



EVD LCD PROG

Контроллеры, программируемые на языке C



РУКОВОДСТВО ПО ЭКСПЛУАТАЦИИ



Важная информация

Перед началом монтажа и использования устройств полностью прочитайте данное руководство и выполняйте все приведенные инструкции по технике безопасности. Сохраните руководство для использования в будущем, храните его поблизости от устройств.

Используйте устройства только так, как описано в руководстве, не используйте их в качестве устройств безопасности.



Утилизация

Устройства следует утилизировать в соответствии с действующими местными нормами и правилами, регламентирующими сбор отходов электрического и электронного оборудования и их утилизацию.

1	ВВЕДЕНИЕ	4
1.1	Введение	4
1.2	Доступные модели, их технические особенности и коды для заказа	4
2	ОПИСАНИЕ	6
2.1	Описание прибора EVD	6
3	РАЗМЕРЫ И УСТАНОВКА	7
3.1	Размеры и установка прибора EVD	7
3.2	Меры техники безопасности при установке приборов	8
4	ЭЛЕКТРОПОДКЛЮЧЕНИЕ	9
4.1	Входы и выходы	9
4.2	Описание разъемов	10
4.2.1	Описание разъемов EVD	10
4.3	Подключение шунтирующего резистора к порту подчиненного устройства с интерфейсом RS-485 MODBUS	13
4.4	Пример электроподключения	13
4.4.1	Пример электроподключения контроллера EVD	13
4.5	Меры техники безопасности при работе с электрооборудованием	14
4.6	Примечания	14
4.7	Программирование прибора	14
4.7.1	Начальное программирование	14
4.7.2	Программирование при обновлении версии FW	14
5	ОПИСАНИЕ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА	15
5.1	Функции кнопок	15
5.2	Дисплей	15
5.3	Сигналы	15

6	ПРОГРАММИРОВАНИЕ	16
6.1	Предварительная информация	16
6.2	Компиляция и отладка проекта	16
6.2.1	С чего начать	16
6.2.2	Отладка	18
6.3	Использование кириллицы	20
6.4	Функции и макрокоманды	20
6.5	ПАРАМЕТРЫ	26
6.5.1	Таблица параметров par_ind[]	27
6.5.2	Таблицы дескриптора par_lim_word[] и par_lim_byte[]	27
6.5.3	Процедура добавления нового параметра	28
7	ВХОДЫ И ВЫХОДЫ	28
7.1	Входы	28
7.2	ВЫХОДЫ	28
7.2.1	Цифровые выходы	28
7.2.2	Аналоговые выходы	29
7.2.2.1	Тип отключения фазы	30
7.2.2.2	Тип 0-10 В	30
7.2.2.3	Тип PWM	30
7.2.2.4	Тип "частота"	30
7.2.3	Выход с открытым коллектором	30
7.3	Порт последовательного интерфейса с протоколом MODBUS	31
8	ПРИНАДЛЕЖНОСТИ	32
8.1	EVIF20SUXI -последовательный интерфейс RS-485/USB без оптронной развязки	32
8.1.1	Предварительная информация	32
8.2	CJAV - набор для подключения	32
8.2.1	Предварительная информация	32
9	ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ	33
9.1	Технические характеристики	33

1 ВВЕДЕНИЕ

1.1 Введение

Прибор EVD LCD PROG - программируемый контроллер для применений HVAC/R. Он может быть запрограммирован непосредственно на языке C.

Прибор поставляется в стандартном 4-модульном формате DIN; в версии с встроенным дисплеем присутствует жидкокристаллический (ЖК) дисплей с светодиодной подсветкой и разрешением 128x64 пикселей.

В настоящее время прибор EVD LCD PROG поставляется с блоком питания от электросети переменного тока напряжением 115 - 230 В.

Конструкция прибора EVD LCD PROG рассчитана на его установку на DIN-рейку в панели управления.

1.2 Доступные модели, их технические особенности и коды для заказа

Доступные модели прибора, их технические особенности и коды для заказа приведены в таблице ниже.

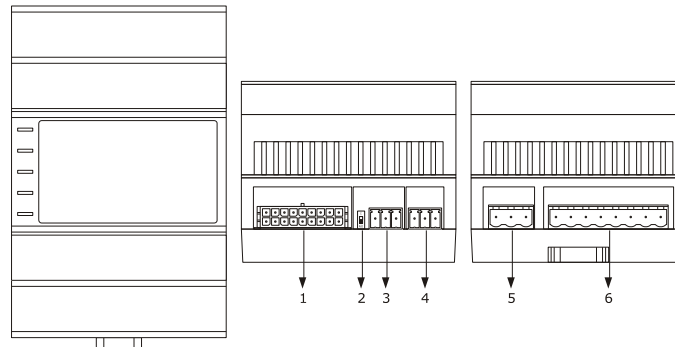
Доступные модели >	EVD LCD (низкое напряжение)	EVD LCD (высокое напряжение)
Исполнение		
Без дисплея		
с встроенным ЖК-дисплеем (разрешение 128x64 пикселей)		•
Светодиодные индикаторы сигналов		5
Соединения		
Разъемы Micro-Fit		•
Разъемы Edge		
Съемные блоки клемм на винтах		•
Электропитание		
12..24 Vac/dc (не изолированное?)	•	
115... 230 Vac, изолированное)		•
Аналоговые входы		
NTC		5
NTC/4-20 мА/0-10 В		2
Цифровые входы		
Переключаемый контакт		5

Аналоговые выходы		
0-10 В/с отключением фазы/PWM/ частота		2
Цифровые выходы (электромеханические реле; активная нагрузка @ 250 Vac)		
Реле		4
Цифровые выходы (открытый коллектор)		
12 Vdc, 40 mA макс.		1
Коммуникационные порты		
INTRABUS		ОТКЛЮЧЕН
RS485 modbus		•
Другие особенности		
динамик		•
RTC		•

2 ОПИСАНИЕ

2.1 Описание прибора EVD

Компоновочный чертеж прибора EVD см. ниже.



Все части прибора EVD описаны в таблице ниже.

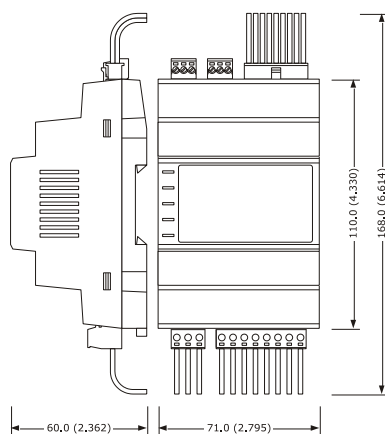
ЧАСТЬ	ОПИСАНИЕ
1	Штекерный разъем Micro-Fit для подключения кабелей к аналоговым входам, цифровым входам, аналоговым выходам и к цифровому выходу с открытым коллектором.
2	Микрореле для шунтирующего резистора порта подчиненного устройства с интерфейсом RS-485 MODBUS.
3	Съемный блок клемм на винтах, только штекер, для подключения к порту подчиненного устройства с интерфейсом RS-485 MODBUS.
4	Съемный блок клемм на винтах, только штекер, для подключения к порту INTRABUS с питанием.
5	Съемный блок клемм на винтах, только штекер, для подключения кабелей к цифровым выходам на электромеханическое реле (для будущего использования, цифровые выходы DO1 and DO2).
6	Съемный блок клемм на винтах, только штекер, для подключения кабелей к цифровым выходам на электромеханическое реле (для будущего использования, цифровые выходы DO3 и DO4).
7	Съемный блок клемм на винтах, только штекер, для подключения к блоку питания.

Дальнейшую информацию см. в следующих разделах.

3 РАЗМЕРЫ И УСТАНОВКА

3.1 Размеры и установка прибора EVD

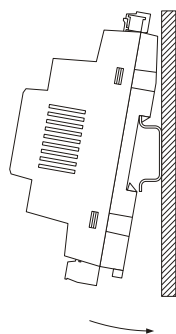
Размеры прибора EVD (4 модуля DIN) приведены на рисунках ниже; размеры даны в мм (дюймах).



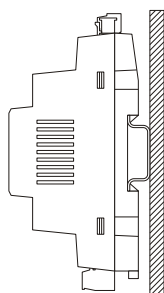
При установке на DIN-рейке: 35,0 x 7,5 мм (1,377 x 0,295 дюйма); при установке в панели управления: 35,0 x 15,0 мм (1,377 x 0,590 дюйма).

Компоновочный чертеж прибора EVD см. на рисунке ниже.

1

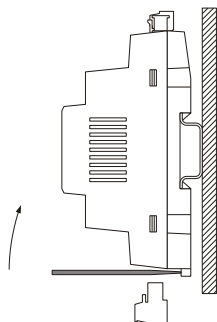


2

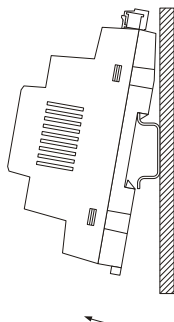


Чтобы извлечь EVD, сначала снимите все съемные блоки клемм на винтах, находящиеся в нижней части корпуса; затем с помощью отвертки освободите контроллер от скоб с защелками, крепящих контроллер к DIN-рейке (см. рисунок ниже).

3



4



Чтобы снова установить EVD, сначала нажмите на защелку DIN-рейки, чтобы полностью ввести ее.

3.2 Меры техники безопасности при установке приборов

- Условия работы прибора должны быть допустимыми (рабочие температуры, влажность и т. д. должны быть в пределах допустимых значений). См. раздел ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ.
- Не устанавливайте прибор вблизи источников тепла (рядом с нагревательными приборами, потоками горячего воздуха и т. д.); в местах, где присутствуют сильные электромагнитные поля (около больших вентиляторов и т.п.), в местах, подверженных воздействию прямых солнечных лучей, дождя, влаги, пыли, механических вибраций и ударов;
- в соответствии с нормами безопасности, защита от постороннего электрического воздействия должна быть обеспечена точной установкой прибора; части, обеспечивающие защиту, должны быть установлены таким образом, чтобы демонтаж осуществлялся только при помощи специального инструмента.

4 ЭЛЕКТРОПОДКЛЮЧЕНИЕ

4.1 Входы и выходы

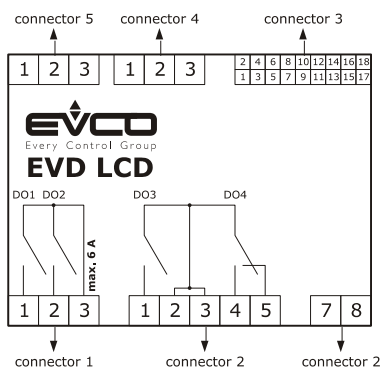
Назначение входов и выходов приведено в таблице ниже.

