



Программное обеспечение для контроллеров ACU-PROG-PLC

ООО ИВЦ Технологика

Краткое руководство пользователя

АННОТАЦИЯ

В этом руководстве приведено описание библиотеки «Типовых функциональных блоков» ООО «ИВЦ Технологика»:

1. Руководство содержит описание функциональных блоков, их функциональные возможности и перечень переменных.
2. Руководство предназначено для пользователей, имеющих базовые знания по технологиям промышленной автоматизации.
3. Руководство разделено на разделы. Каждый раздел содержит описание определенного функционального блока.

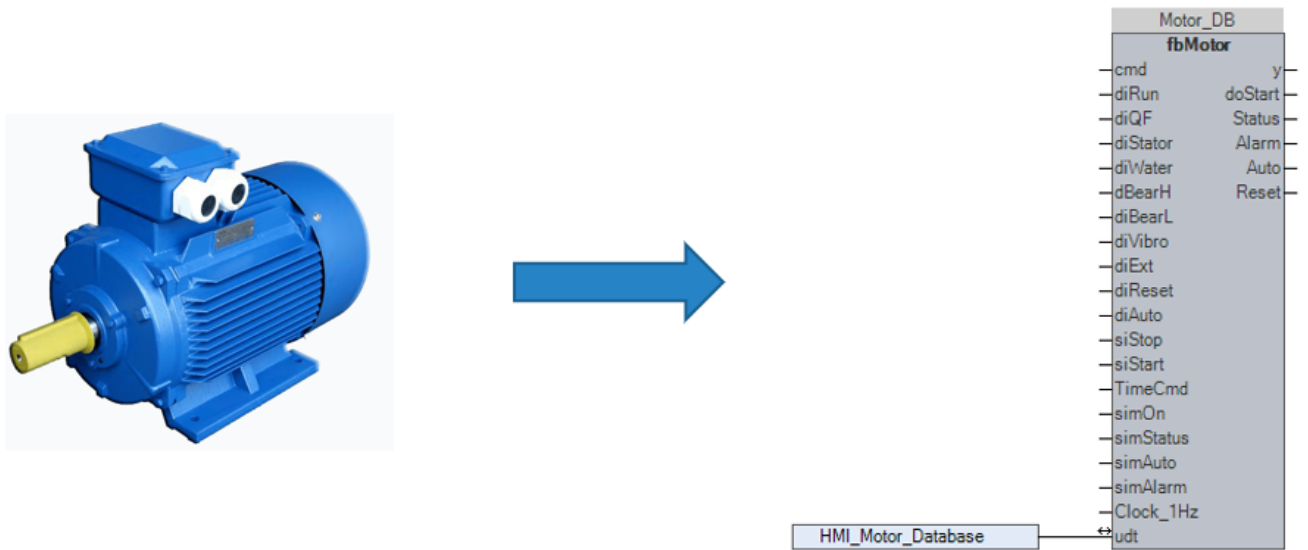
СОДЕРЖАНИЕ

1. Назначение	4
2. Принцип работы	5
3. Условия работы программы.....	6
4. Инструкция по установке.....	7
5. Функциональные возможности	8
5.1. AI	8
5.2. DI	10
5.3. Motor.....	12
5.1. SolenoidValve.....	14
5.1. Valve	16

1. НАЗНАЧЕНИЕ

Типовой функциональный блок — это отлаженный фрагмент кода, работающий согласно информационной модели конкретного технологического оборудования (насоса, задвижки, клапана, дискретного и аналогового датчика).

2. ПРИНЦИП РАБОТЫ



ACU-Prog-PLC состоит из программных модулей - функциональных блоков (FB).

Функциональный блок (FB) – это подпрограмма, работающая согласно информационной модели конкретного технологического оборудования (преобразователя частоты, насоса, задвижки, клапана, дискретного и аналогового датчика, нагревателя, конвейера и др.).

Например, FB fbMotor — это функциональный блок, который реализует управление электродвигателем с прямым пуском.

