Контроллер ИП – Структура алгоритмов

Содержание

[1 Принцип построения системы управления Инвертором 2](#_Toc82720443)

[2 Структура алгоритмов основного микроконтроллера 4](#_Toc82720444)

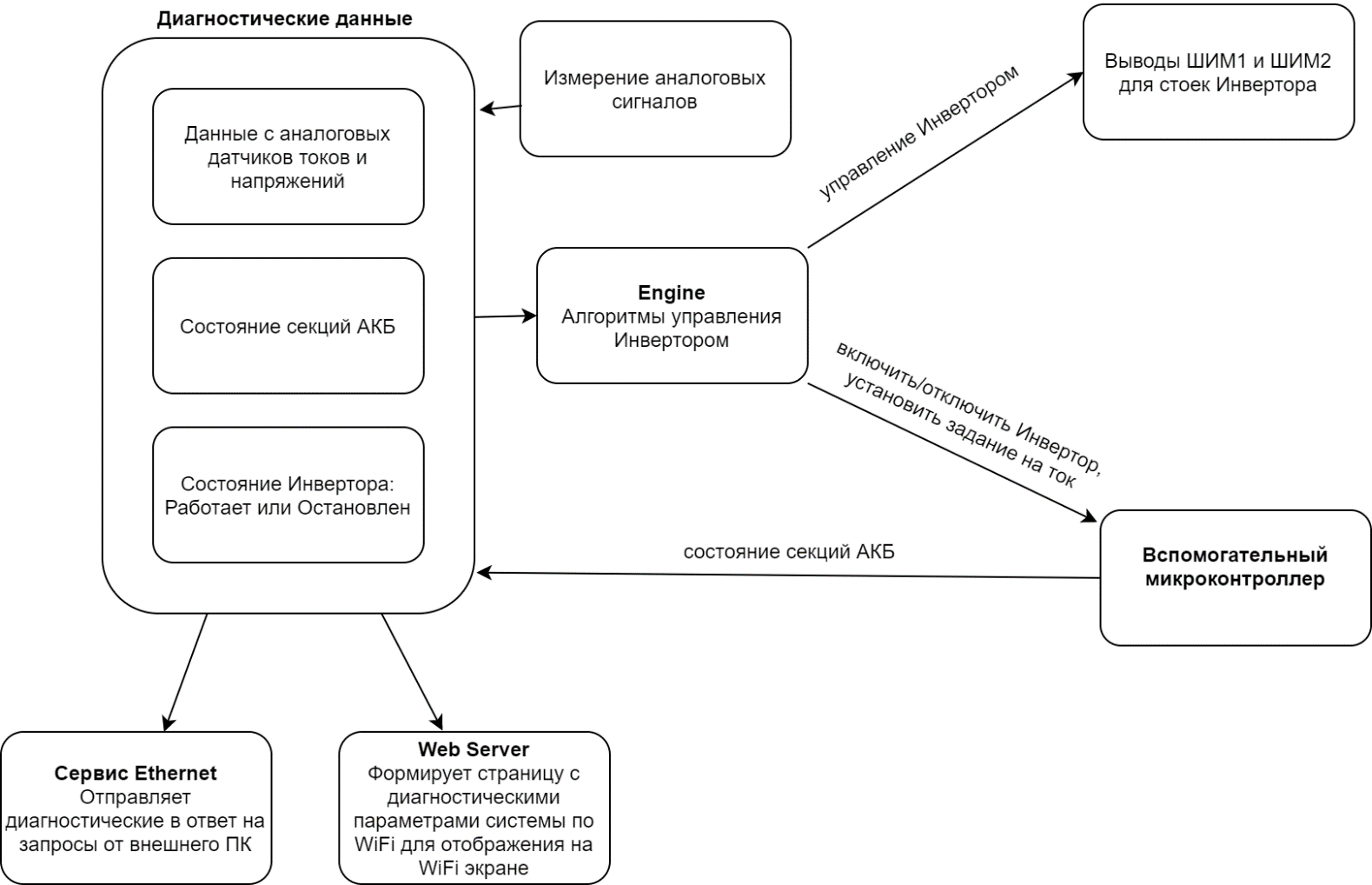
[2.2 Диагностика 5](#_Toc82720445)

[2.2 Управление Инвертором 7](#_Toc82720446)

[3 Структура алгоритмов вспомогательного микроконтроллера 9](#_Toc82720447)

# 1 Принцип построения системы управления Инвертором

Контроллер ИП построен на работе двух микроконтроллеров общего назначения, каждый из которых выполняет свои функции в рамках общей задачи управления режимами работы Источника Питания (ИП) и диагностики состояния его компонентов.



*Рисунок 1 – Инфорамционная модель системы управления инвертором*

Основной микроконтроллер ESP32 WROOM от компании Espressif предназначен для:

* формирования ШИМ сигналов на преобразователь,
* измерения необходимых аналоговых сигналов с датчиков тока и напряжения,
* вывода диагностической информации на дисплей посредством WiFi,
* передачи диагностической информации посредством Ethernet,
* обработки информации о состоянии аккумуляторных батарей.

