

ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«РАКУРС-ИНЖИНИРИНГ»



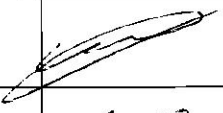



Заказчик: ООО «Ракурс-инжиниринг»

Договор №: В14.2020

Руководство пользователя

Библиотека программных модулей для реализации
типовых функций ПТК «Апогей-М»

САНКТ-ПЕТЕРБУРГ
2021

Наименование организации, предприятия	Должность исполнителя	Фамилия, инициалы	Подпись	Дата
ООО «Ракурс-инжиниринг»	Разработал	Лебедев С. А.		12.02.21
	Проверил	Малышев Б. М.		15.02.21
	Нормоконтроль	Алексеев С. Д.		15.02.21
	Утвердил	Фенрих М. В.		17.02.21

Настоящее руководство устанавливает основные правила эксплуатации и назначение программного обеспечения, входящего в библиотеку программных модулей для реализации типовых функций ПТК «Апогей-М».

Содержание

1	Структура ПО	6
2	Разделы библиотеки	7
2.1	nGeneral	7
2.2	nTime	9
2.3	nArh	10
2.4	nPsw	11
2.5	nDI	12
2.6	nDO.....	14
2.7	nAI.....	16
2.8	nAO.....	22
2.9	nEmg.....	23
2.10	nProtect	26
2.11	nDSP	30
2.12	nPolinom.....	31
2.13	nSettings.....	32
2.14	nValve.....	36
2.15	nCommon	40
3	Жизненный цикл ПО	43
3.1	Требования к эксплуатационному персоналу	43
3.2	Устранение неисправностей.....	43
3.3	Техническая поддержка.....	43
3.4	Жизненный цикл ПО.....	44

Программное обеспечение, входящее в библиотеку программных модулей для реализации типовых функций ПТК «Апогей-М (далее ПО) позволяет выполнить:

- прием и обработку команд от устройств человеко-машинного интерфейса (ЧМИ);
- формирование очереди истории событий и их передачу в устройства ЧМИ;
- первичную обработку величин, полученных от модулей дискретных входов;
- первичную обработку величин, полученных от модулей аналоговых входов;
- подготовку данных для передачи в модули дискретных выходов;
- подготовку данных для передачи в модули аналоговых выходов;
- формирование списков предупредительной и аварийной сигнализации, обобщенную сигнализацию;
- обработку аналоговых сигналов, интерполяцию;
- преобразование типов данных;
- получение аналоговых и дискретных уставок с устройств ЧМИ;
- управление электромеханическими устройствами (задвижка, клапан, насос).

В рамках этих модулей определены структуры данных, предоставляемых ЧМИ, кодирование команд ЧМИ и событий.

1 СТРУКТУРА ПО

Функции входящие в библиотеку распределены на несколько разделов по функциональной принадлежности.

В раздел обычно входит:

- определение типов используемых данных;
- функции и функциональные блоки;
- определение конфигурационных констант, определяющих режим работы ПО;

Перечень разделов приведен в табл.1.1.

Таблица 1.1.

№	Наименование	Описание
1	nGeneral	Определение основных типов данных, базовые функции и функции преобразования типов
2	nTime	Определение типов данных, функций и функциональных блоков, используемых для работы со временем.
3	nArh	Определение типов данных и функций для реализации механизма истории событий ПТК «Апогей-М»
4	nPsw	Определение типов данных и функций проверки уровня доступа. Используется как универсальный интерфейс между программными модулями и различными подсистемами идентификации пользователей.
5	nDI	Определение типов данных и функций для первичной обработки величин, полученных от модулей дискретных входов.
6	nDO	Определение типов данных и функций подготовки данных для передачи в модули дискретных выходов.
7	nAI	Определение типов данных и функций для первичной обработки величин, полученных от модулей аналоговых входов.
8	nAO	Определение функций подготовки данных для передачи в модули аналоговых выходов.
9	nEmg	Определение типов данных и функций для формирования сигналов индивидуальной и обобщенной сигнализации.
10	nProtect	Определение типов данных и функций для формирования сигналов технологических защит.
11	nDSP	Определение функций для реализации цифровых фильтров сигналов.
12	nPolinom	Определение типов данных и функций для решения задач интерполяции.
13	nSettings	Определение типов данных и функций для реализации интерфейса ввода аналоговых и дискретных уставок.
14	nValve	Определение типов данных и функций для реализации задач управления электромеханическими устройствами.
15	nCommon	Определение типов данных и функций для реализации приема обобщенных команд ПТК «Апогей-М»

ПО реализовано на языке Structured Text по ГОСТ Р МЭК 61131-3-2016.

2 РАЗДЕЛЫ БИБЛИОТЕКИ

2.1 nGeneral

В данном разделе определяются типы данных, функции и функциональные блоки используемые практически во всех разделах библиотеки.

Зависимостей от функций других разделов ПО нет.

Перечень типов данных приведен в табл.2.1.1.

Таблица 2.1.1.

Наименование	Тип	Описание
tCmdHMI	STRUCT	Структура приема команд ЧМИ
tCommonCmd	STRUCT	Флаги обобщенных команд
uByte	UNION	Объединение типов на основе BYTE
uWord	UNION	Объединение типов на основе WORD
uDWord	UNION	Объединение типов на основе DWORD
uLWord	UNION	Объединение типов на основе LWORD

Перечень функций (Ф) и функциональных блоков (ФБ) приведен в табл.2.1.2.

Таблица 2.1.2.

№	Наименование	Тип	Интерфейс
Триггеры с внешней памятью (для синхронизации состояния в резервном ПЛК)			
1	fbRS RS-триггер с внешней памятью (функциональный блок)	ФБ	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • Set: BOOL; // Вход Set триггера • Reset1: BOOL; // Вход Reset триггера (приоритетный) <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> • Q1: BOOL; // Выход триггера <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> • extQ: BOOL; // Внешняя переменная состояния триггера
2	fbSR SR-триггер с внешней памятью (функциональный блок)	ФБ	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • Set1: BOOL; // Вход Set триггера (приоритетный) • Reset: BOOL; // Вход Reset триггера <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> • Q1: BOOL; // Выход триггера <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> • extQ: BOOL; // Внешняя переменная состояния триггера
3	fRS RS-триггер с внешней памятью (функция)	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • Set: BOOL; // Вход Set триггера • Reset1: BOOL; // Вход Reset триггера (приоритетный)

№	Наименование	Тип	Интерфейс
			<u>Выход (Return)</u> <ul style="list-style-type: none"> • fRS: BOOL; // Выход триггера
			<u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> • extQ: BOOL; // Внешняя переменная состояния триггера
4	fSR SR-триггер с внешней памятью (функциональный блок)	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • Set1: BOOL; // Вход Set триггера (приоритетный) • Reset: BOOL; // Вход Reset триггера
			<u>Выход (Return)</u> <ul style="list-style-type: none"> • fSR: BOOL; // Выход триггера
			<u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> • extQ: BOOL; // Внешняя переменная состояния триггера
Преобразование типов данных			
5	AryBoolToWords Преобразование массива BOOL в массив WORD	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • pAryBool: DWORD; // Указатель на массив BOOL • pAryWord: DWORD; // Указатель на массив WORD • WordCount: UINT; // Количество слов для переноса
6	BoolArrayToWord Преобразование массива из 16 BOOL в WORD	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • pBoolArray: POINTER TO ARRAY[0..15] OF BOOL; // Указатель на массив из 16 BOOL
			<u>Выход (Return)</u> <ul style="list-style-type: none"> • BoolArrayToWord: WORD; // Итоговая величина
7	CopyDwordToReal Интерпретируем DWORD как Real	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • In: DWORD; // Данные для интерпретации как REAL
8	CopyRealToDword Интерпретируем Real как DWORD	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • In: REAL; // Данные для интерпретации как DWORD
			<u>Выход (Return)</u> <ul style="list-style-type: none"> • CopyRealToDword: DWORD; // Итоговая величина
Проверка достоверности			
9	CheckRealCom Проверка REAL на	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • In: REAL; // Число в формате REAL для проверки

№	Наименование	Тип	Интерфейс
	достоверность (число, NaN, +Inf, -Inf)		<u>Выходы (Outputs)</u> <ul style="list-style-type: none"> • Nan: BOOL; // Не число • PosInfinite: BOOL; // Положительная бесконечность • NegInfinite: BOOL; // Отрицательная бесконечность <u>Выход (Return)</u> <ul style="list-style-type: none"> • CheckRealCom: BOOL; // Итоговая величина, TRUE -- если число, FALSE -- если не число
10	CheckLRealCom Проверка LREAL на достоверность (число, NaN, +Inf, -Inf)	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • In: LREAL; // Число в формате LREAL для проверки <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> • Nan: BOOL; // Не число • PosInfinite: BOOL; // Положительная бесконечность • NegInfinite: BOOL; // Отрицательная бесконечность <u>Выход (Return)</u> <ul style="list-style-type: none"> • CheckRealCom: BOOL; // Итоговая величина, TRUE -- если число, FALSE -- если не число

2.2 nTime

В данном разделе определяются типы данных, функции и функциональные блоки, используемые для работы со временем.

Зависимостей от функций других разделов ПО нет.