АНАЛГОВЫЕ И ЦИФРОВЫЕ ВХОДЫ	EVD
IN1 (NTC-DI / 4-20 мА / 0-10 В)	•
IN2 (NTC-DI / 4-20 мА / 0-10 В)	•
IN3 (NTC-DI)	•
IN4 (NTC-DI)	•
IN5 (NTC-DI)	•
IN9 (NTC-DI)	•
IN10 (NTC-DI)	•
ЦИФРОВЫЕ ВХОДЫ (переключаемый контакт)	EVD
IN6	•
ЦИФРОВЫЕ ВХОДЫ (переключаемый контакт) / СЧЕТЧИК ИМПУЛЬСОВ	EVD
IN7	•
IN8	•
АНАЛГОВЫЕ И ЦИФРОВЫЕ ВЫХОДЫ	EVD
AO1 (0-10 В / PWM / Отключение фазы / Частота)	•
AO2 (0-10 В / PWM / Отключение фазы / Частота)	•
ЦИФРОВЫЕ ВЫХОДЫ	EVD
DO1 (3A)	•
DO2 (3A)	•
DO3 (8A)	•
DO4 (12A)	•
Цифровые выходы с открытым коллектором	EVD
OC1	•

4.2 Описание разъемов

4.2.1 Описание разъемов EVD

Расположение разъемов прибора EVD см. на рисунке ниже.



connector	разъем
-----------	--------

Разъемы контроллера EVD описаны в таблицах ниже.

Разъем 1

ЧАСТЬ	ОПИСАНИЕ
1	Цифровой выход на электромеханическое реле DO1 (3A SPST): нормально разомкнутые контакты.
2	Цифровой выход на электромеханическое реле DO2 (3A SPST): нормально разомкнутые контакты.
3	Цифровые выходы на электромеханические реле DO1 и DO2 (макс. 6 А): общий провод.

Разъем 2

ЧАСТЬ	ОПИСАНИЕ
1	Цифровой выход на электромеханическое реле DO3 (12A SPST): нормально разомкнутые контакты.
2	Цифровой выход на электромеханические реле DO3 и DO4 (макс. 6 А): общий провод.
3	Цифровой выход на электромеханические реле DO3 и DO4 (макс. 6 А): общий провод.
4	Цифровой выход на электромеханическое реле DO4 (8A SPDT): нормально разомкнутые контакты.
5	Цифровой выход на электромеханическое реле DO4 (8A SPST): нормально замкнутые контакты.

Разъем 2

ЧАСТЬ	ОПИСАНИЕ
7	Блок питания EVD HP (115... 230 Vac, изолированное)
8	Блок питания EVD HP (115... 230 Vac, изолированное)

Разъем 3

ЧАСТЬ	ОПИСАНИЕ
1	Аналоговый выход AO2
2	Аналоговый выход AO1
3	Нулевой потенциал (земля)
4	Аналоговый вход IN1
5	Цифровой вход IN10: переключаемый контакт
6	Аналоговый вход IN2
7	Цифровой вход IN9: переключаемый контакт
8	Аналоговый вход IN3
9	Цифровой вход (переключаемый контакт) / импульсный вход (макс. 200 Гц) IN8
10	Аналоговый вход IN4 (NTC)
11	Цифровой вход (переключаемый контакт) / импульсный вход (макс. 200 Гц) IN7
12	Аналоговый вход IN5
13	Нулевой потенциал (земля)
14	Цифровой вход IN6: переключаемый контакт
15	Не используется
16	Выход 12 Vdc, 120 мА макс.
17	Цифровой выход с открытым коллектором OC1 (12 В, макс. 40 мА)
18	Нулевой потенциал (земля)

Разъем 4 (если установлен)

ЧАСТЬ	ОПИСАНИЕ
1	Положительный сигнал порта ведомого устройства RS-485 шины MODBUS
2	Отрицательный сигнал порта ведомого устройства RS-485 шины MODBUS
3	Нулевой потенциал (земля)

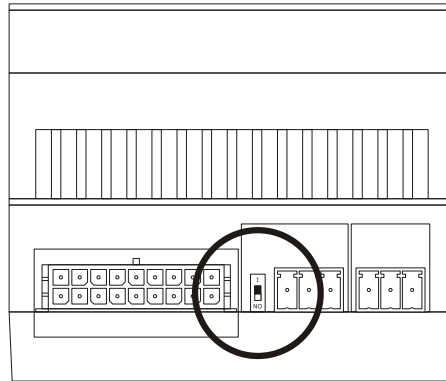
Разъем 5

ЧАСТЬ	ОПИСАНИЕ
1	Порт INTRABUS, сигнал "питание подано"
2	Нулевой потенциал (земля)
3	Блок питания 12 Vdc

4.3 Подключение шунтирующего резистора к порту подчиненного устройства с интерфейсом RS-485 MODBUS

Чтобы уменьшить отражения сигнала, передаваемого по кабелям, которыми устройство подключено к сети RS-485, необходимо установить резисторы нагрузки на первом и последнем элементах сети.

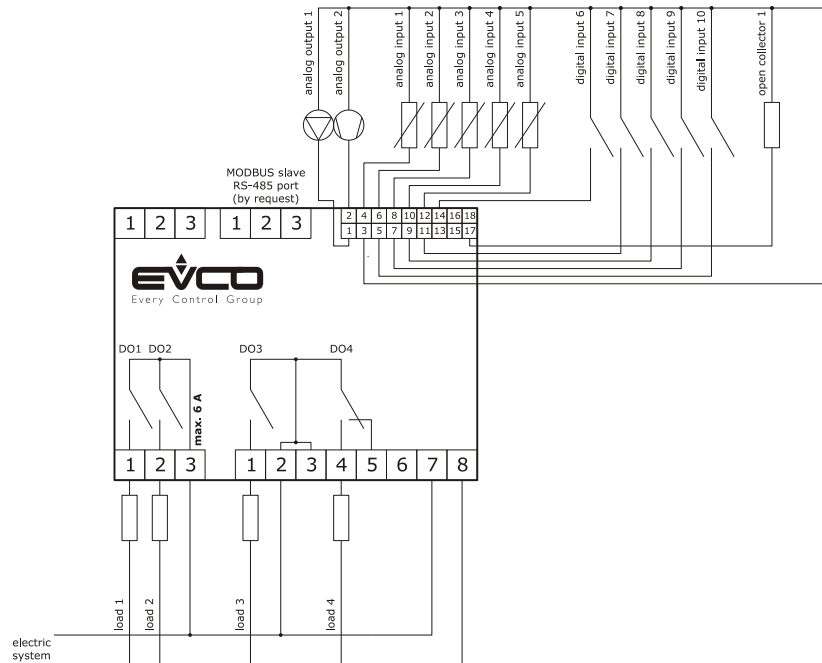
Чтобы подключить шунтирующий резистор к контроллеру EVD, установите микропереключатель 1 в положение ON.



4.4 Пример электроподключения

4.4.1 Пример электроподключения контроллера EVD

Примерная схема электроподключения контроллера EVD приведена на рисунке ниже.



MODBUS slave RS-485 port (by request)	Порт RS-485 с протоколом подчиненного устройства MODBUS (по заказу)
analog output	аналоговый выход
analog input	аналоговый вход
digital input	цифровой вход
open collector	открытый коллектор
load	нагрузка
electric system	электрическая система

4.5 Меры техники безопасности при работе с электрооборудованием

- при работе с блоками клемм прибора запрещается использовать электрические или пневматические отвертки;
- если прибор был перенесен из холодного помещения в теплое, внутри него может сконденсироваться влага; Дайте прибору побыть в тепле хотя бы 1 час, только после этого на него можно подавать питание.
- Убедитесь, что напряжение питания, частота электросети и мощность, потребляемая прибором соответствуют параметрам местной электросети. См. раздел ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ.
- Перед выполнением какого-либо обслуживания, отсоедините прибор от источника питания.
- Питание прибора должно осуществляться от той же фазы, от которой питаются и модули с сигналом команды отключения фазы.
- Если используются цифровые выходы с симисторами, рекомендуется включить шумовой фильтр; не касайтесь радиатора - его поверхность может нагреваться до высоких температур.
- При подключении прибора к сети RS-485 используйте экранированный кабель "витая пара", с третьим проводом для подключения к точке с нулевым потенциалом (GND). Рекомендуется использовать кабель BELDEN 3106A.
- Прокладывайте кабели электропитания как можно дальше от сигнальных кабелей;
- Не используйте прибор в качестве устройства безопасности.
- Для ремонта и информации о приборе, пожалуйста, свяжитесь с отделом продаж EVCO.

4.6 Примечания

- От главного контроллера не должно осуществляться электропитание каких-либо устройств.
- Не подключайте к прибору удлинители или удаленные дисплеи.

4.7 Программирование прибора

4.7.1 Начальное программирование

Является первым этапом процедуры программирования, использующей файл *.hex, устанавливает частоту подстройки 38400 Гц.

4.7.2 Программирование при обновлении версии FW

Прибор может быть обновлен по последовательному интерфейсу RS485 при использовании файла *.bin (этапы создания программы и файла прошивки *.bin см §6.2).

Следуйте следующей процедуре:

- запустите приложение "гипертерминал" со следующими параметрами: 38400,N,8,1 по HW check, в установках ASCII можно включить опцию "Echo typed...", чтобы видеть вводимые символы;

Организуйте последовательное соединение с интерфейсом 485:

- выключите устройства и снова включите его;
- в строке ввода/вывода команд должен появиться символ параграфа;
- в течение первых 10 секунд после появления символа параграфа напечатайте на клавиатуре 'boot';
- если устройство запустит процедуру, будет отображено "PA60/32 Bootloader By Evco".
- очистите память (1);
- включите программирование (2);
- войдите в меню menu -> transfer--> send file, тип передачи – Xmodem, выберите файл *.bin;






При программировании через загрузчик (bootloader) параметры будут перезагружены только с значениями по умолчанию (если версия была изменена, см. соответствующий параграф).

Калибровка останется работоспособной.

5 ОПИСАНИЕ ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА

5.1 Функции кнопок

Функции кнопок прибора описаны в таблице ниже.


ОБОЗНАЧЕНИЕ	НАИМЕНОВАНИЕ
	Esc (Выход)
SET	Enter (Ввод)
	Up (Вверх)
	Down (Вниз)
	Left (Влево)
	Right (Вправо)

5.2 Дисплей

Для интерфейса с пользователем применен ЖК дисплей с подсветкой; разрешение дисплея 64x128 пикселей.

5.3 Сигналы

Описание светодиодных индикаторов сигналов на передней панели прибора EVD см. в таблице ниже.

СИГНАЛ	ЦВЕТ	ОПИСАНИЕ
ON	ЗЕЛЕНЫЙ	Индикатор включения аппаратной части. Если значок горит, на прибор подано питание.
RUN	ЗЕЛЕНЫЙ	Индикатор работоспособности. Программируется.
	КРАСНЫЙ	Значок сигнала тревоги Программируется.
IB	ОРАНЖЕВЫЙ	Индикатор INTRABUS - мигает при обмене данными; - Если значок не горит, связь отсутствует.
RS485	ОРАНЖЕВЫЙ	Индикатор Modbus - мигает при обмене данными; - Если значок не горит, связь отсутствует.

6 Программирование

6.1 Предварительная информация

Прикладную часть (настройки и пользовательский интерфейс) контроллера можно запрограммировать на языке C. Контроллер поставляется вместе с средой разработки Code Warrior (на основе Eclipse) и библиотеками (низкоуровневым драйвером), уже протестированными и откалиброванными в процессе производства. Среда разработки уже поставляется - вместе с файлами разработчика и подробным примером программирования.

Архитектура программного обеспечения (ПО) базируется на главной задаче **main**, использующей "бесконечный цикл" **Control()**. Присвоение начальных значений осуществляется входящей в комплект ПО функцией **init_Control()**.

```
main()
{
...
// присвоение различных начальных значений
init_Control
...
while (1)
{
....
Control();
...
}
```

Эти файлы можно как угодно редактировать; они находятся в папках Control и Lcd. Для ввода новых параметров вместо имеющихся, необходимо отредактировать файлы tabpar.c и tabpar.h.

