

ООО «Миконт»

Универсальный контроллер

СМАРТ

Руководство разработчика

МФКЕ.425200.003 РР

Версия 1

г. Пермь, 2018 г.

МФКЕ.425200.003 РР

ВВЕДЕНИЕ

Настоящее руководство по эксплуатации (РЭ) предназначено для разработки ПО контроллеров SMART серий В02, В03, В04. Руководство содержит описание контроллера, основные характеристики, принципы разработки ПО, инструкции по настройке и обслуживанию, а так же сведения для правильной и безопасной его эксплуатации. Изделие соответствует требованиям технических условий МФКЕ.425200.003 ТУ.

Данное руководство создано для служебного пользования, для разработчиков программного обеспечения.

1. Общее описание

Универсальный контроллер предназначен для сбора, регистрации и передачи данных в системах коммерческого, технического учета, системах АСУТП. Все модификации головного модуля контроллера снабжены цифровым дисплеем и имеют кнопки управления.

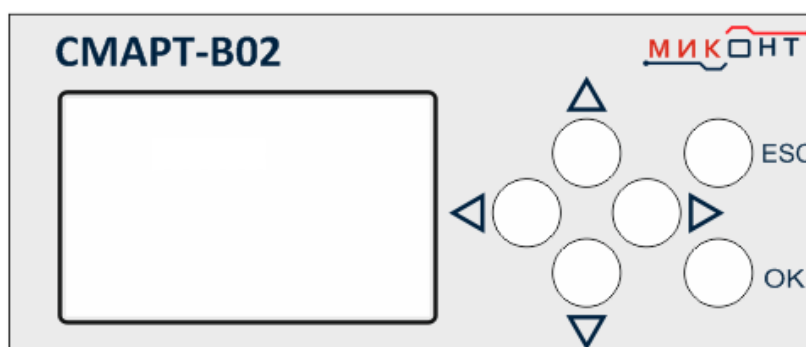


Рисунок 1 – Общий вид панели контроллера

В зависимости от модификации контроллер может быть счетчиком газа, счетчиком воды, пара, дизельного топлива, регистратором технологических параметров, контроллером автоматики и управления, устройствам связи с объектом(УСПД).

Существует два варианта разработки Программное обеспечение:

- Разработка в среде разработки Veremiz семейство языков IEC 61131-3.
- Разработка на языке C99 — стандарт языка программирования Си. Определен в ISO/IEC 9899:1999, современная версия - ISO/IEC 9899:1999/Cor 3:2007 от 2007-11-15

1. Введение

1.1 Структура ОС

Программного обеспечения Smart-OS(далее ОС) разработано на базе FreeRTOS — многозадачная операционная система реального времени (ОСРВ) для встраиваемых систем. Портирована на 35 микропроцессорных архитектур. Распространяется под лицензией MIT с 2017 года[1]. До 2017 года распространялась под модифицированной лицензией GPL с исключением, позволяющим разработчику присвоить модифицированный код операционной системы. Программное обеспечение ОС позволяет обрабатывать входные сигналы с плат УСО(устройство связи с объектом), выдавать управляющие воздействия, отображать информацию на графическом экране, обрабатывать запросы по интерфейсам связи.

Также в состав контроллера может входить специализированное программное обеспечение bootloader. Bootloader позволяет при старте контроллера загружать ПО по интерфейсу usb или ethernet(начиная с версии 0.3)

Таблица 1 - адреса загрузки ОС

Адрес начала	с bootloader v 0.3	с bootloader v 0.2	без bootloader
smart_os	4000h	2000h	0000h

Контроллер для коммерческого учета может поставлять без возможности программирования по USB и Ethernet. Не влияет на работоспособность.