Перечень типов данных приведен в табл.2.2.1.

Таблица 2.2.1.

Наименование	Тип	Описание
tDateTime	STRUCT	Структура даты и времени в формате ПЛК Omron CJ

Перечень функций приведен в табл.2.2.2.

Таблица 2.2.2.

№	Наименование	Тип	Интерфейс
Функции преобразования величин времени			
1	TimeRefresh Получение даты и времени часов ПЛК в формате Omron CJ	Ф	<u>Входы (Inputs)</u> Нет <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> • RDT: tDateTime; // Дата и время в формате Omron CJ

№	Наименование	Тип	Интерфейс
2	LocalTimeRefresh Получение даты и времени часов ПЛК в формате Omron CJ с учетом часового пояса	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> TimeShift: REAL; // Часовой пояс
			<u>Выходы (Outputs)</u> <ul style="list-style-type: none"> RDT: tDateTime; // Дата и время в формате Omron CJ
3	DateTimeToLocal Коррекция времени в формате Omron CJ с учетом часового пояса	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> inRDT: tDateTime; // Дата и время в формате Omron CJ TimeShift: REAL; // Часовой пояс
			<u>Выходы (Outputs)</u> <ul style="list-style-type: none"> outRDT: tDateTime; // Дата и время в формате Omron CJ
4	FSecToTIME Преобразование числа секунд, представленных в формате REAL в формат TIME	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> Second: REAL; // Количество секунд
			<u>Выход (Return)</u> <ul style="list-style-type: none"> FSecToTIME: TIME; // Результат преобразования

2.3 nArh

В данном разделе определяются типы данных, функции и функциональные блоки используемые для реализации истории событий ПТК «Апогей-М».

Поскольку период считывания данных с ПЛК существенно больше, чем период выполнения программы, то существует риск пропустить короткие события или потерять правильную последовательность наступления событий.

Для того, чтобы гарантированно записать все события и сохранить последовательность их возникновения, разработан специальный механизм.

Каждое событие кодируется записью, в которую входят следующие поля:

- Дата и время возникновения события
- ID группы событий, например «аналоговые сигналы»
- ID элемента группы, например «Датчик №1»
- ID события, например «Изменение нижнего предела измерения»
- Данные 1, например «Старое значение нижнего предела измерения»
- Данные 2, например «Новое значение нижнего предела измерения»
- Некоторые другие, например ID источника событий и т. п.

При наступлении события вызывается функция ArhWrite, в которую передаются данные для формирования такой записи, затем данная запись помещается в конец очереди FIFO.

Для реализации обмена с устройствами верхнего уровня используется функция ArhRead.

При наличии данных в очереди, и отсутствии данных в выходном буфере, в последний переносится порция данных и число, определяющее количество записей, которое должно быть прочитано устройством верхнего уровня.

После считывания, обнулением числа записей в выходном буфере устройство верхнего уровня подтверждает, что данные прочитаны и буфер освобождается.

Если очередь событий при этом оказывается не пустая, то операция повторяется. Если новых данных нет, то ничего не происходит до их появления.

Новые записи добавляются в конец очереди.

Для использования необходимы функции из разделов nGeneral, nTime.

Перечень типов данных приведен в табл.2.3.1.

Таблица 2.3.1.

Наименование	Тип	Описание
tHistRecord	STRUCT	Единичная запись истории событий
tHistory	STRUCT	Структура истории событий
tHistBuffer	STRUCT	Структура буфера чтения для ЧМИ

Перечень функций и функциональных блоков приведен в табл.2.3.2.

Таблица 2.3.2.

№	Наименование	Тип	Интерфейс
Функции для работы с очередью истории событий			
1	ArhWrite Добавление новой записи в очередь истории событий	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> ActTime: BOOL; // Флаг необходимости использования метки времени, переданной извне (TRUE -- использовать, FALSE – присваивать метку времени, регистрируя момент записи) <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> Hist: tHistory; // Блок данных дискретной истории NewRec: tHistRecord; // Блок данных новой записи
2	ArhRead Функция выдачи архивных данных на ЧМИ	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> PntID: UINT; // Идентификатор указателя чтения <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> Hist: tHistory; // Блок данных дискретной истории NewRec: tHistRecord; // Блок данных новой записи

2.4 nPsw

Типы данных и функции проверки уровня доступа. Используется как универсальный интерфейс между программными модулями и различными подсистемами идентификации пользователей.

Для использования необходимы функции из разделов nGeneral.

Перечень типов данных приведен в табл.2.4.1.

Таблица 2.4.1.

Наименование	Тип	Описание
tPsw	STRUCT	Уровень доступа пользователя

Перечень функций приведен в табл.2.4.2.

Таблица 2.4.2.

№	Наименование	Тип	Интерфейс
Интерфейсные функции работы с уровнем доступа пользователя			
1	PswCheck Функция сравнения маски фактического уровня доступа и маски слова управления доступом	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • CurrentLevel: WORD; // Маска фактического уровня доступа • Mask: WORD; // Маска требуемого уровня доступа <u>Выход (Return)</u> <ul style="list-style-type: none"> • PswCheck: BOOL; // TRUE – действие разрешено, FALSE – действие запрещено
2	PswSet Формирование битовой маски доступа из уровня доступа	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • Level: UINT; // Уровень доступа <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> • Psw: tPsw; // Уровень доступа (битовая маска)

2.5 nDI

Функции для первичной обработки величин, полученных от модулей дискретных входов.

Набор функций nDI предназначен для:

- записи в очередь событий информации об изменении состояния входных дискретных сигналов;
- реализации функции подмены дискретных входов;
- реализации функции антидребезга;
- обработки пользовательских команд от ЧМИ;
- записи в очередь событий информации об исполнении пользовательских команд;
- подготовки данных для ЧМИ;
- контроля правильности срабатывания перекидного контакта (пара входов NO/NC).

Для использования необходимы функции из разделов nGeneral, nPsw, nArh, nTime.

Перечень типов данных приведен в табл.2.5.1.

Таблица 2.5.1.

Наименование	Тип	Описание
tDI	STRUCT	Массив данных модуля обработки дискретных входов
tDIValue	UNION	Обрабатываемый массив значений дискретных входов
tDISet	STRUCT	Уставки модуля обработки дискретных входов
tDIValueOut	ALIAS	Выходной массив значений дискретных входов

Перечень функций и функциональных блоков приведен в табл.2.5.2.

Таблица 2.5.2.

№	Наименование	Тип	Интерфейс
Функции работы с дискретными сигналами			
1	DIMain Первичная обработка входных дискретных сигналов	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> AllowMask: ARRAY[0..15] OF WORD; // Уровни доступа команд ЧМИ <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> DI: tDI; // Основной массив данных модуля обработки дискретных сигналов DISet: tDISet; // Уставки модуля обработки дискретных сигналов Hist: tHistory; // Массив данных модуля истории событий CmdHMI: tCmdHMI; // Команда ЧМИ Psw: tPsw; // Массив данных модуля разграничения доступа
2	DIErrorStatus Проверка достоверности дискретного сигнала, подключенного парой NO/NC контактов	ФБ	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> DIOpen: BOOL; // Нормально-открытый вход DIClose: BOOL; // Нормально-закрытый вход SwitchTime: TIME; // Время переключения <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> ErrorStatus: BOOL; // Сигнал недостоверен
3	DIOut Преобразование упакованного массива входов (BitString) в массив BOOL	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> WordCount: UINT; // Количество слов для переноса <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> DI: tDI; // Основной массив данных модуля обработки дискретных сигналов Out: tDIValueOut; // Выходной массив

Перечень конфигурационных констант приведен в таблице 2.5.3.

Таблица 2.5.3.

Наименование	Тип	Описание
cDIWordCount	UINT	Количество обрабатываемых слов (16 бит), размерности массивов
cDIWordCountReal	UINT	Количество фактически обрабатываемых слов (16 бит)
cDIBaseUnit	UINT	Базовый тип выходного массива
AllowMask	ARRAY[0..15] OF WORD	Уровни доступа команд nDI

Перечень пользовательских команд приведен в табл. 2.5.4.

Номер группы команд: 30.

Смысл поля номер в группе: порядковый номер дискретного входа.

Таблица 2.5.4.

Номер команды	Название команды	Тип поля «Данные команды»
1	Разрешить архивацию	-
2	Запретить архивацию	-
3	Изменить уставку фильтра	REAL
4	Включить подмену входа	-
5	Отключить подмену входа	-
6	Установить значение подмененного состояния в '1'	-
7	Установить значение подмененного состояния в '0'	-

Перечень регистрируемых событий приведен в табл. 2.5.5.

Номер группы событий: 30.

Смысл поля «Номер в группе»: порядковый номер дискретного входа.

Таблица 2.5.5.

Номер события	Название события
0	Уход сигнала
1	Возникновение сигнала
4	Запись в архив включена
5	Запись в архив отключена
6	Уставка фильтра изменена с Data1 на Data2
7	Включена подмена дискретного входа
8	Отключена подмена дискретного входа
9	Значение подмененного состояния установлено в '1'
10	Значение подмененного состояния установлено в '0'

2.6 nDO

Функции подготовки данных для передачи в модули дискретных выходов;

Набор функций nDO предназначен для:

- записи в очередь событий информации об изменении состояния входных дискретных сигналов;
- реализации функции подмены дискретных выходов;
- обработки пользовательских команд от ЧМИ;
- записи в очередь событий информации об исполнении пользовательских команд;
- подготовки данных для ЧМИ;

Для использования необходимы функции из разделов nGeneral, nPsw, nArh, nTime.