Остальные файлы не следует редактировать ни при каких обстоятельствах.

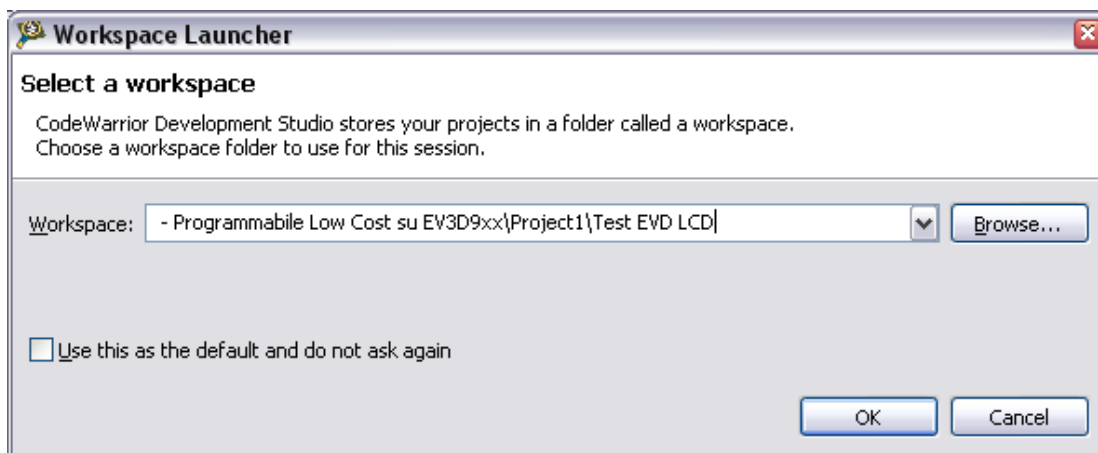
6.2 Компиляция и отладка проекта

6.2.1 С чего начать

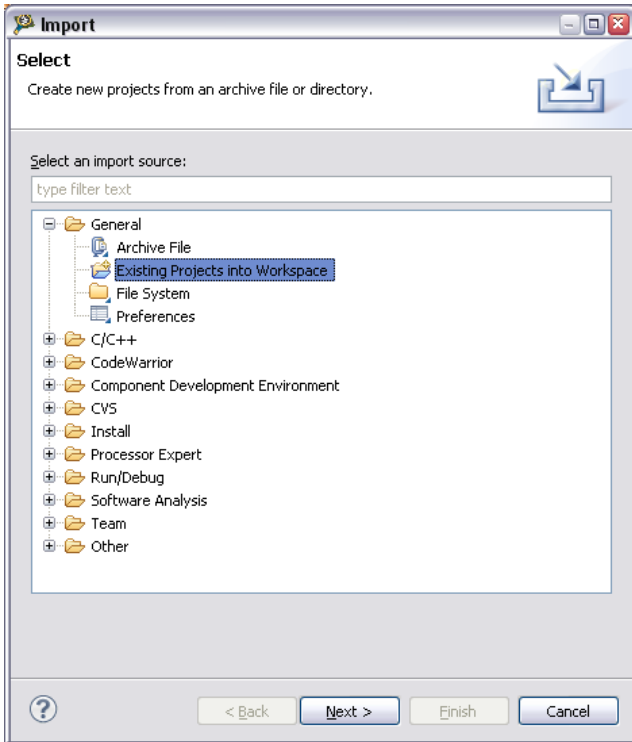
Для компиляции проекта необходимо установить на вашем компьютере среду разработки "Code Warrior". Установочные файлы доступны для загрузки непосредственно с веб-сайта компании NXP по следующей ссылке (потребуется регистрация):

http://www.nxp.com/products/software-and-tools/software-development-tools/codewarrior-development-tools/codewarrior-development-studios/codewarrior-for-microcontrollers/codewarrior-for-mcus-eclipse-ide-coldfire-56800-e-dsc-kinetis-freescale-56xx-rs08-s08-s12z-v10.6:CW-MCU10?code=CW-MCU10&nodeId=015210BAF72726E4C7E4CB&fsp=1&tab=Design_Tools_Tab

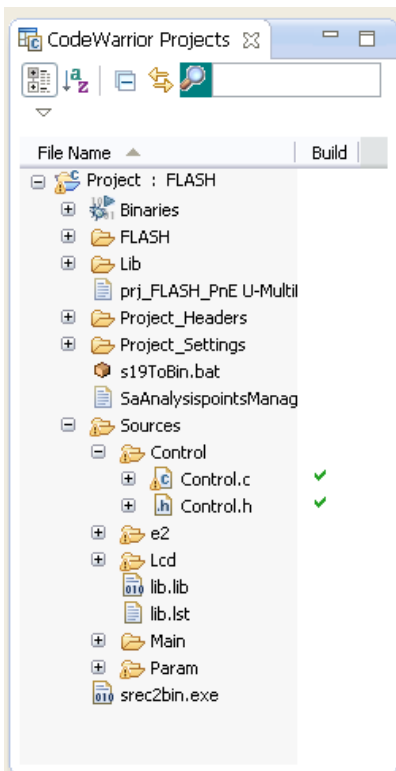
При запуске Code Warrior, вы должны выбрать рабочую папку, в которой будет содержаться рабочее пространство:



При первом запуске программы вам необходимо импортировать существующий проект в рабочее пространство:



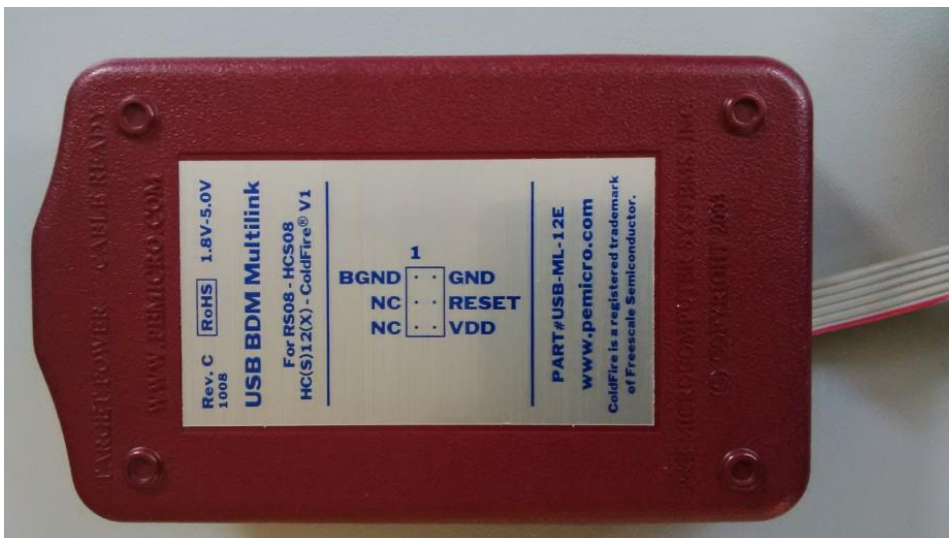
После нажатии на кнопку "Next" нужно указать папку с проектом/проектами на компьютере, выбрать необходимые для импорта. После нажатия на кнопку "Finish", все файлы для работы будут созданы. В дереве проекта будет отображаться следующая структура:



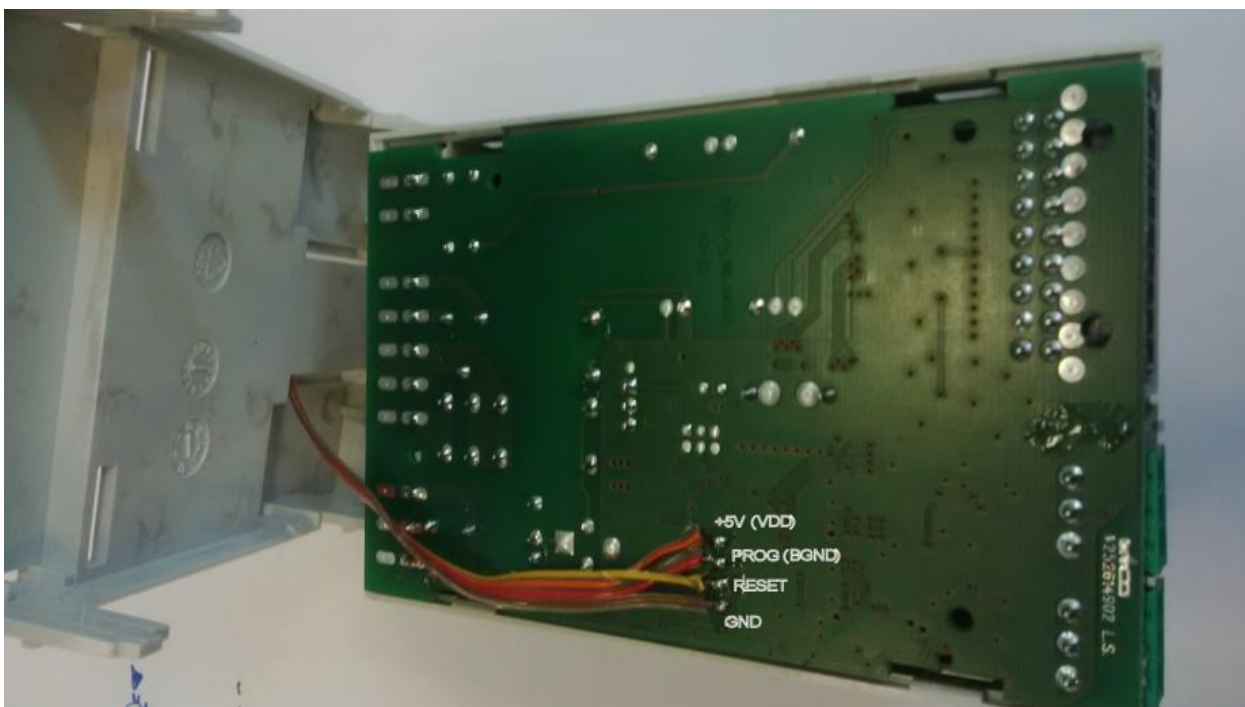
Из меню "Project" можно скомпилировать или очистить проект. Если результат компиляции будет без ошибок, то в подкаталоге "FLASH" будет создан файл *Project.abs.s19*. При запуске пакетного файла **s19ToBin.bat**, будет сгенерирован файл *Project.bin* для программирования контроллера (см §4.7).