Вспомогательный микроконтроллер Atmega328P от компании Microchip предназначен для:

* сбора данных о состоянии исправности секций аккумуляторных батарей,
* восстановления работы БМС после сигнала аварии,
* выдачи сигнала запуска/останова инвертора,
* выдачи сигнала запуска/останова заряда аккумуляторных батарей,
* формирования аналогового сигнала задания на ток.

Взаимодействие между двумя микроконтроллерами осуществляется посредством интерфейса UART.

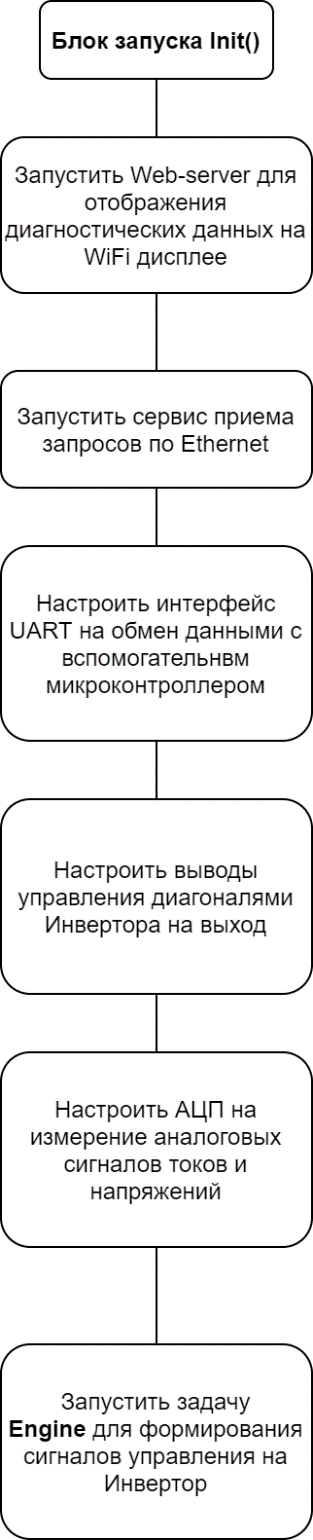
Посредством UART основной микроконтроллер получает от вспомогательного информацию о состоянии секций аккумуляторных батарей и отдает команды на включение и отключение инвертора и зарядного устройства.

# 2 Структура алгоритмов основного микроконтроллера

Программное обеспечение основного микроконтроллера построено на работе операционной системы FreeRTOS. Все структурные элементы алгоритмов программы управления Инвертором выполнены в виде задач.

Задачи представляют собой отдельные функции, которые циклически вызываются планировщиком задач операционной системы.

Прежде, чем будут запущены все функциональные задачи операционной системы, необходимо настроить работу аппаратной части микроконтроллера и проинициализировать ядро операционной системы. Эта процедура производится в блоке запуска Init:



*Рисунок 2 – Блок запуска, настройка системы*

## 2.2 Диагностика

Диагностика построена на работе 3х компонентов:

* Сбор данных аналоговых датчиков и получения состояния аккумуляторных батарей,
* Отправка данных на экран по WiFi,
* Отправка диагностических данных на внешний ПК по Ethernet.

Сбор данных аналоговых датчиков построен на работе АЦП.

Получение данных о состоянии работоспособности секций аккумуляторных батарей производится при помощи UART, на который эти данные пересылает вспомогательный микроконтроллер.

Этот компонент выполнен в рамках задачи StateAcquisition:



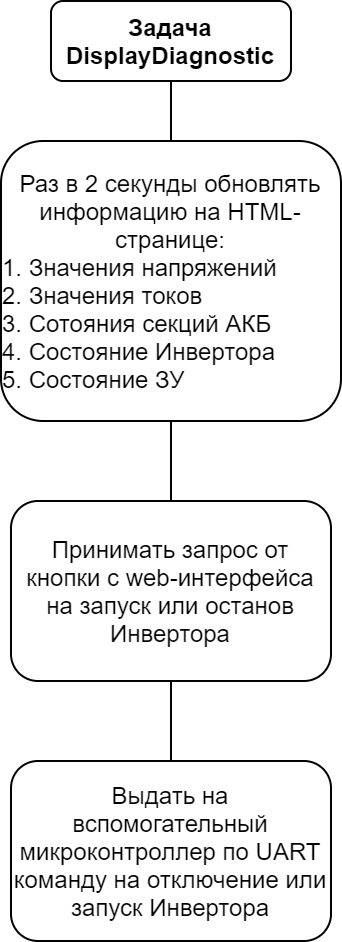
*Рисунок 3 – Структура задачи StateAcquisition*

Отправка диагностических данных и состояния системы на WiFi экран осуществляется при помощи задачи Web-сервера, который транслирует полученную информацию в виде HTML страницы по WiFi.