Перечень типов данных приведен в табл.2.6.1.

Таблица 2.6.1.

Наименование	Тип	Описание
tDO	STRUCT	Массив данных модуля обработки дискретных выходов
tDOValue	UNION	Обрабатываемый массив значений дискретных выходов
tDOSet	STRUCT	Уставки модуля обработки дискретных выходов

Наименование	Тип	Описание
tDOValueIn	ALIAS	Входной массив значений дискретных выходов

Перечень функций и функциональных блоков приведен в табл.2.6.2.

Таблица 2.6.2.

№	Наименование	Тип	Интерфейс
Функции работы с дискретными сигналами			
1	DOMain Подготовка выходных дискретных сигналов	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> AllowMask: ARRAY[0..15] OF WORD; // Уровни доступа команд ЧМИ <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> DOut: tDO; // Основной массив данных модуля обработки дискретных выходов DOSet: tDOSet; // Уставки модуля обработки дискретных выходов Hist: tHistory; // Массив данных модуля дискретной истории CmdHMI: tCmdHMI; // Команда ЧМИ Psw: tPsw; // Массив данных модуля разграничения доступа
2	DOIn Преобразование массива BOOL в упакованный (BitString) массив выходов	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> WordCount: UINT; // Количество слов для переноса <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> In: tDOValueIn; // Входной массив (тип BOOL) DOut: tDO; // Основной массив данных модуля обработки дискретных сигналов

Перечень конфигурационных констант приведен в таблице 2.6.3.

Таблица 2.6.3.

Наименование	Тип	Описание
cDOWordCount	UINT	Количество обрабатываемых слов (16 бит), размерности массивов
cDOWordCountReal	UINT	Количество фактически обрабатываемых слов (16 бит)
cDOBaseUnit	UINT	Базовый тип входного массива
AllowMask	ARRAY[0..15] OF WORD	Уровни доступа команд nDO

Перечень пользовательских команд приведен в табл. 2.6.4.

Номер группы команд: 31.

Смысл поля номер в группе: порядковый номер дискретного выхода.

Таблица 2.6.4.

Номер команды	Название команды	Тип поля «Данные команды»
1	Разрешить архивацию	-
2	Запретить архивацию	-
3	Включить подмену выхода	-
4	Отключить подмену выхода	-
5	Установить значение подмененного состояния в '1'	-
6	Установить значение подмененного состояния в '0'	-

Перечень регистрируемых событий приведен в табл. 2.6.5.

Номер группы событий: 31.

Смысл поля «Номер в группе»: порядковый номер дискретного выхода.

Таблица 2.6.5.

Номер события	Название события
0	Съем сигнала
1	Выдача сигнала
4	Запись в архив включена
5	Запись в архив отключена
6	Включена подмена дискретного входа
7	Отключена подмена дискретного входа
8	Значение подмененного состояния установлено в '1'
9	Значение подмененного состояния установлено в '0'

2.7 nAI

Функции для первичной обработки величин, полученных от модулей аналоговых входов.

Набор функций nAI предназначен для:

- преобразования величины, полученной от АЦП различных модулей аналоговых входов в инженерную величину;
- контроля исправности измерительного канала;
- реализации функции подмены величин аналоговых входов;
- сравнению инженерной величины с уставками предупредительной и аварийной сигнализации и формировании флагов сигнализации с учетом задержек срабатывания и гистерезиса;
- записи в очередь событий фактов изменения исправности измерительного канала и флагов выхода величины за пределы предупредительных и аварийных уставок;
- вычисление обобщенных флагов;
- обработки пользовательских команд от ЧМИ;
- записи в очередь событий информации об исполнении пользовательских команд;
- подготовки данных для ЧМИ;

Для использования необходимы функции из разделов nGeneral, nPsw, nArh, nTime.

Перечень типов данных приведен в табл.2.7.1.

Таблица 2.7.1.

Наименование	Тип	Описание
tSensors	STRUCT	Общая структура аналоговых сигналов
tAISet	STRUCT	Структура уставок модуля обработки аналоговых сигналов
tAICommonData	STRUCT	Структура единичной записи массива общих данных аналоговых сигналов
tAlarmGroup	STRUCT	Структура единичной записи привязки датчиков к группам
tAlarmValues	STRUCT	Структура единичной записи значений уставок сигнализации аналоговых сигналов
tAIStatuses	STRUCT	Структура обобщенного слова состояния аналогового сигнала
tSensStatus	STRUCT	Структура слова состояния аналогового датчика
tAlarmStatus	STRUCT	Структура слова состояния уставок аналогового сигнала
tAlarmControlWord	STRUCT	Структура единичной записи управления уставками аналогового сигнала
tAIExtValue	STRUCT	Структура единичной записи массива внешних данных для обработки
tAIExtValues	ALIAS	Массив внешних данных для обработки
tAIRackValue	STRUCT	Структура единичной записи массива данных от крейтовых модулей
tAIRackValues	ALIAS	Массив данных от крейтовых модулей
tAIAlarmTimers	STRUCT	Структура значений таймеров технологических уставок аналоговых параметров

Перечень функций и функциональных блоков приведен в табл.2.7.2.

Таблица 2.7.2.

№	Наименование	Тип	Интерфейс
Функции работы с аналоговыми сигналами			
1	AIMain Подготовка выходных дискретных сигналов	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • AllowMask: ARRAY[0..63] OF WORD; // Массив флагов разрешения выполнения команд датчиков • AllowMaskGroup: ARRAY[0..63] OF WORD; // Массив флагов разрешения выполнения команд групп

№	Наименование	Тип	Интерфейс
			<p><u>Входы-выходы (InOuts)</u></p> <ul style="list-style-type: none"> AI: tSensors; // Структура данных аналоговых сигналов AISet: tAISet; // Структура уставок аналоговых сигналов AIRackData: ARRAY[0..nAISettings.cAIRackCount - 1] OF tAIRackValue; // Входной массив величин от AI-блоков AIExtData: ARRAY[0..nAISettings.cAIExtCount - 1] OF tAIExtValue; // Данные для модуля AI, расчетные и сетевые Hist: tHistory; // Структура дискретной истории CmdHMI: tCmdHMI; // Команда ЧМИ Psw: tPsw; // Уровень доступа
2	<p>AIValidValue</p> <p>Формирование флага достоверности аналогового сигнала</p>	Ф	<p><u>Входы (Inputs)</u></p> <ul style="list-style-type: none"> Status: tAIStatuses; // Обобщенное слово состояния аналогового сигнала <p><u>Выход (Return)</u></p> <ul style="list-style-type: none"> AIValidValue: BOOL; // Флаг достоверности аналогового сигнала
3	<p>AINotValidValue</p> <p>Формирование флага недостоверности аналогового сигнала</p>	Ф	<p><u>Входы (Inputs)</u></p> <ul style="list-style-type: none"> Status: tAIStatuses; // Обобщенное слово состояния аналогового сигнала <p><u>Выход (Return)</u></p> <ul style="list-style-type: none"> AINotValidValue: BOOL; // Флаг недостоверности аналогового сигнала
4	<p>GetLowerLimitStatus</p> <p>Сравнение с нижней уставкой с учетом задержки и гистерезиса единичного аналогового сигнала</p>	ФБ	<p><u>Входы (Inputs)</u></p> <ul style="list-style-type: none"> Value: REAL; // Значение LowerLimit: REAL; // Нижняя уставка Delay: REAL; // Задержка срабатывания, сек Histeresis: REAL; // Гистерезис QualityFlag: BOOL; // Бит качества <p><u>Выходы (Outputs)</u></p> <ul style="list-style-type: none"> LimitStatus: BOOL; // Состояние уставки
5	<p>GetUpperLimitStatus</p> <p>Сравнение с верхней уставкой с учетом задержки и гистерезиса единичного аналогового сигнала</p>	ФБ	<p><u>Входы (Inputs)</u></p> <ul style="list-style-type: none"> Value: REAL; // Значение UpperLimit: REAL; // Верхняя уставка Delay: REAL; // Задержка срабатывания, сек Histeresis: REAL; // Гистерезис QualityFlag: BOOL; // Бит качества <p><u>Выходы (Outputs)</u></p> <ul style="list-style-type: none"> LimitStatus: BOOL; // Состояние уставки

№	Наименование	Тип	Интерфейс
6	AIScaling Масштабирование кода АЦП в инженерную аналоговую величину	ФБ	<p><u>Входы (Inputs)</u></p> <ul style="list-style-type: none"> • InRaw: REAL; // Значение для масштабирования • RawMin: REAL; // Минимальное входное значение кода • RawMax: REAL; // Максимальное входное значение кода • RealMin: REAL; // Минимальное входное значение инженерной величины • RealMax: REAL; // Максимальное входное значение инженерной величины • SqrtScaling: BOOL; // Квадратичная градуировка • ReverseScaling: BOOL; // Обратная градуировка <p><u>Выход (Return)</u></p> <ul style="list-style-type: none"> • AIScaling: REAL; // Инженерная аналоговая величина

Перечень конфигурационных констант приведен в таблице 2.7.3.