6.2.2 Отладка

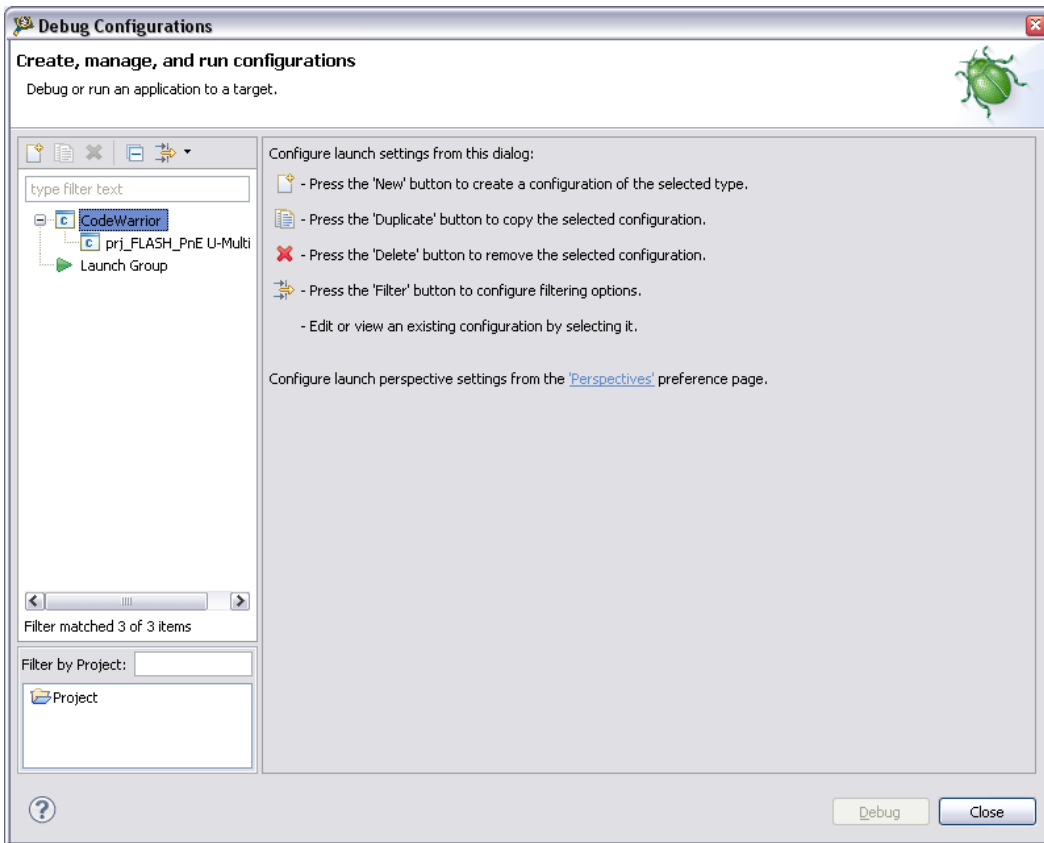
Для отладки используется устройство USB Multilink Interface:



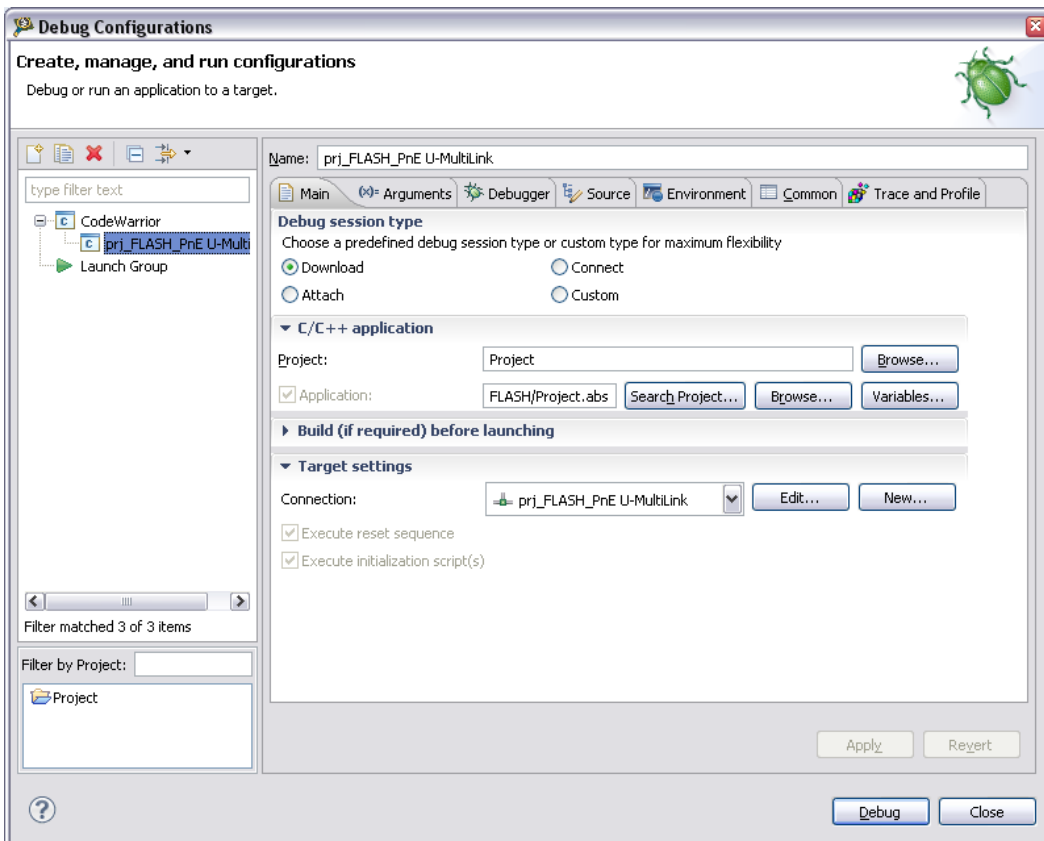
Контакты разъема устройства должны быть подключены к контроллеру, описание контактов показано на следующем рисунке:



В меню "Run" выберите "Debug Configurations":



Раскройте элемент "CodeWarrior" для выбора "prj_FLASH_PnE U-MultiLink":



Для того, чтобы запустить программу и начать сеанс отладки, выберите меню "Run->Debug". Все функции отладки описаны в документации к среде разработки "CodeWarrior" (точки останова, переменные часы и т.д.).

6.3 Использование кириллицы

В комплект поставляемого ПО входит инструмент **U2C**, осуществляющий преобразование кириллических строк в формат UTF-8.

Пример:

- Введите строку, набранную кириллицей, которую требуется преобразовать, в файл с именем, например, c.txt.
- Запустите программу U2C, которая выберет этот файл.
- Из преобразованного файла с.с. извлеките строку в формате


```
my_string[] = "\x....."
```

Когда строки с буквами и другими знаками будут сгенерированы, будут созданы и файлы для управления ими.

Примеры созданных файлов:

ascii_arial19.c и ascii_arial19.h

cyrillic_arial19.c и cyrillic_arial19.h

symbol.c и symbol.h

Эти файлы должны быть исключены из компиляции.

Включить содержимое таблиц этих файлов в проект можно, добавив в начало файла lcd.c следующие строки:

```
#include "ascii_arial19.c"
#include "cyrillic_arial19.c"
#include "symbol.c"
```

6.4 Функции и макрокоманды

Файл ControlDefines.h содержит все функции и макросы, которые должны использоваться для интерфейса с библиотекой драйвера; этот файл дан лишь для информационных целей, изменять его не следует.

```
//Области видимости параметров
#define P_HID_0 0
#define P_USR_1 1
#define P_SER_2 2
#define P_MAN_3 3

//Список системных параметров
#define IND_COM_IB_ADD 0
#define IND_COM_IB_BAUD 1
#define IND_COM_IB_PAR 2
#define IND_COM_MB_ADD 3
#define IND_COM_MB_BAUD 4
#define IND_COM_MB_PAR 5
#define IND_LCD_BACKLIGHT_MODE 6
#define IND_LCD_BACKLIGHT_TIME 7
#define IND_LCD_CONTRAST 8
#define IND_PASSWORD_USR_1 9
#define IND_PASSWORD_SER_2 10
#define IND_PASSWORD_MAN_3 11
#define IND_TYPE_AI1 12
#define IND_TYPE_AI2 13
#define IND_TYPE_AI 14
#define IND_TYPE_AO1 15
#define IND_TYPE_AO2 16
#define IND_TYPE_TK1 17
#define IND_TYPE_TK2 18
#define IND_PWM_FREQ 19
#define IND_PC_TIME_AO1 20
#define IND_PC_TIME_AO2 21
#define IND_PC_TIME_TK1 22
#define IND_PC_TIME_TK2 23
#define IND_PC_SHIFT_AO1 24
#define IND_PC_SHIFT_AO2 25
#define IND_PC_SHIFT_TK1 26
#define IND_PC_SHIFT_TK2 27
#define IND_FREE_01 28
#define IND_FREE_02 29
#define IND_FREE_03 30
#define IND_FREE_04 31
#define IND_FREE_05 32
#define IND_FREE_06 33
#define IND_FREE_07 34
#define IND_FREE_08 35
#define IND_FREE_10 36
#define IND_FREE_11 37
#define IND_FREE_12 38
#define IND_FREE_13 39

//Маски и индексы ввода-вывода
#define ANALOGIN1 0
#define ANALOGIN2 1
#define ANALOGIN3 2
#define ANALOGIN4 3
```

```

#define ANALOGIN5          4
#define ANALOGIN10         5
#define ANALOGIN9          6
#define DIGITALIN1         0
#define DIGITALIN2         1
#define DIGITALIN3         2
#define DIGITALIN4         3
#define DIGITALIN5         4
#define DIGITALIN10        5
#define DIGITALIN9         6
#define DIGITALIN8         7
#define DIGITALIN7         8
#define DIGITALIN6         9
#define DIGITALIN1_MSK     0x0001
#define DIGITALIN2_MSK     0x0002
#define DIGITALIN3_MSK     0x0004
#define DIGITALIN4_MSK     0x0008
#define DIGITALIN5_MSK     0x0010
#define DIGITALIN10_MSK    0x0020
#define DIGITALIN9_MSK     0x0040
#define DIGITALIN8_MSK     0x0080
#define DIGITALIN7_MSK     0x0100
#define DIGITALIN6_MSK     0x0200
#define DIGITALIN8_PULSE   0
#define DIGITALIN7_PULSE   1
#define ANALOGOUT1         0
#define ANALOGOUT2         1
#define OPENCOLL1          2
#define RELAY1              0
#define RELAY2              1
#define RELAY3              2
#define RELAY4              3
#define RELAYA01            4
#define RELAYA02            5
#define RELAYOC1           6
#define RELAY1_MSK         0x00000001
#define RELAY2_MSK         0x00000002
#define RELAY3_MSK         0x00000004
#define RELAY4_MSK         0x00000008
#define RELAYA01_MSK       0x00000010
#define RELAYA02_MSK       0x00000020
#define RELAYOC1_MSK       0x00000040

// Конфигурация входов и выходов
#define AICONF_TYPE_NTC     0
#define AICONF_TYPE_420mA  1
#define AICONF_TYPE_010V   2
#define AITEMP_TYPE_NTC    0
#define AO_TYPE_DIS         0
#define AO_TYPE_PHCUT       1
#define AO_TYPE_10V         2
#define AO_TYPE_PWM         3
#define AO_TYPE_FREQ        4
#define OC_TYPE_DIS         0
#define OC_TYPE_PHCUT       1

//RTC
#define RTC_SECOND          0
#define RTC_MINUTE          1
#define RTC_HOUR            2
#define RTC_DAY             3
#define RTC_MONTH           4
#define RTC_YEAR            5

//Светодиодные индикаторы, клавиатура и дисплей
#define KEY_NULL            0x00
#define KEY_ENTER           0x01
#define KEY_UP              0x02
#define KEY_DOWN            0x04
#define KEY_ESC             0x08
#define KEY_LEFT            0x10
#define KEY_RIGHT           0x20
#define LCD_BACKLIGHT_MODE_OFF 0
#define LCD_BACKLIGHT_MODE_ON 1
#define LCD_BACKLIGHT_MODE_TIMED 2
#define LED_RUN              0 //prog
#define LCD_NORMAL           0
#define LCD_INVERT           1
#define LCD_ROW1             0
#define LCD_ROW2             2
#define LCD_ROW3             4
#define LCD_ROW4             6
#define LED_ALARM            1 //prog
//Режим мигания
#define ModeOFF              0
#define ModeON               1
#define ModeBLK              2 //1s
#define ModeBLKF             3 //400ms