3. УСЛОВИЯ РАБОТЫ ПРОГРАММЫ

Для использования ACU-PROG_PLC требуется любой программируемый логический контроллер с поддержкой языков программирования ST (Structured Text) и FBD (Function Block Diagram) стандарта ГОСТ Р МЭК 61131-3-2016.

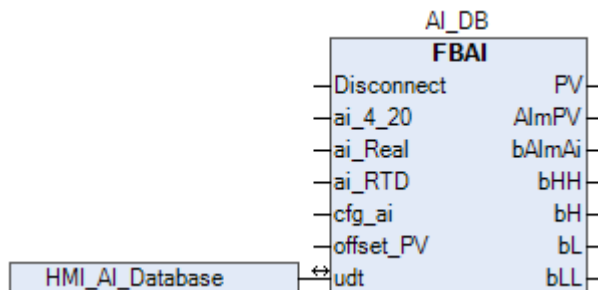
4. ИНСТРУКЦИЯ ПО УСТАНОВКЕ

Последовательность действий:

1. Откройте среду разработки (IDE) вашего ПЛК (ПЛК - программируемый логический контроллер);
2. Импортируйте функциональные блоки и свяжите их с входами/выходами ПЛК в соответствии с требованиями вашего проекта;
3. Подключитесь по поддерживаемому сетевому интерфейсу к ПЛК;
4. Загрузите программу в ПЛК.

5. ФУНКЦИОНАЛЬНЫЕ ВОЗМОЖНОСТИ

5.1. AI



Назначение. Преобразование промышленных нормированных сигналов, полученных с датчиков и вторичных преобразователей, в инженерные единицы (температура - °С, давление – Па и т.п.).

Функциональные возможности:

- Диагностика состояния линии соединения с датчиком
- Контроль выхода показаний за шкалу датчика
- Масштабирование аналогового сигнала в инженерные единицы (5 Бар, 100 °С и т.п.)
- Контроль срабатывания уставок НН, Н, L, LL
- Режим симуляции

Переменные:

Входные	Тип	Описание
Disconnect	Bool	1=Потеря соединения с источником ai_Real (Например, ведомое Modbus устройство)
ai_4_20	Int	Сигнал с поля 4..20 mA (код АЦП -4864...32767)
ai_Real	Real	Значение процесса. 0-100% или инженерные единицы 100 МПа, 140 мм и т.п.
ai_RTD	DINT	Сигнал с поля термосопротивление (520=52,0 °С)
cfg_ai	INT	0 = ai_4_20, 1 = ai_Real, 2 = ai_RTD, 3 = ai_PTC
offset_PV	Real	Смещение HMI_PV

Выходные	Тип	Описание
PV	ANALOG_TYPE	Параметр процесса (структура)
AlmPV	Int	Состояние процесса: 0=Норма, 1=LL, 2=L, 3=H, 4=HH, 5=Обрыв датчика

Выходные	Тип	Описание
bAlmAi	Bool	Состояние линии датчика: 0=Норма, 1=Авария
bNH	Bool	Срабатывание верхней аварийной уставки
bH	Bool	Срабатывание верхней предупредительной уставки
bL	Bool	Срабатывание нижней предупредительной уставки
bLL	Bool	Срабатывание нижней аварийной уставки

Входные- выходные	Тип	Описание
udt	udtAI	Для передачи информации в Прикладное программное обеспечение ACU-Prog-HMI (структура)

Пользовательские типы данных:

ANALOG_TYPE
<p>STRUCT</p> <p>PV : Real; // Параметр процесса в инженерных единицах (5 Бар, 100 °C).</p> <p>ScaleLo : Real; // Нижний предел шкалы PV</p> <p>ScaleHi : Real; // Верхний предел шкалы PV</p> <p>Units : String := '°C'; // Единица измерения PV</p> <p>Format : String := '9999'; // Формат PV</p> <p>AlmAi : Bool; // Состояние линии датчика: 0=Норма, 1=Авария</p> <p>END_STRUCT</p>