Таблица 2.7.3.

Наименование	Тип	Описание
cAICount	UINT	Максимальное количество аналоговых величин
cAICountReal	UINT	Фактически обрабатываемое количество аналоговых величин
cAIGroupCount	UINT	Максимальное количество групп аналоговых величин
cAIExtCount	UINT	Максимальное количество внешних аналоговых величин
cAIRackCount	UINT	Максимальное количество AI-модулей
AllowMask	ARRAY[0..63] OF WORD	Уровни доступа команд nAI
AllowMaskGroup	ARRAY[0..63] OF WORD	Уровни доступа команд групп nAI

Перечень пользовательских команд аналоговых сигналов приведен в табл. 2.7.4.

Номер группы команд: 1.

Смысл поля номер в группе: порядковый номер аналоговой величины.

Таблица 2.7.4.

Номер команды	Название команды	Тип поля «Данные команды»
1	Квитировать/сбросить неисправность	-
2	Режим подмены включить (использовать фактическое значение параметра)	-

Номер команды	Название команды	Тип поля «Данные команды»
3	Режим подмены отключить	-
4	Включить	-
5	Исключить	-
6	Изменить нижний предел измерения	REAL
7	Изменить верхний предел измерения	REAL
12	Установить значение для подмены	REAL
16	Изменить номер группы	REAL
17	Включить/отключить НАУ	-
19	Включить/отключить НПУ	-
21	Включить/отключить ВПУ	-
23	Включить/отключить ВАУ	-
25	Включить квадратичную шкалу	-
26	Отключить квадратичную шкалу	-
27	Включить обратную шкалу	-
28	Отключить обратную шкалу	-
34	Изменить номер блока	REAL
35	Изменить номер канала	REAL
36	Изменить адрес подключения	REAL
38	Изменить тип сигнала	REAL
39	Изменить тип канала	REAL
40	Изменить уставку компенсации	REAL
42	Режим подмены включить (использовать предустановленное значение параметра)	-

Перечень регистрируемых событий приведен в табл. 2.7.5.

Номер группы событий: 1.

Смысл поля «Номер в группе»: порядковый номер аналоговой величины.

Таблица 2.7.5.

Номер события	Название события
0	Неисправность сброшена
1	Неисправность квитирована
2	Возникновение неисправности
3	Включен режим подмены (Data1)
4	Отключен режим подмены
5	Датчик включен в обработку
6	Датчик исключен из обработки
7	Нижний предел измерения изменен с Data1 на Data2
8	Верхний предел измерения изменен с Data1 на Data2
13	Значение подмены изменено с Data1 на Data2
16	Уход неисправности
17	Обратная шкала включена
18	Обратная шкала отключена
20	Срабатывание НАУ (на Data1)

Номер события	Название события
21	Уход НАУ (на Data1)
22	Срабатывание НПУ (на Data1)
23	Уход НПУ (на Data1)
24	Срабатывание ВПУ (на Data1)
25	Уход ВПУ (на Data1)
26	Срабатывание ВАУ (на Data1)
27	Уход ВАУ (на Data1)
28	Группа датчика изменена с Data1 на Data2
30	Обработка НАУ включена
31	Обработка НАУ отключена
32	Обработка НПУ включена
33	Обработка НПУ отключена
34	Обработка ВПУ включена
35	Обработка ВПУ отключена
36	Обработка ВАУ включена
37	Обработка ВАУ отключена
38	Квадратичная шкала включена
39	Квадратичная шкала отключена
40	НАУ квитирована
41	НПУ квитирована
42	ВПУ квитирована
43	ВАУ квитирована
44	НАУ сброшена
45	НПУ сброшена
46	ВПУ сброшена
47	ВАУ сброшена
51	Номер измерительного блока изменен с Data1 на Data2
52	Номер измерительного канала изменен с Data1 на Data2
53	Адрес подключения изменен с Data1 на Data2
55	Код типа сигнала изменен с Data1 на Data2
56	Код типа измерительного канала изменен с Data1 на Data2
57	Уставка компенсации изменена с Data1 на Data2

Перечень пользовательских команд аналоговых сигналов приведен в табл. 2.7.6.

Номер группы команд: 11.

Смысл поля номер в группе: порядковый номер группы аналоговых величин.

Таблица 2.7.6.

Номер команды	Название команды	Тип поля «Данные команды»
1	Изменить НАУ	REAL
2	Изменить НПУ	REAL
3	Изменить ВПУ	REAL
4	Изменить ВАУ	REAL
6	Изменить задержку срабатывания НАУ	REAL

Номер команды	Название команды	Тип поля «Данные команды»
7	Изменить задержку срабатывания НПУ	REAL
8	Изменить задержку срабатывания ВПУ	REAL
9	Изменить задержку срабатывания ВАУ	REAL
10	Изменить нижний предел измерения	REAL
11	Изменить верхний предел измерения	REAL
13	Изменить уставку гистерезиса	REAL
17	Включить/отключить НАУ	-
19	Включить/отключить НПУ	-
21	Включить/отключить ВПУ	-
23	Включить/отключить ВАУ	-
26	Изменить задержку срабатывания НАУ (для NS)	DINT
27	Изменить задержку срабатывания НПУ (для NS)	DINT
28	Изменить задержку срабатывания ВПУ (для NS)	DINT
29	Изменить задержку срабатывания ВАУ (для NS)	DINT

Перечень регистрируемых событий приведен в табл. 2.7.7.

Номер группы событий: 11.

Смысл поля «Номер в группе»: порядковый номер группы аналоговых величин.

Таблица 2.7.7.

Номер события	Название события
1	НАУ изменена с Data1 на Data2
2	НПУ изменена с Data1 на Data2
3	ВПУ изменена с Data1 на Data2
4	ВАУ изменена с Data1 на Data2
6	Задержка срабатывания НАУ изменена с Data1 на Data2
7	Задержка срабатывания НПУ изменена с Data1 на Data2
8	Задержка срабатывания ВПУ изменена с Data1 на Data2
9	Задержка срабатывания ВАУ изменена с Data1 на Data2
13	Гистерезис изменен с Data1 на Data2
30	Обработка НАУ включена
31	Обработка НАУ отключена
32	Обработка НПУ включена
33	Обработка НПУ отключена
34	Обработка ВПУ включена
35	Обработка ВПУ отключена
36	Обработка ВАУ включена
37	Обработка ВАУ отключена

2.8 пАО

Функции подготовки данных для передачи в модули аналоговых выходов.

Зависимостей от функций других разделов ПО нет.

Перечень функций приведен в табл.2.8.1.

Таблица 2.8.1.

№	Наименование	Тип	Интерфейс
Функции работы с аналоговыми сигналами			
1	AOScaling Масштабирование инженерной аналоговой величины в код ЦАП.	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • InReal: REAL; // Значение инженерной величины • RawMin: REAL; // Минимальное входное значение кода • RawMax: REAL; // Максимальное входное значение кода • RealMin: REAL; // Минимальное входное значение инженерной величины • RealMax: REAL; // Максимальное входное значение инженерной величины <u>Выход (Return)</u> <ul style="list-style-type: none"> • AOScaling: REAL; // Код ЦАП <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> • Error: BOOL; // Ошибка масштабирования

2.9 nEmg

Функции для формирования признаков индивидуальной и обобщенной сигнализации.

Набор функций nEmg предназначен для:

- формирования массива слов состояния сигнализации с учетом состояния входного сигнала, типа события, команд оператора;
- формирования обобщенных слов состояния сигнализации по группам сигналов;
- формирования стека активной сигнализации для представления на ЧМИ;
- записи в очередь событий фактов изменения состояния сигнализации;
- обработки пользовательских команд от ЧМИ;
- записи в очередь событий информации об исполнении пользовательских команд;
- подготовки данных для ЧМИ;

Для использования необходимы функции из разделов nGeneral, nPsw, nArh, nTime.

Перечень типов данных приведен в табл.2.9.1.

Таблица 2.9.1.

Наименование	Тип	Описание
tEmg	STRUCT	Общая структура обработки сигнализации
tEmgStatus	STRUCT	Слово состояния сигнализации
tEmgSet	STRUCT	Структура уставок сигнализации
tEmgRSP	STRUCT	Структура оперативных данных для ЧМИ
tEmgRSPStack	STRUCT	Структура стека активной сигнализации для ЧМИ
tEmgRSPStackLine	STRUCT	Структура единичной записи в стеке активной сигнализации для ЧМИ

Перечень функций и функциональных блоков приведен в табл.2.9.2.

Таблица 2.9.2.