```

```

//Связь
#define BAUD_RATE_2400      0
#define BAUD_RATE_4800      1
#define BAUD_RATE_9600      2
#define BAUD_RATE_19200     3
#define BAUD_RATE_38400     4
#define BAUD_RATE_57600     5
#define BAUD_RATE_76800     6
#define NO_PARITY           0
#define ODD_PARITY          1
#define EVEN_PARITY         2

//Информация
#define INFO_PRJNUM          1
#define INFO_PRJVER_PARVER  2
#define INFO_LIBVERS        3
#define INFO_LIBID          7
#define INFO_LIBVAR_LIBREV  8

/**
 * @brief Запись в аналоговый выход
 * @param ind : индекс аналогового выхода.
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg ANALOGOUT1: запись значения в A01
 *          @arg ANALOGOUT2: запись значения в A02
 *          @arg OPENCOLL1: запись значения в OC1
 * @param value: значение, которое будет записано в выбранный аналоговый выход.
 *          Значение может быть в процентах (от 0 до 100%), либо частотой (от 0 до 255 Гц),
 *          в зависимости от типа аналогового выхода.
 * @retval Нет
 */
extern void WriteAOValue (uint8_t ind, uint8_t value);
/**
 * @brief Запись в цифровой выход
 * @param ind : индекс цифрового выхода.
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg RELAY1: запись значения в NO1
 *          @arg RELAY2: запись значения в NO2
 *          @arg RELAY3: запись значения в NO3
 *          @arg RELAY4: запись значения в NO4
 *          @arg RELAYA01: запись значения в A01
 *          @arg RELAYA02: запись значения в A02
 *          @arg RELAYOC1: запись значения в OC1
 * @param value: значение, которое будет записано в выбранный цифровой выход.
 * @retval Нет
 */
extern void WriteDOValue (uint8_t ind, bool value);
/**
 * @brief Запись в упакованные цифровые выходы
 * @param value: значение, которое будет записано в выбранный упакованный цифровой выход.
 * @retval Нет
 */
extern void WriteDoValuePacket (uint16_t value);
/**
 * @brief Чтение из аналогового выхода
 * @param ind : индекс аналогового выхода.
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg ANALOGOUT1: запись значения в A01
 *          @arg ANALOGOUT2: запись значения в A02
 *          @arg OPENCOLL1: запись значения в OC1
 * @retval чтение из выбранного аналогового выхода.
 *          Значение может быть в процентах (от 0 до 100%), либо частотой (от 0 до 255 Гц), в
 *          зависимости от типа аналогового выхода.
 */
extern uint16_t ReadAOValue (uint8_t ind);
/**
 * @brief Чтение из цифрового выхода
 * @param ind : индекс цифрового выхода.
 *          Этот параметр может принимать одно из следующих значений:
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg RELAY1: запись значения в NO1
 *          @arg RELAY2: запись значения в NO2
 *          @arg RELAY3: запись значения в NO3
 *          @arg RELAY4: запись значения в NO4
 *          @arg RELAYA01: запись значения в A01
 *          @arg RELAYA02: запись значения в A02
 *          @arg RELAYOC1: запись значения в OC1
 * @retval value: чтение из выбранных цифровых выходов.
 */
extern bool ReadDOValue (uint8_t ind);
/**
 * @brief Чтение упакованных цифровых выходов
 * @retval value: чтение из упакованных цифровых выходов.
 */
extern uint16_t ReadDOValuePacket (void);

```

```

/**
 * @brief Чтение из аналогового входа
 * @param ind : индекс аналогового входа.
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg ANALOGIN1
 *          @arg ANALOGIN2
 *          @arg ANALOGIN3
 *          @arg ANALOGIN4
 *          @arg ANALOGIN5
 *          @arg ANALOGIN10
 *          @arg ANALOGIN9
 * @retval чтение из выбранного аналогового входа.
 *          Значение может быть либо температурой (с 1 значащей цифрой после запятой), либо током или
напряжением (2 значащие цифры после запятой), в зависимости от типа аналогового выхода.
 */
extern int16_t ReadAIValue (uint8_t ind);
/**
 * @brief Чтение из цифрового входа
 * @param ind : индекс цифрового входа.
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg DIGITALIN1
 *          @arg DIGITALIN2
 *          @arg DIGITALIN3
 *          @arg DIGITALIN4
 *          @arg DIGITALIN5
 *          @arg DIGITALIN10
 *          @arg DIGITALIN9
 *          @arg DIGITALIN8
 *          @arg DIGITALIN7
 *          @arg DIGITALIN6
 * @retval чтение из выбранного цифрового входа.
 */
extern uint16_t ReadDIValue (uint8_t ind);
/**
 * @brief Чтение упакованных цифровых входов
 * @param ind : индекс цифрового входа.
 * @retval чтение из упакованного цифрового входа.
 */
extern uint16_t ReadDIValuePacket (void);
/**
 * @brief Чтение значения импульса
 * @param ind : индекс входа импульса.
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg DIGITALIN8_PULSE
 *          @arg DIGITALIN7_PULSE
 * @retval чтение значения импульса (от 0 до 200 Гц).
 */
extern uint16_t ReadDIpulseValue (uint8_t ind);
/**
 * @brief Запись состояния динамика
 * @param value: значение, которое будет записано в динамик.
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg ModeOFF
 *          @arg ModeON
 *          @arg ModeBLK
 *          @arg ModeBLKF
 * @retval Нет
 */
extern void WriteBuzzerValue (uint8_t value);
/**
 * @brief Запись состояния динамика
 * @retval чтение состояния динамика.
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg ModeOFF
 *          @arg ModeON
 *          @arg ModeBLK
 *          @arg ModeBLKF
 */
extern uint8_t ReadBuzzerValue (void);
/**
 * @brief Запись состояния светодиода
 * @param ind : индекс светодиода.
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg LED_RUN
 *          @arg LED_ALARM
 * @param value: значение, которое будет записано в светодиод.
 *          Этот параметр может принимать одно из следующих значений:
 *          @arg ModeOFF
 *          @arg ModeON
 *          @arg ModeBLK
 *          @arg ModeBLKF
 * @retval Нет
 */
extern void WriteLEDValue (uint8_t ind, uint8_t value);
/**
 * @brief Чтение состояния светодиода
 * @param ind : индекс светодиода.
 *          Этот параметр может принимать одно из следующих значений:

```

```

*           @arg LED_RUN
*           @arg LED_ALARM
* @retval чтение состояния светодиода.
*           Этот параметр может принимать одно из следующих значений:
*           @arg ModeOFF
*           @arg ModeON
*           @arg ModeBLK
*           @arg ModeBLKF
*/
extern uint8_t ReadLEDValue (uint8_t ind);
/**
* @brief Чтение состояния кнопок
* @retval чтение состояния кнопки.
* Кнопка: KEY_NULL
*         KEY_ENTER
*         KEY_UP
*         KEY_DOWN
*         KEY_ESC
*         KEY_LEFT
*         KEY_RIGHT
* flagKeyFirst:
* flagKeyLong:
* flagKeyRepeat:
* flagKeyRelease:
*/
extern T_Keys WhatKeys(void);
/**
* @brief Запись RTC
* @param ind : индекс поля RTC.
*           Этот параметр может принимать одно из следующих значений:
*           @arg RTC_SECOND
*           @arg RTC_MINUTE
*           @arg RTC_HOUR
*           @arg RTC_DAY
*           @arg RTC_MONTH
*           @arg RTC_YEAR
* @param value: значение, которое будет записано в выбранное поле RTC.
* @retval Нет
*/
extern void WriteRTCValue (uint8_t ind, uint8_t value);
/**
* @brief Чтение RTC
* @param ind : индекс поля RTC.
*           Этот параметр может принимать одно из следующих значений:
*           @arg RTC_SECOND
*           @arg RTC_MINUTE
*           @arg RTC_HOUR
*           @arg RTC_DAY
*           @arg RTC_MONTH
*           @arg RTC_YEAR
* @retval чтение из выбранного поля RTC.
*/
extern uint8_t ReadRTCValue (uint8_t ind);
/**
* @brief Чтение состояния RTC
* @retval считанное состояние RTC. Ошибки:
*         bit0 : общая ошибка RTC
*         bit1 : ошибка шины RTC
*/
extern uint8_t ReadRTCState(void);
/**
* @brief Чтение тика секунды
* @retval флаг тика секунды
*/
extern uint8_t SecondTic (void);
/**
* @brief Чтение тика 100ms
* @retval флаг тика 100 ms
*/
extern uint8_t HundredMillisecTic (void);
/**
* @brief Чтение тика половины секунды
* @retval флаг тика 500ms
*/
extern uint8_t HalfSecondTic (void);
/**
* @brief Чтение тика 10 секунд
* @retval флаг тика 10 секунд
*/
extern uint8_t TenSecondsTic (void);
/**
* @brief Чтение тика минуты
* @retval флаг тика минуты
*/

```



```

*
*/
extern uint8_t MinuteTic (void);
/**
 * @brief Чтение переключателя 500ms
 * @retval переключатель 500ms : HI 500ms или low 500ms
 */
extern uint8_t Toggle05 (void);
/**
 * @brief Чтение переключателя 200ms
 * @retval переключатель 200ms : HI 200ms или low 200ms
 */
extern uint8_t Toggle02 (void);
/**
 * @brief Чтение информации проекта
 * @param ind : индекс поля информации проекта.
 *           Этот параметр может принимать одно из следующих значений:
 *           @arg INFO_PRJNUM
 *           @arg INFO_PRJVER_PARVER
 *           @arg INFO_LIBVERS
 *           @arg INFO_LIBID
 *           @arg INFO_LIBVAR_LIBREV
 * @retval чтение из выбранного поля информации проекта.
 */
extern uint16_t ReadInfoProject (uint8_t ind);
/**
 * @brief Запись значения параметра
 * @param ind : индекс параметра.
 *           Этот параметр может принимать одно из следующих значений:
 *           @arg IND_COM_IB_ADD
 *           @arg IND_COM_IB_BAUD
 *           @arg IND_COM_IB_PAR
 *           .....
 *           @arg Другие IND_прикладные_параметры
 * @param value: значение, которое будет записано в выбранный параметр.
 * @retval Нет
 */
extern void WritePARValue (uint8_t ind, int16_t value);
/**
 * @brief Чтение значения параметра
 * @param ind : индекс параметра.
 *           Этот параметр может принимать одно из следующих значений:
 *           @arg IND_COM_IB_ADD
 *           @arg IND_COM_IB_BAUD
 *           @arg IND_COM_IB_PAR
 *           .....
 *           @arg Другие IND_прикладные_параметры
 * @retval чтение выбранного параметра.
 */
extern int16_t ReadPARValue (uint8_t ind);
/**
 * @brief Запись области видимости параметра
 * @param ind : индекс параметра.
 *           Этот параметр может принимать одно из следующих значений:
 *           @arg IND_COM_IB_ADD
 *           @arg IND_COM_IB_BAUD
 *           @arg IND_COM_IB_PAR
 *           .....
 *           @arg Другие IND_прикладные_параметры
 * @param value: значение области видимости, которое будет записано в выбранный параметр.
 * @retval Нет
 */
extern void WritePARVisib (uint8_t ind, uint8_t value);
/**
 * @brief Чтение области видимости параметра
 * @param ind : индекс параметра.
 *           Этот параметр может принимать одно из следующих значений:
 *           @arg IND_COM_IB_ADD
 *           @arg IND_COM_IB_BAUD
 *           @arg IND_COM_IB_PAR
 *           .....
 *           @arg Другие IND_прикладные_параметры
 * @retval чтение области видимости выбранного параметра.
 */
extern uint8_t ReadPARVisib (uint8_t ind);
/**
 * @brief Чтение минимума параметра
 * @param ind : индекс параметра.
 *           Этот параметр может принимать одно из следующих значений:
 *           @arg IND_COM_IB_ADD
 *           @arg IND_COM_IB_BAUD
 *           @arg IND_COM_IB_PAR
 *           .....
 *           @arg Другие IND_прикладные_параметры
 * @retval считанный минимум выбранного параметра.
 */
extern int16_t ReadPARmin (uint8_t ind);
/**