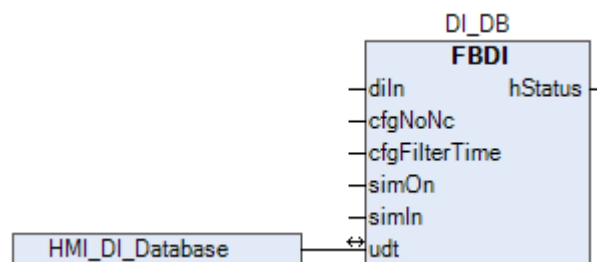
udtAI**STRUCT**

```

simOn : Bool; // 1 = Включить режим симуляции
simIn : Real; // Значение симуляции. Задается в диапазоне шкалы датчика
hScaleLo : Real := 0.0; // Нижний предел шкалы датчика
hScaleHi : Real := 100.0; // Верхний предел шкалы датчика
hDB : Real := 2.5; // Гистерезис. Формат шкалы
hHH : Real := 99999.0; // Верхняя аварийная уставка. Формат шкалы
hH : Real := 99999.0; // Верхняя предупредительная уставка. Формат шкалы
hL : Real := -99999.0; // Нижняя предупредительная уставка. Формат шкалы
hLL : Real := -99999.0; // Нижняя аварийная уставка. Формат шкалы
hPV : Real; // Значение процесса. Формат шкалы
hmA : Real; // Ток, мА
hAlmPV : Int; // Состояние процесса: 0 = Норма, 1 = LL, 2 = L, 3 = H, 4 = HH
hAlmAi : Bool; // Состояние линии датчика: 0 = Норма, 1 = Авария
hUnits : String := '°C';
hFormat : String := '9999';

```

END_STRUCT

5.2. DI

Назначение: Фильтрация входного дискретного сигнала.

Функциональные возможности:

- Фильтрация от дребезга (многократные неконтролируемые замыкания и размыкания контакта)
- Выбор типа дискретного сигнала NO/NC
- Режим симуляции.

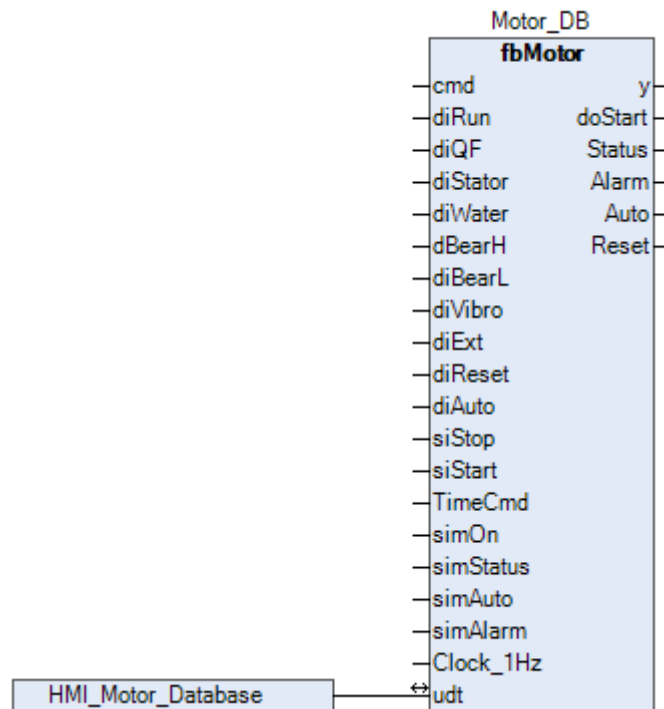
Вход	Тип	Описание
diIn	Bool	Вход полевого сигнала
cfgNoNc	Bool	Тип дискретного сигнала: 0 = нормально открытый NO, 1 = нормально закрытый (NC)
cfgFilterTime	TIME	Время фильтрации
simOn	Bool	0 = Рабочий режим, 1 = Симуляция
simIn	Bool	Вход симуляции: 0 = Разомкнут, 1 = Замкнут