№	Наименование	Тип	Интерфейс
Функции работы с сигнализацией			
1	EmgMain Обработка сигнализации	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> AllowMask: ARRAY[0..15] OF WORD; // Уровни доступа команд ЧМИ <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> Emg: tEmg; // Структура событий EmgSet: tEmgSet; // Структура уставок событий EmgStack: tEmgRSPStack; // Структура стека событий Hist: tHistory; // Структура дискретной истории CmdHMI: tCmdHMI; // Структура команд ЧМИ Psw: tPsw; // Уровень доступа
2	EmgToRSP Подготовка массива данных для визуализации на ЧМИ	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> ForceUpdate: BOOL; // Принудительно обновить данные в выходном массиве <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> Emg: tEmg; // Структура событий EmgSet: tEmgSet; // Структура уставок событий EmgRSP: tEmgRSP; // Структура визуализации на RSP
Функции работы со стеком сигнализации			
3	EmgStackInit Первичная инициализация стека сигнализации	Ф	<u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> Emg: tEmg; // Структура данных предупреждений EmgSet: tEmgSet; // Структура уставок событий EmgStack: tEmgRSPStack; // Структура стека RSP
4	EmgStackWrite Запись в стек нового события	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> EmgID: UINT; // ID события <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> EmgStack: tEmgRSPStack; // Структура стека RSP
5	EmgStackClear Очистка стека от сброшенных событий	Ф	<u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> Emg: tEmg; // Структура данных предупреждений EmgSet: tEmgSet; // Структура уставок событий EmgStack: tEmgRSPStack; // Структура стека RSP
Функции управления устройствами сигнализации			
6	EmgBlinkLamp Программный блинкер	ФБ	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> In: BOOL; // Вход блинкера Reset: BOOL; // Сброс сигнализации <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> Lamp: BOOL; // Лампа блинкера

№	Наименование	Тип	Интерфейс
7	EmgSWToLamp Формирование лампы на базе слова состояния события	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> SW: tEmgStatus; // Слово состояния события <u>Выход (Return)</u> <ul style="list-style-type: none"> EmgSWToLamp: BOOL; // Выход на лампу
8	EmgSWGrouptoLamp Формирование лампы на базе слова состояния группы событий	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> GSW: WORD; // Слово состояния по группы событий <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> AlarmLamp: BOOL; // Выход на лампу аварийной сигнализации WarningLamp: BOOL; // Выход на лампа предупредительной сигнализации <u>Выход (Return)</u> <ul style="list-style-type: none"> EmgSWGrouptoLamp: BOOL; // Выход на обобщенную лампу

Перечень конфигурационных констант приведен в таблице 2.9.3.

Таблица 2.9.3.

Наименование	Тип	Описание
cEmgCount	UINT	Максимальное количество обрабатываемых событий
cEmgCountReal	UINT	Фактическое количество обрабатываемых событий
cEmgStackLineCount	UINT	Максимальное количество событий в стеке
cEmgStackLineCountReal	UINT	Фактическое обрабатываемое количество событий в стеке
cEmgGroupCount	UINT	Максимальное количество обрабатываемых групп событий
AllowMask	ARRAY[0..15] OF WORD	Уровни доступа команд nEmg

Перечень пользовательских команд аналоговых сигналов приведен в табл. 2.9.4.

Номера групп команд: 32 – событие, 33 – предупреждение 34 -- авария.

Смысл поля «Номер в группе»: порядковый номер сигнала.

Таблица 2.9.4.

Номер команды	Название команды	Тип поля «Данные команды»
1	Квитировать/сбросить сигнализацию	-
2	Изменить тип события	REAL
3	Изменить группу события	REAL

Перечень регистрируемых событий приведен в табл. 2.9.5.

Номера групп событий: 32 – событие, 33 – предупреждение 34 -- авария.

Смысл поля «Номер в группе»: порядковый номер сигнала.

Таблица 2.9.5.

Номер события	Название события
0	Уход события
1	Возникновение события
2	Событие квитировано
10	Тип события изменен с Data1 на Data2
11	Группа сигнализации изменена с Data1 на Data2

2.10 nProtect

Функции для формирования сигналов технологических защит.

Набор функций nProtect предназначен для:

- реализации технологических защит, с учетом задержек срабатывания, ввода накладок, признаков включения в работу;
- тестирования технологических защит;
- записи в очередь событий фактов изменения состояния защит;
- обработки пользовательских команд от ЧМИ;
- записи в очередь событий информации об исполнении пользовательских команд;
- подготовки данных для ЧМИ;
- реализации алгоритмов 2002 и 2003.

Для использования необходимы функции из разделов nGeneral, nPsw, nArh, nTime.

Перечень типов данных приведен в табл.2.10.1.

Таблица 2.10.1.

Наименование	Тип	Описание
tProtect	STRUCT	Структура единичной записи состояния технологической защиты
tProtects	ALIAS	Массив структур состояния технологических защит
tProtectSet	STRUCT	Структура единичной записи уставок технологической защиты
tProtectSets	ALIAS	Массив структур уставок технологической защиты
tProtectHMI1	STRUCT	Структура единичной записи оперативных данных для ЧМИ
tProtectsHMI1	ALIAS	Массив структур оперативных данных для ЧМИ
tProtectHMI2	STRUCT	Структура единичной записи времен срабатывания и теста для ЧМИ
tProtectsHMI2	ALIAS	Массив структур времен срабатывания и теста для ЧМИ

Перечень функций и функциональных блоков приведен в табл.2.10.2.

Таблица 2.10.2.

№	Наименование	Тип	Интерфейс
Функции работы с технологическими защитами			

№	Наименование	Тип	Интерфейс
1	Protect Формирование сигналов технологических защит	Ф	<p><u>Входы (Inputs)</u></p> <ul style="list-style-type: none"> • Enabled: BOOL; // Защита включена • ID: UINT; // Номер защиты • In: BOOL; // Вход условия срабатывание защиты • TechCondition: BOOL; // Технологическое разрешение работы защиты • Deblock: BOOL; // Деблокировка • DeblockAllowed: BOOL; // Деблокировка разрешена <p><u>Входы-выходы (InOuts)</u></p> <ul style="list-style-type: none"> • ProtectData: tProtects; // Основной массив данных модуля обработки защит • ProtectSet: tProtectSets; // Уставки модуля обработки защит • Hist: tHistory; // Данные модуля истории событий <p><u>Выходы (Outputs)</u></p> <ul style="list-style-type: none"> • AlarmOut: BOOL; // Срабатывание защиты на сигнал <p><u>Выход (Return)</u></p> <p>Protect: BOOL; // Срабатывание защиты на останов</p>
2	ProtectToRSP Подготовка массива данных для визуализации на ЧМИ	Ф	<p><u>Входы (Inputs)</u></p> <ul style="list-style-type: none"> • TimeShift: REAL; // Часовой пояс для вычисления локального времени для отображения <p><u>Входы-выходы (InOuts)</u></p> <ul style="list-style-type: none"> • ProtData: tProtects; // Основной массив данных модуля обработки защит • ProtSet: tProtectSets; // Уставки модуля обработки защит • HMI1: tProtectsHMI1; // Массив оперативных данных защит для ЧМИ • HMI2: tProtectsHMI2; // Массив времен срабатывания защит для ЧМИ
Алгоритмы 2002 и 2003 для аналоговых сигналов с учетом достоверности			

№	Наименование	Тип	Интерфейс
3	a2of2great Алгоритм 2oo2 по повышению величин аналоговых сигналов с учетом достоверности	Ф	<p><u>Входы (Inputs)</u></p> <ul style="list-style-type: none"> • SensValue1: REAL; // Значение величины №1 • SensValue2: REAL; // Значение величины №2 • SetValue: REAL; // Уставка защиты • SensFault1: BOOL; // Неисправность датчика №1 • SensFault2: BOOL; // Неисправность датчика №2 • CheckValue: BOOL; // Учет допустимых отклонений • Tolerance: REAL; // Допустимое отклонение <p><u>Выход (Return)</u></p> <ul style="list-style-type: none"> • a2of2great: BOOL; // Срабатывание
4	a2of2less Алгоритм 2oo2 по повышению величин аналоговых сигналов с учетом достоверности	Ф	<p><u>Входы (Inputs)</u></p> <ul style="list-style-type: none"> • SensValue1: REAL; // Значение величины №1 • SensValue2: REAL; // Значение величины №2 • SetValue: REAL; // Уставка защиты • SensFault1: BOOL; // Неисправность датчика №1 • SensFault2: BOOL; // Неисправность датчика №2 • CheckValue: BOOL; // Учет допустимых отклонений • Tolerance: REAL; // Допустимое отклонение <p><u>Выход (Return)</u></p> <ul style="list-style-type: none"> • a2of2less: BOOL; // Срабатывание
5	A2of3great Алгоритм 2oo3 по повышению величин аналоговых сигналов с учетом достоверности	ФБ	<p><u>Входы (Inputs)</u></p> <ul style="list-style-type: none"> • SensValue1: REAL; // Значение величины №1 • SensValue2: REAL; // Значение величины №2 • SensValue3: REAL; // Значение величины №3 • SetValue: REAL; // Уставка защиты • SensFault1: BOOL; // Неисправность датчика №1 • SensFault2: BOOL; // Неисправность датчика №2 • SensFault3: BOOL; // Неисправность датчика №3 • CheckValue: BOOL; // Учет допустимых отклонений • Tolerance: REAL; // Допустимое отклонение • ToleranceTimer: TIME; // Задержка выхода за допустимое отклонение

№	Наименование	Тип	Интерфейс
			Выходы (Outputs) <ul style="list-style-type: none"> Out2of3: BOOL; // Сработала 2of3 Out1of3: BOOL; // Сработала 1of3
6	A2of3less Алгоритм 2oo3 по понижению величин аналоговых сигналов с учетом достоверности	ФБ	Входы (Inputs) <ul style="list-style-type: none"> SensValue1: REAL; // Значение величины №1 SensValue2: REAL; // Значение величины №2 SensValue3: REAL; // Значение величины №3 SetValue: REAL; // Уставка защиты SensFault1: BOOL; // Неисправность датчика №1 SensFault2: BOOL; // Неисправность датчика №2 SensFault3: BOOL; // Неисправность датчика №3 CheckValue: BOOL; // Учет допустимых отклонений Tolerance: REAL; // Допустимое отклонение ToleranceTimer: TIME; // Задержка выхода за допустимое отклонение
			Выходы (Outputs) <ul style="list-style-type: none"> Out2of3: BOOL; // Сработала 2of3 Out1of3: BOOL; // Сработала 1of3

Перечень конфигурационных констант приведен в таблице 2.10.3.