```

```

* @brief Чтение максимума параметра
* @param ind : индекс параметра.
*          Этот параметр может принимать одно из следующих значений:
*          @arg IND_COM_IB_ADD
*          @arg IND_COM_IB_BAUD
*          @arg IND_COM_IB_PAR
*          .....
*          @arg Другие IND_прикладные_параметры
* @retval считанный максимум выбранного параметра.
*/
extern int16_t ReadPARmax (uint8_t ind);
/**
* @brief Чтение атрибутов параметра
* @param ind : индекс параметра.
*          Этот параметр может принимать одно из следующих значений:
*          @arg IND_COM_IB_ADD
*          @arg IND_COM_IB_BAUD
*          @arg IND_COM_IB_PAR
*          .....
*          @arg Другие IND_прикладные_параметры
* @retval считанные атрибуты выбранного параметра. Атрибуты:
* bit0 : Параметр является (числом с 1 значащей цифрой после запятой) - MDEC
* bit1 : нижним пределом другого параметра - MRL
* bit1 : верхним пределом другого параметра - MRH
* bit3 : параметр является 8-разрядным числом с знаком - SCHAR
* bit4..7: не используются
*/
extern int16_t ReadPARatt (uint8_t ind);
/**
* @brief Чтение состояния буфера связи с ЖК-дисплеем
* @retval 1: шина свободна
*          0: шина занята (идет новая передача данных)
*/
extern uint8_t IsDispFree(void);
/**
* @brief Команды обновления ЖК-дисплея
* @retval Нет
*/
extern void UpdateDisp(void);
/**
* @brief Подготовка буфера к отправке данных в дисплей
* @param string : указатель на строку, которая должна быть отображена на дисплее
* @param x : отступ в пикселях
* @param row : индекс строки. *          Этот параметр может принимать одно из следующих значений:
*          @arg LCD_ROW1
*          @arg LCD_ROW2
*          @arg LCD_ROW3
*          @arg LCD_ROW4
* @param invert : подсветка строки. *          Этот параметр может принимать одно из следующих значений:
*          @arg LCD_NORMAL
*          @arg LCD_INVERT
* @retval Нет
*/
extern void Lcdprintf (char* string, uint8_t x,uint8_t row,uint8_t invert);
/**
* @brief пользовательская программа инициализации. Вызывается при запуске.
* @retval Нет
*/
extern void Init_Control(void);
/**
* @brief пользовательская программа управления. Всегда вызывается в главном цикле.
* @retval Нет
*/
extern void Control(void);
/**
* @brief чтение состояния занятости e2
* @retval 0: запрос записи в e2 отсутствует
*          1: запись в e2 остается отложенной
*/
extern bool IsE2Busy(void);
/**
* @brief пользовательская программа записи в modbus (FC 0x06, 0x10)
* @param Addr : адрес modbus.
* @param valueWrite: значение, которое требуется записать
* @retval Нет
*/
extern void WriteMBUser(uint16_t Addr, int16_t valueWrite);
/**
* @brief пользовательская программа считывания modbus (FC 0x03)
* @param Addr : адрес modbus.
* @retval прочитанное значение
*/
extern int16_t ReadMBUser(uint16_t Addr);

```

6.5 ПАРАМЕТРЫ

Для ввода или изменения параметров, необходимо отредактировать файлы tabpar.c и tabpar.h.

6.5.1 Таблица параметров par_ind[]

Каждая строка таблицы par_ind[] описывает один параметр; структура строки состоит из следующих полей:

- 1- индекс памяти (определен в таблице enum parametri в tabpar.h)
- 2- дескриптор типа (определен в таблице enum limiti_word and enum limiti_byte в tabpar.h)
- 3- значение по умолчанию
- 4- уровень защиты

Первые 40 параметров зарезервированы для конфигурации системы, их изменять не следует (по Вашему усмотрению может быть задано только значение по умолчанию).

В примере ниже, поэтому, некоторые прикладные параметры введены после этих 40 параметров - как PAR1, PAR2 и т.д.

```
const desc_par par_ind[] = {
//***** Параметры системы
COM_IB_ADD      ,L_1_247      ,1          ,P_MAN_3, //000 SY00 ЗАРЕЗЕРВИРОВАН,
COM_IB_BAUD     ,L_0_5         ,3          ,P_MAN_3, //001 SY01 ЗАРЕЗЕРВИРОВАН,
COM_IB_PAR      ,L_2_2         ,2          ,P_MAN_3, //002 SY02 ЗАРЕЗЕРВИРОВАН,
COM_MB_ADD      ,L_1_247      ,1          ,P_MAN_3, //003 SY03 адрес коммуникационного порта RS485,
COM_MB_BAUD     ,L_0_5         ,BAUD_RATE_9600 ,P_MAN_3, //004 SY04 скорость передачи на порту
RS485,[0=2400,1=4800,2=9600,3=19200,4=38400,5=57600,6=76800]
COM_MB_PAR      ,L_0_2         ,EVEN_PARITY  ,P_MAN_3, //005 SY05 четность на порту RS485,[0=БЕЗ ПРОВЕРКИ ЧЕТНОСТИ, 1=НЕЧЕТ,
2=ЧЕТ]
LCD_BACKLIGHT_MODE ,L_0_2         ,LCD_BACKLIGHT_MODE_TIMED ,P_MAN_3, //006 SY06 режим подсветки ЖК-дисплея,[0=РЕЖИМ ВЫКЛ.,1=РЕЖИМ ВКЛ.,2=РЕЖИМ
ОТЛОЖЕН]
LCD_BACKLIGHT_TIME ,L_BYTE        ,60         ,P_MAN_3, //007 SY07 время подсветки ЖК-дисплея,с
LCD_CONTRAST     ,L_BYTE        ,15         ,P_MAN_3, //008 SY08 контрастность ЖК-дисплея,
PASSWORD_USR_1   ,L_SBYTE       ,0          ,P_MAN_3, //009 SY09 Пароль пользователя,
PASSWORD_USR_2   ,L_SBYTE       ,1          ,P_MAN_3, //010 SY10 Пароль на обслуживание,
PASSWORD_MAN_3   ,L_SBYTE       ,2          ,P_MAN_3, //011 SY11 Пароль изготовителя
TYPE_AI1         ,L_0_2         ,AICONF_TYPE_NTC ,P_MAN_3, //012 SY12 тип IN1 [0=NTC,1=4-20mA,2=0-10V]
TYPE_AI2         ,L_0_2         ,AICONF_TYPE_NTC ,P_MAN_3, //013 SY13 тип IN2[0=NTC,1=4-20mA,2=0-10V]
TYPE_AI          ,L_NULL        ,AITEMP_TYPE_NTC ,P_MAN_3, //014 SY14 тип IN3/IN4/IN5/IN9/IN10 [0=NTC]
TYPE_AO1         ,L_0_4         ,AO_TYPE_10V    ,P_MAN_3, //015 SY15 тип AO1 [0=ВЫКЛЮЧЕН,1=ОТКЛЮЧЕНИЕ ФАЗЫ,2=0-10V,3=PWM,4=ЧАСТОТА]
TYPE_AO2         ,L_0_4         ,AO_TYPE_10V    ,P_MAN_3, //016 SY16 тип AO2,[0=ВЫКЛЮЧЕН,1=ОТКЛЮЧЕНИЕ ФАЗЫ,2=0-10V,3=PWM,4=ЧАСТОТА]
TYPE_TK1         ,L_BIT         ,OC_TYPE_DIS    ,P_MAN_3, //017 SY17 тип OC1 [0=ВЫКЛЮЧЕН,1=ОТКЛЮЧЕНИЕ ФАЗЫ]
TYPE_TK2         ,L_BIT         ,0            ,P_MAN_3, //018 SY18 ЗАРЕЗЕРВИРОВАН
PWM_FREQ         ,L_1_200      ,100         ,P_MAN_3, //019 SY19 Частота PWM,Гц*10
PC_TIME_AO1      ,L_BYTE        ,20          ,P_MAN_3, //020 SY20 Время начала отключения фазы AO1,с/10
PC_TIME_AO2      ,L_BYTE        ,20          ,P_MAN_3, //021 SY21 Время начала отключения фазы AO2,с/10
PC_TIME_TK1      ,L_BYTE        ,20          ,P_MAN_3, //022 SY22 Время начала отключения фазы OC1,с/10
PC_TIME_TK2      ,L_BYTE        ,20          ,P_MAN_3, //023 SY23 ЗАРЕЗЕРВИРОВАН,с/10
PC_SHIFT_AO1     ,L_0_10       ,1          ,P_MAN_3, //024 SY24 Фазовый сдвиг при отключении фазы AO1,мс/2
PC_SHIFT_AO2     ,L_0_10       ,1          ,P_MAN_3, //025 SY25 Фазовый сдвиг при отключении фазы AO2,мс/2
PC_SHIFT_TK1     ,L_0_10       ,1          ,P_MAN_3, //026 SY26 Фазовый сдвиг при отключении фазы OC1,мс/2
PC_SHIFT_TK2     ,L_0_10       ,1          ,P_MAN_3, //027 SY27 ЗАРЕЗЕРВИРОВАН,мс/2
FREE_01         ,L_NULL        ,0            ,P_MAN_3, //028 SY28 СВОБОДЕН,
FREE_02         ,L_NULL        ,0            ,P_MAN_3, //029 SY29 СВОБОДЕН,
FREE_03         ,L_NULL        ,0            ,P_MAN_3, //030 SY30 СВОБОДЕН,
FREE_04         ,L_NULL        ,0            ,P_MAN_3, //031 SY31 СВОБОДЕН,
FREE_05         ,L_NULL        ,0            ,P_MAN_3, //032 SY32 СВОБОДЕН,
FREE_06         ,L_NULL        ,0            ,P_MAN_3, //033 SY33 СВОБОДЕН,
FREE_07         ,L_NULL        ,0            ,P_MAN_3, //034 SY34 СВОБОДЕН,
FREE_08         ,L_NULL        ,0            ,P_MAN_3, //035 SY35 СВОБОДЕН,
FREE_10         ,L_NULL        ,0            ,P_MAN_3, //036 SY37 СВОБОДЕН,
FREE_11         ,L_NULL        ,0            ,P_MAN_3, //037 SY38 СВОБОДЕН,
FREE_12         ,L_NULL        ,0            ,P_MAN_3, //038 SY39 СВОБОДЕН,
FREE_13         ,L_NULL        ,0            ,P_MAN_3, //039 SY40 СВОБОДЕН,
//***** Прикладные параметры
PAR1             ,L_WORD        ,0            ,P_USR_1,
PAR2             ,L_BYTE        ,0            ,P_USR_1,
PAR3             ,L_SBYTE       ,0            ,P_USR_1,
PAR4             ,L_NIB         ,0            ,P_USR_1,
PAR5             ,L_BIT         ,0            ,P_USR_1,
PAR11            ,L_PAR11       ,10          ,P_USR_1,
PAR12            ,L_PAR12       ,50          ,P_USR_1,
PAR13            ,L_PAR13       ,30          ,P_USR_1,};
```

6.5.2 Таблицы дескриптора par_lim_word[] и par_lim_byte[]

par_lim_word[] - это таблица пределов 16-разрядных параметров, тогда как par_lim_byte[]- таблица пределов 8-разрядных параметров, без знака и со знаком, полубайтов, групп разрядов и одиночных разрядов. В каждой строке таблицы присутствует структура дескриптора со следующими полями:

- 1- минимальный предел
- 2- максимальный предел
- 3- атрибут; на данный момент доступны следующие опции:

```
#define MDEC      0x01    // представление десятичных данных (0.1)
#define MRL      0x02    // нижний предел в оперативной памяти
#define MRH      0x04    // верхний предел в оперативной памяти
#define SCHAR    0x08    // 8 разрядов со знаком
```

Если необходимо, в gestpar.h можно добавить другие опции.