Выход	Тип	Описание
hStatus	Bool	0 = Разомкнут, 1 = Замкнут

Входные- выходные	Тип	Описание
udt	udtDI	Для передачи информации в Прикладное программное обеспечение АСУ-Prog-НМІ

Пользовательские типы данных:

udtDI
STRUCT hStatus :BOOL; // 0 = Разомкнут, 1 = Замкнут hStatusINT :INT; // 0 = Разомкнут, 1 = Замкнут END_STRUCT

5.3. Motor

Назначение. Управление электродвигателем с прямым пуском (так же можно применить для других устройств с состояниями Отключен/Работает. Например, нагреватель).

Функциональные возможности.

- Контроль состояния электродвигателя (Отключен/Работает, Норма/Авария)
- Индикация и смена режимов: местный, автоматический и дистанционный
- Подача команд управления с учетом активного режима управления
- Автоматическое отключение электродвигателя в случае аварийного события (перегрев, протечка и т.п.)
- Контроль выполнения команды по сигналу обратной связи
- Приоритетное управление по сигналам технологических блокировок
- Режим симуляции

Вход	Тип	Описание
cmd	Bool	Команды в Авто: 0=Стоп, 1=Пуск

Вход	Тип	Описание
diRun	Bool	1=Работает, 0=Остановлен
diQF	Bool	1=Автомат защиты двигателя в норме
diStator	Bool	1=Температура статора в норме
diWater	Bool	1=Протечка воды отсутствует
dBearH	Bool	1=Температура верхнего подшипника в норме
diBearL	Bool	1=Температура нижнего подшипника в норме
diVibro	Bool	1=Вибрация в норме
diExt	Bool	1=Другая авария
diReset	Bool	Сброс аварии с кнопки ША
diAuto	Bool	Режим Авто. Переключатель на двери
siStop	Bool	Блокировка стоп. Приоритет над siStart
siStart	Bool	Блокировка пуска
TimeCmd	TIME	Время контроля выполнения команды (udtMotor.hAlmCmd)
simOn	Bool	Включить симуляцию
simStatus	Bool	Симуляция: 1=Работает, 0=Остановлен
simAuto	Bool	Симуляция автоматического режима
simAlarm	Bool	Симуляция аварии
Clock_1Hz	Bool	Системная переменная

Выходные	Тип	Описание
y	Int	Состояние автомата: 0=Отключен, 1=Работает, 3=Не готов
doStart	Bool	Команда на контактор Пуск
Status	Bool	Состояние: 0=Отключен, 1=Работает
Alarm	Bool	1=Общая авария
Auto	Bool	Активен режим Авто
Reset	Bool	Сброс аварий для внешних систем. Импульс

Входные- выходные	Тип	Описание
udt	udtMotor	Для передачи информации в Прикладное программное обеспечение АСУ-Prog-НМІ

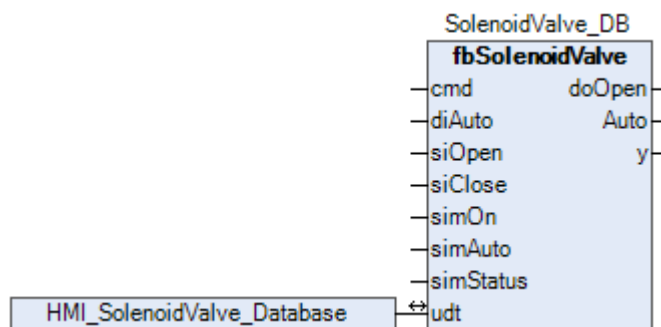
Пользовательские типы данных:**udtMotor**