Таблица 2.10.3.

Наименование	Тип	Описание
cSetCount	UINT	Максимальное количество обрабатываемых защит
AllowMask	ARRAY[0..31] OF WORD	Уровни доступа команд nProtect

Перечень пользовательских команд аналоговых сигналов приведен в табл. 2.10.4.

Номер группы команд: 8.

Смысл поля номер в группе: порядковый номер защиты.

Таблица 2.10.4.

Номер команды	Название команды	Тип поля «Данные команды»
1	Включить ручной ввод защиты	-
2	Отключить ручной ввод защиты	-
3	Включить ремонтный вывод защиты (действие на сигнал)	-
4	Отключить ремонтный вывод защиты (действие на сигнал)	-
5	Включить режим тестирования защиты	-
6	Отключить режим тестирования защиты	-

Номер команды	Название команды	Тип поля «Данные команды»
9	Деблокировать	-
11	Изменить задержку срабатывания	REAL
20	Отключить защиту	-
21	Включить защиту	-

Перечень регистрируемых событий приведен в табл. 2.10.5.

Номер группы событий: 8.

Смысл поля «Номер в группе»: порядковый номер защиты.

Таблица 2.10.5.

Номер события	Название события
1	Включен ручной ввод защиты
2	Отключен ручной ввод защиты
3	Включен ремонтный вывод защиты (действие на сигнал)
4	Отключен ремонтный вывод защиты (действие на сигнал)
5	Включен режим тестирования защиты
6	Отключен режим тестирования защиты
9	Защита деблокирована
11	Изменена задержка срабатывания с Data1 на Data2
12	Защита введена в работу
13	Защита выведена из работы
14	Начало отсчета времени задержки срабатывания защиты
15	Срабатывание защиты на сигнал
16	Срабатывание защиты на исполнение
20	Защита отключена
21	Защита включена

2.11 nDSP

Набор функций nDSP предназначен для реализации цифровых фильтров сигналов.

Зависимостей от функций других разделов ПО нет.

Перечень функциональных блоков приведен в табл.2.11.1.

Таблица 2.11.1.

№	Наименование	Тип	Интерфейс
Цифровые фильтры			
1	FirstGradeFilter Звено запаздывания первого порядка	ФБ	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • X0: REAL; // Входная величина • X0Error: BOOL; // Флаг ошибки • K: REAL; // Коэффициент усиления • T0: REAL; // Постоянная времени (сек) <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> • Y0: REAL; // Выходная величина

№	Наименование	Тип	Интерфейс
2	MovingAverageFilter Фильтр со скользящим средним	ФБ	<u>Входы (Inputs)</u> • X0: REAL; // Входная величина
			<u>Выходы (Outputs)</u> • Y0: REAL; // Выходная величина
3	NoiseFilter Сглаживание сигнала, звено первого порядка в сочетании с фильтрацией выбросов (иголок)	ФБ	<u>Входы (Inputs)</u> • X0: REAL; // Входная величина • X0Error: BOOL; // Флаг ошибки • X0ErrorTime: REAL; // Время сохранения выхода при неисправности сигнала • T0: REAL; // Постоянная времени фильтра первого порядка • dX: REAL; // Допустимый уровень изменения сигнала • TdX: REAL; // Время пропуска скачка изменения • YErrValue: REAL; // Значение выходной величины при неисправности
			<u>Выходы (Outputs)</u> • Y0: REAL; // Выходная величина

2.12 nPolinom

Интерполяционные функции.

Зависимостей от функций других разделов ПО нет.

Перечень типов данных приведен в табл.2.12.1.

Таблица 2.12.1.

Наименование	Тип	Описание
tCoord	STRUCT	Структура узла полинома
tInterData	STRUCT	Структура последовательность узлов полинома

Перечень функций и функциональных блоков приведен в табл.2.10.2.

Таблица 2.12.2.

№	Наименование	Тип	Интерфейс
Интерполяционные функции			
1	LinearIntrepolation Линейная интерполяция	Ф	<u>Входы (Inputs)</u> • In: REAL; // Вход • Inter: tInterData; // Последовательность узлов полинома
			<u>Выходы (Outputs)</u> • OutOfBounds: BOOL; // Выход за пределы интерполяции
			<u>Выход (Return)</u> LinearIntrepolation: REAL; // Результат интерполяции

№	Наименование	Тип	Интерфейс
2	QuadraticInterpolation Кусочно-квадратичная интерполяция	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> In: REAL; // Вход Inter: tInterData; // Последовательность узлов полинома <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> OutOfBounds: BOOL; // Выход за пределы интерполяции errData: BOOL; // Неправильные входные данные <u>Выход (Return)</u> QuadraticInterpolation: REAL; // Результат интерполяции
3	SplineInterpolation Интерполяция сплайном 3-го порядка	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> In: REAL; // Вход Inter: tInterData; // Последовательность узлов полинома <u>Выходы (Outputs)</u> <ul style="list-style-type: none"> OutOfBounds: BOOL; // Выход за пределы интерполяции errData: BOOL; // Неправильные входные данные <u>Выход (Return)</u> SplineInterpolation: REAL; // Результат интерполяции

2.13 nSettings

Функции для ввода аналоговых и дискретных уставок.

Набор функций предназначен:

- для ввода аналоговых уставок в формате REAL;
- для проверки допустимости диапазона введенных аналоговых уставок;
- для проверки правильности представления введенной аналоговой уставки (NaN, +Inf, -Inf);
- для автоматического вычисления величины аналоговой уставки в относительных единицах;
- для ввода дискретных уставок;
- записи в очередь событий информации об исполнении пользовательских команд;
- подготовки данных для ЧМИ;

Для использования необходимы функции из разделов nGeneral, nPsw, nArg, nTime.

Перечень типов данных приведен в табл.2.13.1.

Таблица 2.13.1.

Наименование	Тип	Описание
tSetValue	STRUCT	Массив аналоговых уставок общего назначения
tSetAdvValue	STRUCT	Массив уставок общего назначения с масштабированием

Наименование	Тип	Описание
tSetAdvSet	STRUCT	Структура единичной записи параметров уставок с масштабированием
tSetBoolValue	STRUCT	Массив дискретных уставок общего назначения
tSetPackBoolValue	STRUCT	Упакованный в слова массив дискретных уставок (для ЧМИ)

Перечень функций и функциональных блоков приведен в табл.2.13.2.

Таблица 2.13.2.

№	Наименование	Тип	Интерфейс
Функции работы с аналоговыми уставками общего назначения			
1	SettingsCmd Ввод аналоговых уставок общего назначения	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • GroupID: UINT; // Номер группы • Offset: UINT; // Смещение ID команды • Count: UINT; // Количество обрабатываемых уставок • AllowMask: ARRAY[0..15] OF WORD; // Уровень доступа <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> • Settings: tSetValue; // Значения уставок • Hist: tHistory; // Массив данных модуля дискретной истории • CmdHMI: tCmdHMI; // Команда ЧМИ • Psw: tPsw; // Массив данных модуля разграничения доступа
2	SettingsAdvCmd Ввод аналоговых уставок общего назначения с проверкой достоверности и вычислением относительной величины	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • GroupID: UINT; // Номер группы • Offset: UINT; // Смещение ID команды • Count: UINT; // Количество обрабатываемых уставок • AllowMask: ARRAY[0..15] OF WORD; // Уровень доступа <u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> • Settings: tSetAdvValue; // Значения уставок • Hist: tHistory; // Массив данных модуля дискретной истории • CmdHMI: tCmdHMI; // Команда ЧМИ • Psw: tPsw; // Массив данных модуля разграничения доступа
Функции работы с дискретными уставками общего назначения			

№	Наименование	Тип	Интерфейс
3	SettingsBoolCmd Ввод дискретных уставок общего назначения	ФБ	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • GroupID: UINT; // Номер группы • Offset: UINT; // Смещение ID команды • Count: UINT; // Количество обрабатываемых уставок • AllowMask: ARRAY[0..15] OF WORD; // Уровень доступа
			<u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> • Settings: tSetBoolValue; // Значения уставок • Hist: tHistory; // Массив данных модуля дискретной истории • CmdHMI: tCmdHMI; // Команда ЧМИ • Psw: tPsw; // Массив данных модуля разграничения доступа
4	SettingsBoolToRSP Упаковка уставок в слова для передачи на ВУ	Ф	<u>Входы (Inputs)</u> <ul style="list-style-type: none"> • Count: UINT; // Количество битов для переноса
			<u>Входы-выходы (InOuts)</u> <ul style="list-style-type: none"> • Settings: tSetBoolValue; // Значения уставок • PackSettings: tSetPackBoolValue; // Упакованные значения уставок

Перечень конфигурационных констант приведен в таблице 2.13.3.