Например, прикладной параметр PAR13 ограничен снизу параметром PAR11 и сверху - параметром PAR12.

```

const limiti_16 par_lim_word[] = {
{ 0 ,65535 ,0 }, // L_WORD ,
{ -32768 ,32767 ,0 }, // L_INT ,
{ -32768 ,PAR12 ,MRH }, // L_PAR11 ,// EXAMPLE
{ PAR11 ,32767 ,MRL }, // L_PAR12 ,// EXAMPLE
{ PAR11 ,PAR12 ,MRL+MRH }, // L_PAR13 ,// EXAMPLE
};

const limiti_8 par_lim_byte[] = {
// MIN ,MAX ,ATTR
{ 0 ,0 ,0 }, //L_NULL ,
{ 0 ,1 ,0 }, //L_BIT ,
{ 0 ,3 ,0 }, //L_NIB ,
{ 0 ,255 ,0 }, //L_BYTE ,
{ -127 ,127 ,SCHAR }, //L_SBYTE ,
{ 0 ,2 ,0 }, //L_0_2 ,
{ 2 ,2 ,0 }, //L_2_2 ,
{ 0 ,4 ,0 }, //L_0_4 ,
{ 0 ,5 ,0 }, //L_0_5 ,
{ 0 ,10 ,0 }, //L_0_10 ,
{ 1 ,200 ,0 }, //L_1_200 ,
{ 1 ,247 ,0 }, //L_1_247 ,
};

```

6.5.3 Процедура добавления нового параметра

1. Вставьте новую строку в par_ind[]


```

...
PAR13 ,L_PAR13 ,30 ,P_USR_1,

```
2. Вставьте новый индекс (PAR13) в счетчик всех параметров - не забудьте поставить его на правильном месте - после желаемого типа.
3. Добавьте индекс этого нового параметра в заголовок файла tabpar.h: `#define IND_PAR13 47`
4. Если требуется, вставьте новый раздел (L_PAR13) в счетчик предела limiti_word или limiti_byte
5. Если требуется, вставьте описание нового раздела L_PAR13 в par_lim_word или par_lim_byte

7 ВХОДЫ И ВЫХОДЫ

7.1 Входы

Все 10 входов можно сконфигурировать, как цифровые входы. Как аналоговые входы, можно сконфигурировать только входы IN3, IN4, IN5, IN8, IN9 и IN10.

Входы IN1 и IN2 конфигурируются с помощью параметров TYPE_AI1 и TYPE_AI2 в соответствии с таблицей ниже, в которой остальные аналоговые входы IN3 – IN7 используются для температуры.

Таблица 1

ЗНАЧЕНИЕ ПАРАМЕТРА TYPE_AI1 и TYPE_AI2	ОПИСАНИЕ
0	NTC / Цифровой вход
1	4-20 мА
2	0-10 В

Датчики (преобразователи) входов IN3, IN4, IN5, IN8, IN9 и IN10 рекомендуется использовать при высоких температурах, так как они более точны в экстремумах диапазона измерений.

Измерение температуры проводится с точностью до десятых долей °C (один верный знак после запятой); ток и напряжение измеряются с точностью до сотых долей мА или В (два верных знака после запятой).

7.2 ВЫХОДЫ

7.2.1 Цифровые выходы

И аналоговые, и выходы ОС могут быть сконфигурированы, как цифровые выходы, если они отключены конфигурационными параметрами TYPE_A01, TYPE_A02, TYPE_TK1. Имеется 4 релейных цифровых выходов.

7.2.2 Аналоговые выходы

Имеется 2 аналоговых выхода, AO1 and AO2. Параметры TYPE_A01 и TYPE_A02 определяют тип выхода: 0-10 В, отключение фазы, PWM, частота или выключен (используется, как DO).

Таблица 4

ЗНАЧЕНИЯ ПАРАМЕТРОВ TYPE_A01 и TYPE_A02	ОПИСАНИЕ
0	Выключен (или DO)
1	Отключение фазы
2	0-10 В
3	PWM
4	Частота

Выбор типа выхода влияет на единицы измерения. Выходы частоты имеют размерность Гц (0-255); на выходах отключения фазы, мощности и PWM используется размерность "%".

Вследствие аппаратных ограничений, конфигурирование одного выхода 0-10 В, а остальных - как PWM или частоты, невозможно. Допустимые (O) и запрещенные (X) конфигурации перечислены в таблице ниже.

	TYPE_A01 = 0 (ОТКЛЮЧЕН)	TYPE_A01 = 1 (ОТКЛЮЧЕНИЕ ФАЗЫ)	TYPE_A01 = 2 (0 – 10 В)	TYPE_A01 = 3 (PWM)	TYPE_A01 = 4 (Частота)
TYPE_A02 = 0 (ОТКЛЮЧЕН)	O	O	O	O	O
TYPE_A02 = 1 (ОТКЛЮЧЕНИЕ ФАЗЫ)	O	O	O	O	O
TYPE_A02 = 2 (0 – 10 В)	O	O	O	X	X
TYPE_A02 = 3 (PWM)	O	O	X	O	X
TYPE_A02 = 4 (Частота)	O	O	X	X	X

7.2.2.1 Тип отключения фазы

На выходе синхронно, с привязкой к точке "пересечения нуля" сети генерируется импульс длительностью 500 мкс. Задержка импульса относительно точки "пересечения нуля" рассчитывается таким образом, чтобы заданное значение соответствовало эффективному напряжению на нагрузке: до 20% выход всегда выключен, выше 90% выход всегда включен.

Параметры *PC_TIME_A01*, *PC_TIME_A02* и *PC_TIME_TK1* позволяют задать время запуска, которое будет применено на выходе; оно будет определять включение выхода и переход его от 0% до другого значения: в течение этого времени выход принудительно устанавливается на 100%.

Параметры *PC_SHIFT_A01*, *PC_SHIFT_A02* и *PC_SHIFT_A01* позволяют задать сдвиг для адаптации к нагрузке.

7.2.2.2 Тип 0-10 В

Напряжение на выходе меняется в соответствии с заданным значением: 0% выход всегда выключен, 100% на выходе 10 В.

7.2.2.3 Тип PWM

На выходе генерируется сигнал с постоянной частотой и изменяемым циклом занятости.

Частота на выходе сконфигурированном, как PWM, определяется параметром *PWM_FREQ*.

Цикл занятости меняется в соответствии с заданным значением: 0% выход всегда выключен, 100% - выход всегда включен.

7.2.2.4 Тип "частота"

На выходе генерируется сигнал с изменяемой частотой и фиксированным циклом занятости.

Частота на выходе меняется в соответствии с заданным значением: ниже 10 Гц выход всегда выключен. Максимальная частота 255 Гц. Цикл занятости всегда равен 50%.

7.2.3 Выход с открытым коллектором

Выход с открытым коллектором может быть сконфигурирован с параметрами *TYPE_TK1* и если включен - управляет отключением фазы (см. описание выше).

Если решено отключить выход ОС, то его можно использовать, как цифровой выход DO.

На выходах ОС в качестве единицы измерения используется %.

7.3 Порт последовательного интерфейса с протоколом MODBUS

Порт последовательного интерфейса RS485 использует протокол подчиненного устройства Modbus RTU с параметрами COM_MB_ADD (по умолчанию 1), COM_MB_BAUD (по умолчанию 9600), COM_MB_PAR (по умолчанию EVEN).

Адреса между 0 и 0x7FFF свободны, они могут быть сконфигурированы пользователем.

Для примера ниже представлено содержимое регистров:

```
#define MB_ADDR_INFO      0x0000
#define MB_NUM_INFO      6
#define MB_ADDR_DI       0x0100
#define MB_NUM_DI        1
#define MB_ADDR_DI_PULSE 0x0101
#define MB_NUM_DI_PULSE  2
#define MB_ADDR_DO       0x0180
#define MB_NUM_DO        1
#define MB_ADDR_AI       0x2000
#define MB_NUM_AI        7
#define MB_ADDR_AO       0x0280
#define MB_NUM_AO        4
#define MB_ADDR_RTC      0x0500
#define MB_NUM_RTC       6
#define MB_ADDR_PARAMETERS 0x601
#define MB_NUM_PARAMETERS (LAST_FLAG+1)
#define MB_ADDR_PARVIS   0x1001
#define MB_NUM_PARVIS    MB_NUM_PARAMETERS
```

```
int16_t ReadMBUser(uint16_t Addr)
{
if (Addr == MB_ADDR_INFO + 0) {return ReadInfoProject(INFO_PRJNUM);}
else if (Addr == MB_ADDR_INFO + 1) { return ReadInfoProject( INFO_PRJVER_PARVER);}
else if (Addr == MB_ADDR_INFO + 2) { return ReadInfoProject(INFO_LIBVERS);}
else if (Addr == MB_ADDR_INFO + 3) { return ReadInfoProject(INFO_LIBID);}
else if (Addr == MB_ADDR_INFO + 4) { return ReadInfoProject(INFO_LIBVAR_LIBREV);}
else if ((Addr >= MB_ADDR_DI ) && (Addr < MB_ADDR_DI + MB_NUM_DI )) {return ReadDIValuePacket( ) ;}
else if ((Addr >= MB_ADDR_DI_PULSE ) && (Addr < MB_ADDR_DI_PULSE + MB_NUM_DI_PULSE )) {return ReadDIpulseValue(Addr - MB_ADDR_DI_PULSE);}
else if ((Addr >= MB_ADDR_DO ) && (Addr < MB_ADDR_DO + MB_NUM_DO )) {return ReadDOValuePacket( ) ;}
else if ((Addr >= MB_ADDR_AI ) && (Addr < MB_ADDR_AI + MB_NUM_AI )) {return ReadAIValue(Addr - MB_ADDR_AI) ;}
else if ((Addr >= MB_ADDR_AO ) && (Addr < MB_ADDR_AO + MB_NUM_AO )) {return ReadAOValue(Addr - MB_ADDR_AO) ;}
else if ((Addr >= MB_ADDR_RTC ) && (Addr < MB_ADDR_RTC + MB_NUM_RTC )) {return ReadRTCValue(Addr - MB_ADDR_RTC) ;}
else if ((Addr == MB_ADDR_PARAMETERS-1)) {return MB_NUM_PARAMETERS ;}
else if ((Addr >= MB_ADDR_PARAMETERS) && (Addr < MB_ADDR_PARAMETERS + MB_NUM_PARAMETERS)) {return ReadPARValue(Addr - MB_ADDR_PARAMETERS) ;}
else if ((Addr >= MB_ADDR_PARVIS ) && (Addr < MB_ADDR_PARVIS + MB_NUM_PARVIS )) {return ReadPARVisib(Addr - MB_ADDR_PARVIS) ;}
else return 0;
}

void WriteMBUser(uint16_t Addr, int16_t valueWrite)
{
if ((Addr >= MB_ADDR_PARAMETERS) && (Addr < MB_ADDR_PARAMETERS + MB_NUM_PARAMETERS)) {WritePARValue(Addr - MB_ADDR_PARAMETERS, valueWrite);}
else if ((Addr >= MB_ADDR_PARVIS ) && (Addr < MB_ADDR_PARVIS + MB_NUM_PARVIS )) {WritePARVisib(Addr - MB_ADDR_PARVIS , valueWrite);}
}
}
```

Адреса выше 0xF000 зарезервированы для тестирования и, поэтому, недоступны пользователю.