STRUCT

```

hRemote : BOOL; // Дистанционный режим управления с НМІ (При условии diAuto=1)
  hStart : Bool; // Команда в дист Пуск: Импульс
  hStop : Bool; // Команда в дист Стоп: Импульс
  hReset : Bool; // Сброс аварии с НМІ: Импульс
  hStatus : Int; // Состояние: 0=Отключен,1=Работает
  hSi : Bool; // Активна блокировка
  hMode : Int; // Режим управления: 0=Местный, 1=Автоматический, 2=Дистанционный
  hAlmCom : Bool; // Общая авария
  hAlmCmd : Bool; // Невыполнение команды
  hAlmQF : Bool; // Сработал автомат защиты двигателя
  hAlmStator : Bool; // Перегрев статора
  hAlmWater : Bool; // Протечка
  hAlmBearH : Bool; // Перегрев верхнего подшипника
  hAlmBearL : Bool; // Перегрев нижнего подшипника
  hAlmVibro : Bool; // Вибрация
  hAlmExt : Bool; // 1=Другая авария
  hRunTime : DInt; // Время наработки, ч
  hResetRunTime : Bool; // Сброс наработки. Удерживайте 15 с
END_STRUCT

```

5.1. SolenoidValve

Назначение. Управление двухпозиционным электромагнитным(соленоидным) клапаном.

Функциональные возможности:

- Контроль состояния двухпозиционного клапана (Открыт/Закрыт)
- Индикация и смена режимов: местный, автоматический и дистанционный

- Подача команд управления с учетом активного режима управления
- Приоритетное управление по сигналам технологических блокировок
- Режим симуляции

Вход	Тип	Описание
cmd	Bool	Команды в Авто : 0=Закрывать, 1=Открыть
diAuto	Bool	Режим Авто. Переключатель на двери ША
siOpen	Bool	Блокировка Открыть.
siClose	Bool	Блокировка Закрывать. Приоритет над siOpen
simOn	Bool	Режим симуляции
simAuto	Bool	Симуляция автоматического режима
simStatus	Bool	Симуляция: 0=Закрывает, 1=Открыт

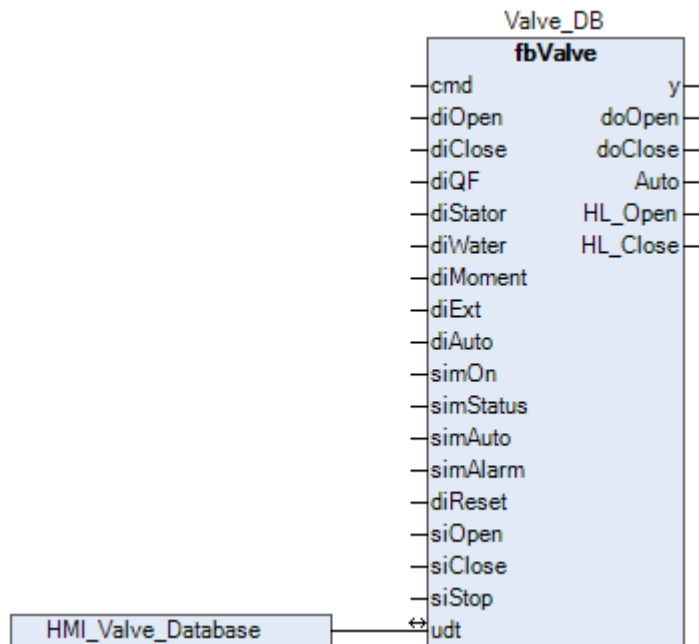
Выходные	Тип	Описание
doOpen	Bool	Команда на контактор: 1 = Открыть, 0 = Закрывать
Auto	Bool	Активен режим Авто
y	Int	Состояние автомата: 0=Закрывает, 1=Открыт

Входные- выходные	Тип	Описание
udt	udtSolenoidValv e	Для передачи информации в Прикладное программное обеспечение АСУ-Prog-HMI

