

Текст сообщения:

Синтаксическая ошибка в элементе ВЫРАЖЕНИЕ.

Описание:

Данное сообщение выдается, если выражение, записанное в элементе, не соответствует правилам грамматики языка СТ.

Ошибка 92

Текст сообщения:

Элемент подключен к контактам разных типов данных.

Описание:

Данное сообщение выдается, если выход элемента с неопределенным типом данных подключен к нескольким входным контактам с несовпадающими типами данных. В данной ситуации транслятор не может принять однозначное решение о реальном типе данных выходного контакта. Для устранения ошибки необходимо использовать подключение выхода к контактам с одинаковым типом данных.

Ошибка 93

Текст сообщения:

Недопустимое соединение контактов неопределенного типа данных.

Описание:

Данное сообщение выдается при подключении выходного контакта неопределенного типа данных к входному контакту также неопределенного типа данных. При этом

транслятору невозможно определить реальный тип данных. Для устранения данной ошибки необходимо использовать подключение контакта неопределенного типа данных (выходного или выходного) только с контактом определенного типа данных.

Ошибка 94

Текст сообщения:

Автоматическое формирование номера алгоблока не поддерживается выбранной платформой.

Описание:

Данное сообщение выдается при отсутствии явного задания номера вызываемого алгоблока в проекте, целевой платформой которого являются СРВК версии менее чем 7.1 или среда исполнения КРУГОЛ версии менее 2.1, предназначенной для станции оператора. Выбор целевой платформы производится в диалоге свойств проекта.

Ошибка 95

Текст сообщения:

Количество номеров системных переменных превысило максимально допустимое значение.

Описание:

Такое сообщение выдается при использовании отдельной программой (а также процедурами, функциями и алгоблоками, необходимыми данной программе) количества номеров системных переменных большего, чем разрешено ключом защиты. Подсчитываются переменные с разными номерами и разными типами данных.

Ошибка 96

Текст сообщения:

При объявлении использован тип данных, не поддерживаемый выбранной платформой.

Описание:

Данное сообщение выдается в случае, когда при объявлении используются типы данных, поддержка которых была добавлена в более новых версиях среды исполнения и в настройках проекта не была выбрана соответствующая платформа. К таким типам относятся Цел8, Цел32, Вещ64, которые поддерживаются СРВК версии не ниже 8.0 и средой исполнения КРУГОЛ не ниже 2.2. Поскольку новые ключевые слова Цел16 и Вещ32 эквивалентны типам данных Цел и Вещ, их можно использовать и для более старых платформ.

Ошибка 97

Текст сообщения:

Слишком много обращений к алгоблоку '<имя алгоблока>'.

Описание:

Данное сообщение выдается при автонумерации алгоблоков, если превышено максимально допустимое количество используемых отдельной программной единицей одинаковых алгоблоков с именем, указанным в сообщении. При анализе возможных причин выдачи данного сообщения необходимо в первую очередь учитывать, что при использовании автонумерации внутри оператора последовательности на каждой итерации будет использован отдельный экземпляр алгоблока. Таким образом, количество используемых алгоблоков соответствует количеству итераций текущего оператора последовательности (независимо от уровня его вложенности в другие операторы последовательности). Обращение к алгоблоку без указания номера за пределами оператора последовательности приводит к использованию только одного экземпляра алгоблока. Необходимо отметить, что данное сообщение связано с принципиальными ограничениями возможностей алгоритма автонумерации и что при программировании с использованием алгоблоков нужно учитывать то количество

алгоблоков, которое может предоставить среда исполнения контроллера или станции оператора.

Ошибка 98

Текст сообщения:

Невозможно создать переменную для хранения состояния линии обратной связи, т.к. тип данных выходного контакта не может быть однозначно определен.

Описание:

Данное сообщение выдается при неудачной попытке транслятора ФБД объявить переменную для хранения состояния линии обратной связи. Такая ситуация возникает при неопределенном типе данных выходного контакта элемента ФБД, от которого идет линия обратной связи и наличии неоднозначности при попытке определить тип данных переменной на основе анализа типа данных входных контактов, подключенных к выходу.

Ошибка 99

Текст сообщения:

Невозможно создать переменную для хранения результата вызова функции, т.к. тип данных выходного контакта не может быть однозначно определен.

Описание:

Такое сообщение выдается при наличии одного или нескольких соединений с выходным контактом элемента ФБД, имеющим неопределенный тип данных. Выдача сообщения происходит в случае, когда транслятору ФБД необходимо создание переменной для временного хранения состояния выхода и при этом невозможно однозначно определить тип данных этой переменной на основе анализа типа данных входных контактов, подключенных к выходу. Примером ситуации, в которой требуется создание временной переменной, может быть вызов функции с несколькими подключенными к выходу линиями связи. При этом обеспечивается однократный вызов функции и последующее многократное использование его результата.

D.2 Предупреждение транслятора

Предупреждение 1001

Текст предупреждения:

Константное условие в условном операторе.

Описание:

Поскольку константное условие может быть проверено на этапе трансляции, транслятор выполняет оптимизацию условного оператора с константным условием и выдает текст данного предупреждения.

Предупреждение 1002

Текст предупреждения:

Пустой список операторов в условном операторе.

Описание:

Транслятор выполняет оптимизацию условного оператора, содержащего пустой список операторов, с выдачей текста данного предупреждения.

Предупреждение 1003

Текст предупреждения:

Пустой список операторов в операторе последовательности.

Описание:

Транслятор выполняет оптимизацию оператора последовательности, содержащего пустой список операторов, с выдачей текста данного предупреждения.

Предупреждение 1004

Текст предупреждения:

Переменные оператора последовательности принимают одно значение.

Описание:

Поскольку в данном случае нет необходимости в организации цикла, выполняется оптимизация оператора последовательности и выдается текст данного предупреждения.

Предупреждение 1005

Текст предупреждения:

Повторное использование номера алгоблока '<номер алгоблока>'.

Описание:

Данное предупреждение выдается, если алгоблок несколько раз вызывается с одним и тем же номером. Такое предупреждение позволяет отследить нежелательные вызовы алгоблоков с одинаковым номером.

Предупреждение 1006

Текст предупреждения:

Номер алгоблока не является константой или переменной оператора последовательности.

Описание:

При использовании в качестве номера алгоблока константы или переменной оператора последовательности их значения известны на этапе трансляции и у транслятора имеется возможность анализа этих значений. Значения остальных операндов становятся известны только на этапе выполнения и могут привести к серьезным ошибкам в программе.