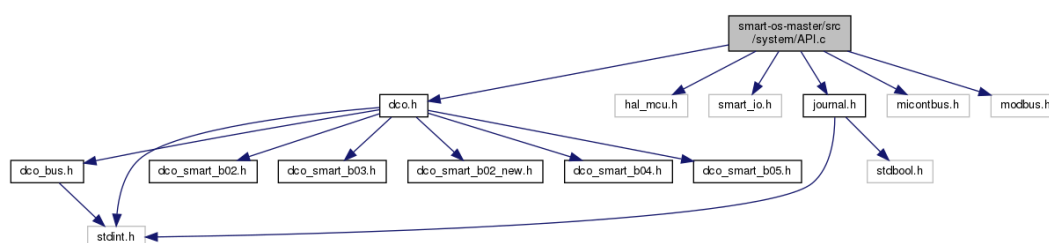


Рисунок 1 — структура ОС

1.2 Модули ОС

Модули	Назначение	Примечание
dco	Модуль связи с УСО производства Миконт	
hmi	Подсистеме отображения	
api	Стандартные функции ОС	Описание в файле «API.pdf»
journal	Подсистема работы с журналом	Описание в файле «1 Структура журнала v1.2.0.pdf»
Modbus	Протоколы Modbus RTU, Modbus TCP	Карта регистров формируется программистом прикладного ПО
Micontbus	Протоколы Micontbus	
Httpd	Мини веб-сервер	

- Модули поставляются скомпилированные в библиотеку smart-os.lib.
- Загрузчик поставляется в виде файла bootloader.hex.

1.3 Сервисное ПО

Наименование	Назначение	Примечание
USOFnc	ПО для конфигурации и юстировки	Поставляется по официальному запросу от метрологического центра
MicontTool	Универсальное ПО для чтение регистров	ОС Linux, Windows
MicontOPC Server 3.0 DA	OPC сервер стандарта DA	
Micont OPC UA	OPC сервер стандарта Unified Architecture	ОС Linux, Windows
Fork	Программа чтение журналов	ОС Linux, Windows
Spoon	Программа чтение журналов	ОС Windows
MicVideo	Программа чтение видеокадров контроллера	

2. Разработка системы на языках IEC 61131-3

Среда разработки Beremiz предназначена для создания и отладки прикладных программ на языках стандарта IEC 61131-3 для целевых устройств на базе отечественных микропроцессоров с архитектурой SPARC. В качестве языков описания алгоритмов и логики работы данных программ, могут выступать как текстовые Structured Text (далее ST) и Instruction List (далее IL), так и графические Function Block Diagram (далее FBD), Ladder Diagram (далее LD), Sequential Function Chart (далее SFC).

Техническими требованиями для работы среды разработки Beremiz является персональный компьютер с тактовой частотой процессора 1000 МГц (поддерживаются как 32-битные, так и 64- битные), 1 Гб

оперативной памяти и установленная операционная система Windows XP/Vista/7. Необходимо наличие монитора (для оптимальной работы не меньше 17"), клавиатуры, мыши (или устройства, полноценно

заменяющего мышь). Среда разработки Beremiz может выполняться на операционных системах Windows и Linux. Она написана с использованием кроссплатформенных языков Python, C, C++ и дополнительных библиотек к ним. Поэтому для запуска и работы Beremiz, необходима сборка интерпретатора Python с определённым набором установленных пакетов (библиотек).

Система разработки Beremiz поставляется по запросу, совместно с контроллерами. Описание среды Beremiz находится в папке «3 Beremiz».

3. Разработка системы на языке СИ

При необходимости разработки сложного программного обеспечения возможно использовать ОС в виде подключаемого файла smart-os.lib и разрабатывать ПО на языке Си. Описание функций и стандарт находится в папке “2 smart-api” Сборка системы для контроллеров на базе Keil uVision или Qt5.9 gcc.

1. Запускаете Keil uVision или Qt Creator 5.9.
2. Открываете keil-project\MDK-ARM\smart-app.uvproj(либо qt-project\smart-app.pro)
3. Открываете default_values.c и default_values.h
4. В default_values.h подробно описано, как сформировать структуру значений default_values, установить пароли изготовителя и инженера, а также заводской номер.
5. Изменяете структуру в default_values.c и, если нужно, в отдельном редакторе изменяете файл src\app\app_description.inc. Файл в кодировке Windows-1251, туда добавляется информация об организации, сервис-центре и т. д.
6. Для получения прошивки в формате bin (smart.bin) выбираете назначение (target) BIN-B02-B04 (для новых B02 и B04, а для старых, соответственно, BIN-OLD-B02) и пересобираете проект (Rebuild all). Зашиваете его через usb или через Ethernet.
7. Для получения прошивки в формате hex (smart.hex) выбираете назначение (target) HEX и пересобираете проект