Таблица 2.13.3.

Наименование	Тип	Описание
cSetCount	UINT	Максимальное количество обрабатываемых аналоговых уставок
cSetAdvCount	UINT	Максимальное количество обрабатываемых аналоговых уставок с масштабированием
cSetBoolCount	UINT	Максимальное количество обрабатываемых дискретных уставок
AllowMask	ARRAY[0..15] OF WORD	Уровни доступа команд ЧМИ по аналоговым уставкам
AllowMaskAdv	ARRAY[0..15] OF WORD	Уровни доступа команд ЧМИ по аналоговым уставкам с масштабированием
AllowMaskBool	ARRAY[0..15] OF WORD	Уровни доступа команд ЧМИ по дискретным уставкам

Перечень пользовательских команд аналоговых уставок приведен в табл. 2.13.4.

Номер группы команд: задается из интерфейса функции, по умолчанию 40.

Смысл поля номер в группе: порядковый номер аналоговой уставки.

Таблица 2.13.4.

Номер команды	Название команды	Тип поля «Данные команды»
1	Изменить уставку	REAL

Перечень регистрируемых событий аналоговых уставок приведен в табл. 2.13.5.
Номер группы событий: задается из интерфейса функции, по умолчанию 40.
Смысл поля «Номер в группе»: порядковый номер аналоговой уставки.

Таблица 2.13.5.

Номер события	Название события
1	Уставка изменена с Data1 на Data2

Перечень пользовательских команд аналоговых уставок с масштабированием приведен в табл. 2.13.6.

Номер группы команд: задается из интерфейса функции, по умолчанию 40.
Смысл поля номер в группе: порядковый номер аналоговой уставки.

Таблица 2.13.6.

Номер команды	Название команды	Тип поля «Данные команды»
1	Изменить уставку	REAL
2	Изменить нижний допустимый предел	REAL
3	Изменить верхний допустимый предел	REAL
4	Изменить коэффициент масштабирования	REAL
5	Изменить смещение	REAL

Перечень регистрируемых событий аналоговых уставок с масштабированием приведен в табл. 2.13.7.

Номер группы событий: задается из интерфейса функции, по умолчанию 40.
Смысл поля «Номер в группе»: порядковый номер аналоговой уставки.

Таблица 2.13.7.

Номер события	Название события
1	Уставка изменена с Data1 на Data2
2	Нижний допустимый предел изменен с Data1 на Data2
3	Верхний допустимый предел изменен с Data1 на Data2
4	Коэффициент масштабирования изменен с Data1 на Data2
5	Уставка смещения изменена с Data1 на Data2

Перечень пользовательских команд дискретных уставок приведен в табл. 2.13.8.

Номер группы команд: задается из интерфейса функции, по умолчанию 41.
Смысл поля номер в группе: порядковый номер дискретной уставки.

Таблица 2.13.8.

Номер команды	Название команды	Тип поля «Данные команды»
1	Сбросить уставку	-
2	Установить уставку	-
3	Установить уставку на один цикл	-
4	Инvertировать уставку	-

Перечень регистрируемых событий дискретных уставок приведен в табл. 2.13.9.
Номер группы событий: задается из интерфейса функции, по умолчанию 41.
Смысл поля «Номер в группе»: порядковый номер дискретной уставки.

Таблица 2.13.8.

Номер события	Название события
1	Уставка установлена
2	Уставка сброшена

2.14 nValve

Универсальный функциональный блок управления электромеханическим устройством (задвижка, клапан, насос и т.п).

Предназначен для:

- формирования управляющих сигналов электромеханическим устройством в соответствии с полученными сигналами состояния устройства и пользовательских, автоматических и аварийных команд;
- определения ряда неисправностей электромеханического устройства;
- контроля времени движения (включения) устройства;
- обработки пользовательских команд от ЧМИ;
- записи в очередь событий информации об исполнении пользовательских команд;
- подготовки данных для ЧМИ;

Для использования необходимы функции из разделов nGeneral, nPsw, nArh, nTime.

Перечень типов данных приведен в табл.2.14.1.

Таблица 2.14.1.

Наименование	Тип	Описание
tValve	STRUCT	Структура данных по механизмам
tValveStored	STRUCT	Структура единичной записи данных по механизмам
tValveStatus	STRUCT	Структура слова состояния механизма
tValveStoredSignals	STRUCT	Структура слова состояния механизма
tValveTimeSet	STRUCT	Структура уставок контроля времени движения
tValveCounters	STRUCT	Структура счетчиков контроля времени движения

Перечень функциональных блоков приведен в табл.2.14.2.

Таблица 2.14.2.

№	Наименование	Тип	Интерфейс
Управление электромеханическими устройствами			
1	FBValve Ввод аналоговых уставок общего назначения	ФБ	Входы (Inputs) <ul style="list-style-type: none"> • ValveID: UINT; // Номер обрабатываемой задвижки • NotFound: BOOL; // Механизм отсутствует • TypePump: BOOL; // Механизм типа насос • TypeFlap: BOOL; // Механизм типа

№	Наименование	Тип	Интерфейс
			эл.магн.клапан <ul style="list-style-type: none"> • QF: BOOL; // Силовой авт. выключатель включен • PowerOn: BOOL; // Питание подано, концевик безопасности не сработал • Faulted: BOOL; // Сигнал "Авария" из схемы управления • SQ_Opened: BOOL; // Концевик "Открыто/включен" • SQ_Closed: BOOL; // Концевик "Закрыто/отключен" • CmdAutoOpen: BOOL; // Технологическая команда "Открыть/включить" в автоматическом режиме • CmdAutoClose: BOOL; // Технологическая команда "Закрыть/отключить" в автоматическом режиме • CmdProtectOpen: BOOL; // Команда "Открыть/включить по защите" • CmdProtectClose: BOOL; // Команда "Закрыть/отключить по защите" • CmdAutoModeOn: BOOL; // Команда "Включить автоматический режим" • CmdRemoteModeOn: BOOL; // Команда "Включить дистанционный режим" • CmdLocalModeOn: BOOL; // Команда "Включить местный режим" • CmdOpenBlockOn: BOOL; // Команда "Блокировка открытия/включения" • CmdCloseBlockOn: BOOL; // Команда "Блокировка закрытия/отключения" • KeyManual: BOOL; // Ключ "Ручное управление с ШУ" • KeyOpen: BOOL; // Кнопка местного поста "Открыть/включить" • KeyClose: BOOL; // Кнопка местного поста "Закрыть/отключить" • KeyStop: BOOL; // Кнопка местного поста "Остановить" • KeyPAO: BOOL; // Ключ в ПАО • IReleFound: BOOL; // Наличие в схеме входа токового реле • IReleOn: BOOL; // Срабатывание токового реле • MoMuffFound: BOOL; // Наличие в схеме

№	Наименование	Тип	Интерфейс
			<p>сигнала уплотнения</p> <ul style="list-style-type: none"> MoMuffOn: BOOL; // Концевик "Закрыто с уплотнением" Reset: BOOL; // Подключаемый вход RESET <p><u>Входы-выходы (InOuts)</u></p> <ul style="list-style-type: none"> Valves: tValve; // Hist: tHistory; // Массив данных модуля дискретной истории CmdHMI: tCmdHMI; // Команда ЧМИ Psw: tPsw; // Массив данных модуля разграничения доступа

Перечень конфигурационных констант приведен в таблице 2.14.3.

Таблица 2.14.3.

Наименование	Тип	Описание
cValveCmdGroupID	UINT	Номер группы
cValveCount	UINT	Максимальное количество обрабатываемых механизмов

Перечень пользовательских команд аналоговых уставок приведен в табл. 2.14.4.
Номер группы команд: задается конфигурационной константой, по умолчанию 2.
Смысл поля «Номер в группе»: порядковый номер механизма.

Таблица 2.14.4.

Номер команды	Название команды	Тип поля «Данные команды»
1	Сбросить неисправность	-
2	Включить автоматический режим	-
3	Включить дистанционный режим	-
4	Включить местный режим	-
5	Включить режим "Опробование"	-
6	Отключить режим "Опробование"	-
7	Включить режим "Не включать – работают люди!"	-
8	Отключить режим "Не включать – работают люди!"	-
13	Открыть/включить	-
14	Закрыть/отключить	-
15	Остановить	-
21	Изменить время открытия/включения	REAL
22	Изменить время закрытия/отключения	REAL
23	Изменить время схода с концевика "открыто"	REAL
24	Изменить время схода с концевика "закрыто"	REAL
25	Изменить время дожатия по моментной муфте	REAL
30	Деблокировать (смена состояния)	-

Номер команды	Название команды	Тип поля «Данные команды»
31	Подмена состояния (смена значения)	-
32	Открыто/включен по подмене (смена значения)	-
33	Закрыто/отключен по подмене (смена значения)	-
71	Изменить время открытия/включения (для NS)	DINT
72	Изменить время закрытия/отключения (для NS)	DINT
73	Изменить время схода с концевика "открыто" (для NS)	DINT
74	Изменить время схода с концевика "закрыто" (для NS)	DINT
75	Изменить время дожатия по моментной муфте (для NS)	DINT

Перечень регистрируемых событий аналоговых уставок приведен в табл. 2.14.5.
 Номер группы событий: задается конфигурационной константой, по умолчанию 2.
 Смысл поля «Номер в группе»: порядковый номер механизма.