Пользовательские типы данных:

udtSolenoidValve
<pre> STRUCT hRemote : BOOL := 0; // Дистанционный режим управления с HMI (При условии di_Auto=1) hOpen : BOOL := 0; // Команда в дист Открыть: Импульс hClose : BOOL := 0; // Команда в дист Закрывать: Импульс hStatus : BOOL := 0; // bool Состояние: 0=Закрывает, 1=Открыт hMode : INT; // Режим управления: 0=Местный, 1=Автоматический, 2=Дистанционный hStatusINT : INT; // int Состояние: 0=Закрывает, 1=Открыт hSi : BOOL; END_STRUCT </pre>

5.1. Valve



Назначение. Управление задвижкой с концевыми выключателями

Функциональные возможности:

- Контроль состояния задвижки (Открыта/промежуточное положение/Закрыта/Неопределённое положение, Норма/Авария)
- Индикация и смена режимов: местный, автоматический и дистанционный
- Подача команд управления с учетом активного режима управления
- Автоматическое отключение задвижки в случае аварийного события (срабатывание муфты момента, превышение времени хода и т.п.)
- Контроль выполнения команды по сигналу обратной связи
- Приоритетное управление по сигналам технологических блокировок
- Режим симуляции

Вход	Тип	Описание
cmd	BOOL	Команды в Авто: 0=Стоп, 1=Пуск
diOpen	BOOL	Открыт (концевой выключатель)
diClose	BOOL	Закрыт (концевой выключатель)
diQF	BOOL	1=Автомат защиты двигателя в норме
diStator	BOOL	1=Температура статора в норме
diWater	BOOL	1=Протечка воды отсутствует
diMoment	BOOL	1=Момент в норме
diExt	BOOL	1=Другая авария

Вход	Тип	Описание
diAuto	BOOL	Режим Авто. Переключатель на двери
simOn	BOOL	Включить симуляцию
simStatus	BOOL	0=Закрыт, 1=Открыт
simAuto	BOOL	Симуляция автоматического режима
simAlarm	BOOL	Симуляция аварии
diReset	BOOL	Сброс аварии с кнопки ША
siOpen	BOOL	Блокировка в режиме "Открыто"
siClose	BOOL	Блокировка в режиме "Закрыто"
siStop	BOOL	Блокировка в режиме "Стоп"

Выходные	Тип	Описание
y	Int	Состояние автомата: 0=Закрыт, 1=Открыт, 2=Промежуточное, 3=Неопределенное
doOpen	Bool	Команда на контактор. Открыть
doClose	Bool	Команда на контактор. Закрыть
Auto	BOOL	Активен режим Авто
HL_Open	BOOL	Лампа на двери "Открыта"
HL_Close	Bool	Лампа на двери "Закрыта"

Выходные выходные	Тип	Описание
udt	udtValve	Для передачи информации в Прикладное программное обеспечение ACU-Prog-НМІ

Пользовательские типы данных:**udtValve****STRUCT**

```

hRemote : BOOL; // Дистанционный режим управления с НМІ (При условии diAuto=1)
hStop : Bool; // Блокировка Стоп. Приоритет над siOpen и siClose
hOpen : Bool; // Команда в дист Открыть: Импульс
hClose : Bool; // Команда в дист Закреть: Импульс
hReset : Bool; // Сброс аварии с НМІ: Импульс
hStatus : Int; // Состояние: 0=Закреть, 1=Открыть, 2=Промежуточное, 3=Неопределенное
hMode : Int; // Режим управления: 0=Местный, 1=Автоматический, 2=Дистанционный
hAlmCom : Bool; // Общая авария
hAlmCmd : Bool; // Невыполнение команды
hAlmKonc : Bool; // Авария концевых выключателей
hAlmQF : Bool; // Сработал автомат защиты двигателя
hAlmStator : Bool; // Перегрев статора
hAlmWater : Bool; // Протечка
hAlmMoment : Bool; // Момент превышен
hAlmExt : Bool; // Другая авария
hSi : BOOL; // Активна блокировка
hDirection : INT; // Направление хода: 0=Закрывается, 1=Открывается, 3=Стоп
hTimeCmd : DINT := 90; // Время хода, с

```

END_STRUCT