Web-сервер обновляет данные с частотой раз в 2 секунды.

Со страницы диагностических данных при помощи элемента управления на WiFi экране можно запустить и остановить работу Инвертора.

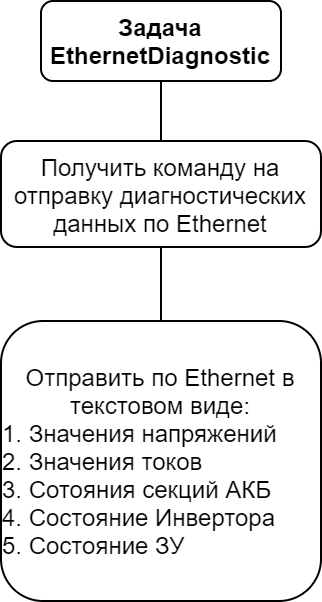
Этот компонент реализуется при помощи задачи DisplayDiagnostic:



*Рисунок 4 – Задача DisplayDiagnostic*

Диагностическую информацию можно запросить при помощи внешнего ПК. Для этого предусмотрен интерфейс Ethernet.

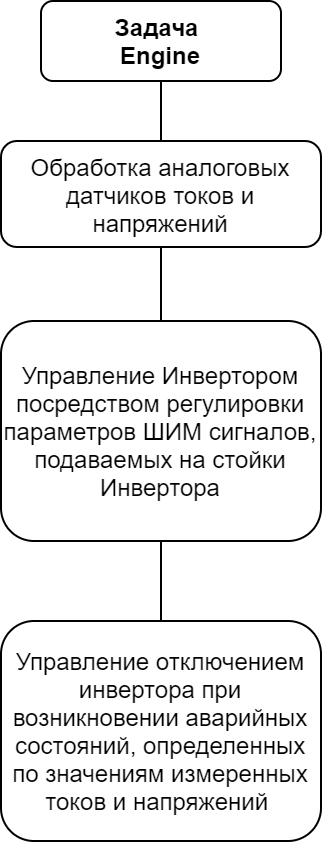
Отправка диагностических данных на внешний ПК посредством интерфейса Ethernet выполняется при помощи задачи обслуживания запросов интерфейса Ethernet – EthernetDiagnostic:



*Рисунок 5 – Задача DisplayDiagnostic*

## 2.2 Управление Инвертором

Управление работой инвертора осуществляет задача Engine:



*Рисунок 6 – Задача Engine*

Эта задача реализует как управление самими ключами Инвертора, так и сигналами включения и отключения Инвертора и зарядного устройства.

Сигналы включения и отключения поступают через вспомогательный микроконтроллер. Вспомогательнвй микроконтроллер воспринимает команды посредством интерфейса UART

# 3 Структура алгоритмов вспомогательного микроконтроллера

Программное обеспечение вспомогательного микроконтроллера выполнено с применением стандартных библиотек от компании Microchip и среды разработки Microchip Studio для микроконтроллеров 8 bit AVR.



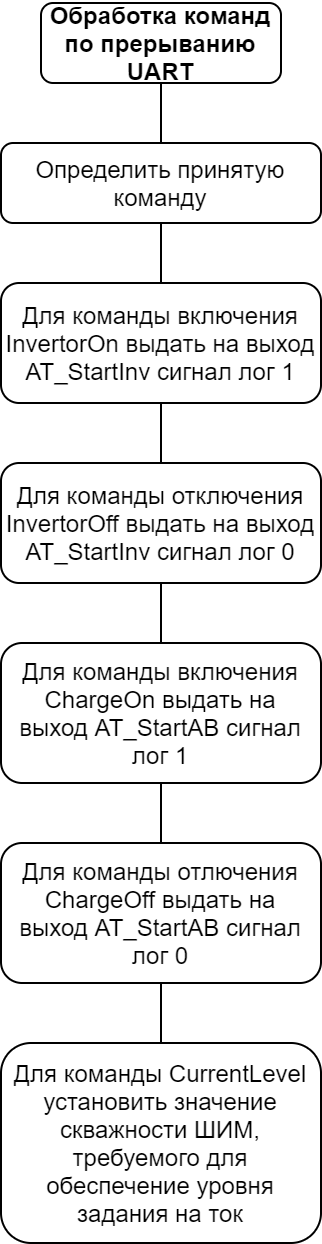
*Рисунок 7 – Блок инициализации вспомогательного микроконтроллера*

Основной цикл работы микроконтроллера выполняет постоянное считывание состояний аккумуляторных батарей и отправку их состоянини на основной микроконтроллер посредством интерфейса UART:



*Рисунок 8 – Основной цикл работы*

Команды на запуск и остановку Инвертора и зарядного устройства микроконтроллер принимает посредством обработчика прерывания UART и исполняет их в нем же:



*Рисунок 9 – Блок обработки команд основного микрококнтроллера*