Таблица 2.14.5.

Номер события	Название события
1	Квитирование неисправности
2	Неисправность
3	Включен автоматический режим
4	Включен дистанционный режим
5	Включен местный режим
8	Включен режим <Опробование>
9	Отключен режим <Опробование>
10	Включен режим <Не включать - работают люди!>
11	Отключен режим <Не включать - работают люди!>
12	Активация блокировки открытия/включения
13	Уход блокировки открытия/включения
14	Активация блокировки закрытия/отключения
15	Уход блокировки закрытия/отключения
16	Команда 'открыть/включить' в автомате
17	Снята команда 'открыть/включить' в автомате
18	Команда 'закрыть/отключить' в автомате
19	Снята команда 'закрыть/отключить' в автомате
20	Положение 'открыто/включен'
21	Уход из положения 'открыто/включен'
22	Положение 'закрыто/отключен'
23	Уход из положения 'закрыто/отключен'
24	Выходной сигнал 'открыть/включить'
25	Снятие выходного сигнала 'открыть/включить'
26	Выходной сигнал 'закрыть/отключить'
27	Снятие выходного сигнала 'закрыть/отключить'
28	Уход неисправности

Номер события	Название события
29	Срабатывание уплотнения по моментной муфте
30	Деблокировка механизма
31	Ввод блокировок механизма
32	Установка режима 'Подмена'
33	Снятие режима 'Подмена'
34	Открыто/включен по подмене
35	Неоткрыто/невключен по подмене
36	Закрыто/отключен по подмене
37	Незакрыто/неотключен по подмене
38	Включен ручной режим
39	Команда 'Открыть/включить по защите'
40	Уход команды 'Открыть/включить по защите'
41	Команда 'Закрыть/отключить по защите'
42	Уход команды 'Закрыть/отключить по защите'
50	Команда оператора 'Сбросить неисправность'
51	Команда оператора 'Включить автоматический режим'
52	Команда оператора 'Включить дистанционный режим'
53	Команда оператора 'Включить местный режим'
54	Команда оператора 'Включить режим <Тестирование>
55	Команда оператора 'Отключить режим <Тестирование>'
56	Команда оператора 'Включить режим <Не включать – работают люди!>'
57	Команда оператора 'Отключить режим <Не включать – работают люди!>'
63	Команда оператора 'Открыть/включить'
64	Команда оператора 'Закрыть/отключить'
65	Команда оператора 'Остановить'
71	Команда оператора 'Изменить время открытия/включения'
72	Команда оператора 'Изменить время закрытия/отключения'
73	Команда оператора 'Изменить время схода с концевика <открыто>'
74	Команда оператора 'Изменить время схода с концевика <закрыто>'
75	Команда оператора 'Изменить время дожатия по моментной муфте'
80	Команда оператора 'Деблокировать (смена состояния)'
81	Команда оператора 'Подмена состояния (смена значения)'
82	Команда оператора 'Открыто/включен по подмене (смена значения)'
83	Команда оператора 'Закрыто/отключен по подмене (смена значения)'

2.15 nCommon

Определение типов данных и функций для реализации приема обобщенных команд ПТК «Апогей-М»

Предназначен для:

- обработки пользовательских команд от ЧМИ;
- записи в очередь событий информации об исполнении пользовательских команд;

- подготовки флагов обобщенных команд для использования различными функциями ПО;

Для использования необходимы функции из разделов nGeneral, nPsw, nArh, nTime.

Перечень типов данных приведен в табл.2.15.1.

Таблица 2.15.1.

Наименование	Тип	Описание
tCommonCmd	STRUCT	Структура флагов обобщенных команд

Перечень функциональных блоков приведен в табл.2.15.2.

Таблица 2.15.2.

№	Наименование	Тип	Интерфейс
Обобщенные команды ПТК			
1	CommonCmd Обработка обобщенных команд ПТК Формирование флагов общих команд по команде с ЧМИ	Ф	Входы (Inputs) <ul style="list-style-type: none"> • AllowMask: ARRAY[0..15] OF WORD := [16(1)]; // Уровень доступа Входы-выходы (InOuts) <ul style="list-style-type: none"> • Cmd: tCommonCmd; // Флаги обобщенных команд • Hist: tHistory; // Структура дискретной истории • CmdHMI: tCmdHMI; // Команда ЧМИ • Psw: tPsw; // Уровень доступа

Перечень обобщенных пользовательских команд приведен в табл. 2.15.3.

Номер группы команд: 300.

Смысл поля «Номер в группе»: не используется.

Таблица 2.15.3.

Номер команды	Название команды	Тип поля «Данные команды»
1	Общий сброс сигнализации	-
4	Общая деблокировка защит	-
6	Опробование световой предупредительной сигнализации	-
8	Опробование световой аварийной сигнализации	-
10	Опробование звуковой предупредительной сигнализации	-
11	Опробование звуковой аварийной сигнализации	-

Перечень регистрируемых обобщенных пользовательских команд приведен в табл. 2.15.4.

Номер группы событий: 300.

Смысл поля «Номер в группе»: не используется.

Таблица 2.15.4.

Номер события	Название события
1	Команда оператора 'Общее квитирование/сброс сигнализации'
4	Команда оператора 'Общая деблокировка защит'
6	Команда оператора 'Опробование световой предупредительной сигнализации'
8	Команда оператора 'Опробование световой аварийной сигнализации'
10	Команда оператора 'Опробование звуковой предупредительной сигнализации'
11	Команда оператора 'Опробование звуковой аварийной сигнализации'

3 ЖИЗНЕННЫЙ ЦИКЛ ПО

3.1 Требования к эксплуатационному персоналу

Персонал, участвующий в эксплуатации ПО, может выступать в роли разработчика прикладного программного обеспечения ПЛК, должен иметь соответствующую квалификацию (программист).

Программист – специалист, осуществляющий разработку прикладного программного обеспечения ПТК «Апогей-М» с использованием ПО.

Требования к программисту:

- знание языков программирования программируемых логических контроллеров в соответствии с ГОСТ Р МЭК 61131-3-2016;
- достаточные знания в части объекта автоматизации;
- достаточные знания в части применяемого контроллерного оборудования;
- прохождение обучения использованию ПО или самообучение на основе документации.

3.2 Устранение неисправностей

В случае нештатной работы ПО в ходе его эксплуатации, программист должен обратиться к разработчику ПО, максимально точно и подробно описав суть неисправности (внешние проявления, условия возникновения и т. д.).

При необходимости, разработчиком может быть запрошена дополнительная информация (фрагменты прикладного обеспечения, системные журналы, конфигурация оборудования и т. д.), которая рассматривается разработчиком как конфиденциальная и не подлежит распространению, но, по согласованию с программистом, может долгосрочно сохраняться разработчиком для обеспечения дальнейшей технической поддержки.

Разработчик, на основе обращения программиста, проводит анализ возникшей неисправности и оказывает консультативную помощь или проводит работу по устранению неисправности в рамках договорных отношений заказчика и разработчика.

3.3 Техническая поддержка

Информацию о замечаниях и неисправностях ПО, выявленных в ходе его эксплуатации, следует направлять разработчику ПО по следующим контактным данным:

- сайт: www.rakurs.com;
- электронная почта: info@rakurs.com;
- телефон: (812) 252-32-44.

По этим же контактным данным можно направлять предложения по развитию и совершенствованию ПО и все другие вопросы, связанные с программой.

При поставке ПО в составе автоматизированной системы, на него распространяются гарантийные обязательства по самой автоматизированной системе. При поставке ПО отдельно, условия поддержки пользователей определяются договором.

В ходе работ по развитию ПО и устранению выявленных замечаний и неисправностей, выпускаются новые версии ПО. Обновление ПО выполняется по согласованию заказчика и

разработчика, и может быть выполнено как специалистами заказчика по инструкциям, предоставляемым разработчиком, так и специалистами разработчика, как удалённо, так и с выездом на место. При этом вопросы совместимости версий и возможности обновления ПО необходимо оговаривать с разработчиком. Перечень версий ПО с указанием внесённых изменений ведётся разработчиком и может быть предоставлен заказчику для решения вопросов совместимости версий и целесообразности обновления ПО.

3.4 Жизненный цикл ПО

Поддержание жизненного цикла ПО, в том числе разработка, пуско-наладочные работы на объекте автоматизации, техническая поддержка заказчика, получение обратной связи от заказчика, регламентируются внутренними документами Системы менеджмента качества разработчика, сертифицированной на соответствие ISO 9001:2015.