Соединения последовательного интерфейса RS485 можно использовать для связи с системой-супервизором (персональным компьютером (ПК)). Только по протоколу Modbus для подчиненных устройств.

FC 03 : Регистр удержания, множественное считывание;

FC 04 : Входной регистр, множественное считывание;

FC 06 : Регистр удержания, одна запись;

FC 16 : Регистр удержания, множественная запись;

8 ПРИНАДЛЕЖНОСТИ

8.1 EVIF20SUXI -последовательный интерфейс RS-485/USB без оптронной развязки

8.1.1 Предварительная информация

Данный интерфейс обеспечивает работу прибора EVD с программным обеспечением настройки Parameters Manager (Менеджер параметров).



8.2 CJAV - набор для подключения

8.2.1 Предварительная информация

Данные наборы обеспечивают кабельное подключение к прибору EVD.

КОНТРОЛЛЕР	КОД ДЛЯ ЗАКАЗА
EVD	CJAV38



9 ТЕХНИЧЕСКИЕ ХАРАКТЕРИСТИКИ

9.1 Технические характеристики

Назначение прибора управления	Контроллер функций.
Конструкция прибора управления	Встраиваемый электронный прибор.
Корпус	Серого цвета, из самогасящегося материала.
Категории устойчивости к нагреванию, пожароопасность	D.
Размеры	71,0 x 110,0 x 60,0 мм (2,795 x 4,330 x 2,362 дюйма; Ш x В x Г); 4 DIN-модуля.
Способ монтажа прибора управления	При установке на DIN-рейке: 35,0 x 7,5 мм (1,377 x 0,295 дюйма); при установке на панели: 35,0 x 15,0 мм (1,377 x 0,590 дюйма).
Класс фронтальной защиты:	IP40.

Соединения	<p>Разъем Micro-Fit (аналоговые входы, цифровые входы, аналоговые выходы и цифровой выход с открытым коллектором);</p> <ul style="list-style-type: none"> - Съемные блоки клемм на винтах (электропитание, цифровые выходы на электромеханические реле, коммуникационные порты). <hr/> <p>Максимальная длина соединительных кабелей:</p> <ul style="list-style-type: none"> - электропитание: 10 м (32,8 фута); - Аналоговые входы: 10 м (32,8 фута); - Блок питания для датчиков аналоговых входов 4-20мА: 10 м (32,8 фута); - Цифровые входы: 10 м (32,8 фута); - Аналоговые выходы 0-10 В: 10 м (32,8 фута); - Аналоговые выходы отключения фазы: 10 м (32,8 фута); - Аналоговые выходы PWM: 1 м (3,28 фута); - Цифровые выходы на электромеханическое реле: 10 м (32,8 фута); - Цифровые выходы с открытым коллектором: 10 м (32,8 фута); - Порты INTRABUS с питанием: 10 м (32,8 фута); <p>Порты подчиненного устройства RS-485 шины MODBUS: 1000 м (3280 футов); см. также руководства по спецификациям и способам реализации шины MODBUS , доступные на сайте http://www.modbus.org/specs.php</p> <p>Сечение жил используемых кабелей должно соответствовать протекающим по ним токам. Для кабельного подключения прибора EVD рекомендуется использовать набор для подключения CJAV38 (поставляется по заказу).</p>
Температура окружающей среды при эксплуатации:	от -10 до 55 °C (от 14 до 131 °F).
Температура окружающей среды при хранении:	от -25 до 70 °C (от -13 до 158 °F).
Относительная влажность при эксплуатации:	от 10 до 90%, без конденсата;

Уровень загрязнения окружающей среды прибором управления	2.
Высота над уровнем моря при эксплуатации	от 0 до 2 000 м (от 0 до 6591 футов).
Высота над уровнем моря при транспортировке	от 0 до 3048 м (от 0 до 10000 футов).
Соответствие стандартам защиты окружающей среды	<ul style="list-style-type: none"> - RoHS 2011/65/EC; - WEEE 2012/19/EU; - REACH (EC) Regulation 1907/2006.
Соответствие стандартам электромагнитной совместимости (ЭМС)	<ul style="list-style-type: none"> - EN 60730-1; - IEC 60730-1.
Электропитание	<p>115... 230 Vac (+10% -15%), 50/60 Гц, 6 VA макс., изолированное.</p> <p>Блок питания должен быть защищен плавким предохранителем с номиналом 2 А-Т 250 В.</p>
Номинальное напряжение импульса:	4 кВ.

Категория перенапряжения	II.
Класс и структура программного обеспечения	A.
Часы	<p>По заказу (с вторичным электропитанием от литиевой аккумуляторной батареи) Время автономной работы от батареи в отсутствие электропитания от сети: ≤ 6 месяцев при 25 °C (77 °F). Время зарядки аккумуляторной батареи: 24 ч. (аккумуляторная батарея заряжается от сетевого блока питания прибора). Дрейф: ≤ 60 сек./месяц при 25 °C (77 °F).</p>
Аналоговые входы	<p>7 входов:</p> <ul style="list-style-type: none"> - 5 для датчиков NTC / Pt1000 - 2 могут быть сконфигурированы с помощью параметра для подключения датчиков или преобразователей NTC 4-20 мА, 0-10 В;
	<p><u>Аналоговые входы NTC (10 КВтΩ @ 25 °C, 77 °F)</u> Тип датчика: В3435. Диапазон измерения: от -50 до 150 °C (от -58 до 248 °F). Разрешающая способность: 0,1 °C). Погрешность: 0,5 °C от -20 до 40 °C, 1 °C от -40 до 120 °C, 2 °C от -50 до 150 °C. Защита: Нет.</p>
	<p><u>Аналоговые входы Pt1000 (Зарезервировано)</u> Тип датчика: Диапазон измерения: от -60 до 200 °C Погрешность: 1,0 °C Разрешающая способность: 0,1 °C). Защита: нет.</p>
<p><u>Аналоговые входы 4-20 мА</u> Входное сопротивление: ≤ 200 Ω. Разрешающая способность: 0,02 мА Защита: нет; максимальный допустимый ток на каждом входе 25 мА.</p>	

	<p>Аналоговые входы 0-10 В</p> <p>Входное сопротивление: $\leq 200 \Omega$.</p> <p>Диапазон измерения: от 0,00 до 12,00 В</p> <p>Разрешающая способность: 0,02 мА</p> <p>Защита: нет; максимальный допустимый ток на каждом входе 12,5 В.</p>
Цифровые входы	<p>5 входов для переключаемых контактов.</p>
	<p>Цифровые входы для переключаемых контактов (5 В пост. тока, 1,5 мА)</p> <p>Блок питания: нет.</p> <p>Входное сопротивление: .</p> <p>Защита: Нет.</p> <p>Минимальное время включения для быстрых входов при обнаружении импульса: 2,5 мс.</p>
Аналоговые выходы	<p>2 выхода, которые могут быть сконфигурированы параметром, как 0-10 В, с отключением фазы или PWM.</p>
	<p><u>Аналоговые выходы 0-10 В (макс. 10 мА)</u></p> <p>Входное сопротивление: 1 КΩ.</p> <p>Разрешающая способность: 0,01 В.</p> <p>Защита: нет.</p>
	<p><u>Аналоговые выходы отключения фазы:</u></p> <p>Выход: 10 Vdc, 10 мА макс.</p> <p>Частота: синхронна с частотой, используемой в блоке питания.</p> <p>Длительность импульса: 500 мкс</p> <p>Сдвиг 20... 90 %.</p> <p>Защита: нет.</p>
	<p><u>Аналоговые выходы PWM</u></p> <p>Выход: 10 Vdc, 10 мА макс.</p> <p>Частота: 10... 2 кГц.</p> <p>Режим ожидания: 5... 95 %.</p> <p>Защита: нет.</p>
<p><u>Аналоговые выходы частоты</u></p> <p>Выход: 10 Vdc, 10 мА макс.</p> <p>Частота: 10 Гц ... 255 Гц.</p> <p>Готовность: 50%</p> <p>Защита: нет.</p>	

Цифровые выходы	До 5 выходов: <ul style="list-style-type: none">- 2 релейных SPST выхода, 3 A @ 250 Vac- 1 релейный SPST выход, 8 A @ 250 Vac- 1 релейный SPST выход, 12 A @ 250 Vac- 1 выход с открытым коллектором, 12 Vdc, 40 mA макс.
Операции типа 1 или типа 2	Тип 1.
Дополнительные функции операций типа 1 или типа 2:	С.
Дисплеи	Жидкокристаллический дисплей Светодиодные индикаторы.
Коммуникационные порты	До 2 портов: <ul style="list-style-type: none">- 1 порт INTRABUS с питанием;- 1 порт подчиненного устройства RS-485 шины MODBUS
Динамик для подачи сигналов тревоги	Встроенный.

EVD LCD
Контроллеры, программируемые на языке C
- SS 20160208 -
Руководство по эксплуатации, ред. 0.03

Данный документ является собственностью компании EVCO; любое его раскрытие и репродуцирование категорически запрещено без получения явно выраженного разрешения на это от компании EVCO.

EVCO не несет никакой ответственности за технические характеристики и функции, описанные в данном документе, а также за любые ошибки, которые он может содержать, а также за ошибки, которые могут возникнуть при использовании данного документа.

EVCO не несет никакой ответственности за любые ущербы, обусловленные несоблюдением инструкций, приведенных в данном документе.

EVCO оставляет за собой право в любое время вносить любые изменения в данный документ, не влияющие на основной функционал прибора и безопасность работы с ним, без предварительного уведомления об этом.



Компания EVCO S.p.A.

Via Feltre 81, 32036 Sedico Belluno, ИТАЛИЯ

Тел. 0437 / 8422

Факс. 0437 / 83648

info@evco.it

www.evco.it